



VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
BRNO UNIVERSITY OF TECHNOLOGY



FAKULTA PODNIKATELSKÁ
ÚSTAV INFORMATIKY

FACULTY OF BUSINESS AND MANAGEMENT
INSTITUTE OF INFORMATICS

TVORBA DATABÁZE SLUŽEB PRO CESTOVNÍ AGENTURY

CREATION OF DATABASE SERVICES FOR TRAVEL AGENCIES

BAKALÁŘSKÁ PRÁCE
BACHELOR'S THESIS

AUTOR PRÁCE
AUTHOR

PETR KADLČÍK

VEDOUCÍ PRÁCE
SUPERVISOR

Ing. JIŘÍ KŘÍŽ, Ph.D.

BRNO 2014

ZADÁNÍ BAKALÁŘSKÉ PRÁCE

Kadlčík Petr

Manažerská informatika (6209R021)

Ředitel ústavu Vám v souladu se zákonem č.111/1998 o vysokých školách, Studijním a zkušebním řádem VUT v Brně a Směrnicí děkana pro realizaci bakalářských a magisterských studijních programů zadává bakalářskou práci s názvem:

Tvorba databáze služeb pro cestovní agentury

v anglickém jazyce:

Creation of Database Services for Travel Agencies

Pokyny pro vypracování:

Úvod

Cíle práce, metody a postupy zpracování

Teoretická východiska práce

Analýza současného stavu

Vlastní návrhy řešení

Závěr

Seznam použité literatury

Přílohy

Seznam odborné literatury:

CONOLLY, T., C. BEGG a R. HOLOWCZAK. Mistrovství – databáze. Brno: Computer Press, 2009. ISBN 978-80-251-2328-7.

GROFF, J. a P. WEINBERG. SQL Kompletní průvodce. 1. vy. Brno: Computer Press, 2005. 936 s. ISBN 80-251-0369-2.

KOCH, M. a B. NEUWIRTH. Datové a funkční modelování. 4. rozš. vyd. Brno: Akademické nakladatelství CERM, 2008. ISBN 978-80-214-4125-5.

KŘÍŽ, J. a P. DOSTÁL. Databázové systémy. Brno: Akademické nakladatelství CERM, 2005. ISBN 80-214-3064-8.

Vedoucí bakalářské práce: Ing. Jiří Kříž, Ph.D.

Termín odevzdání bakalářské práce je stanoven časovým plánem akademického roku 2013/2014.

L.S.

doc. RNDr. Bedřich Půža, CSc.
Ředitel ústavu

doc. Ing. et Ing. Stanislav Škapa, Ph.D.
Děkan fakulty

V Brně, dne 29.05.2014

Abstrakt

Tato bakalářská práce se zaměřuje na návrh a vytvoření databáze v jazyku SQL. Výsledný návrh databáze by měl najít uplatnění v oblasti cestovního ruchu, kde by zjednodušil problematickou situaci, která se týká ověřování volných ubytovacích kapacit, kapacit přepravních společností a tvorby rezervací. Součástí práce je analýza současného stavu, specifikace problému, teoretická východiska a vlastní návrh řešení.

Abstract

This thesis focuses on the design and creation SQL database. The resulting database design should find application in the field of tourism, which would simplify the problematic situation, relating to the verification of free accommodation capacity, transportation capacity and making reservation. Part of this work is to analyze the current situation, the problem specification, theoretical background and custom design solutions.

Klíčová slova

SQL, databáze, relace, entita, data, proces

Key words

SQL, database, relation, entity, data, process

Bibliografická citace práce

KADLČÍK, P. *Tvorba databáze služeb pro cestovní agentury*. Brno: Vysoké učení technické v Brně, Fakulta podnikatelská, 2014. 78 s. Vedoucí bakalářské práce Ing. Jiří Kříž, Ph.D.

Čestné prohlášení

Prohlašuji, že předložená bakalářská 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

.....

Poděkování

Tímto děkuji svému vedoucímu bakalářské práce, panu Ing. Jiřímu Křížovi, Ph.D., za jeho cenné rady a připomínky při zpracování této bakalářské práce a Ondřeji Výmolovi za rady při psaní vlastního návrhu řešení.

Obsah

ÚVOD	9
CÍL A METODIKA PRÁCE	10
Cíl práce	10
Metodika práce.....	10
1 TEORETICKÁ VÝCHODISKA PRÁCE	11
1.1 Základní pojmy	11
1.1.1 Data	11
1.1.2 Informace	11
1.1.3 Entita	11
1.1.4 Atribut	11
1.1.5 Relace.....	11
1.1.6 Primární klíč.....	12
1.1.7 Kandidátní klíč	12
1.1.8 Cizí klíč	12
1.2 Databáze	12
1.2.1 Databázový systém	12
1.2.2 Datové modely	13
1.2.3 Relační databáze	14
1.2.3 Normalizace	14
1.2.4 Entitně – relační model	15
1.2.5 Kardinalita vztahů mezi entitami	15
1.2.6 Životní cyklus vývoje databázového systému	17
1.2.7 Metodologie návrhu databáze	18
1.3 Jazyk SQL	19
1.3.1 Cíle SQL	19

1.3.2	Datové typy	20
1.3.3	Operace s tabulkami	22
1.3.4	Operace s daty	22
1.3.5	Pokročilé databázové objekty	22
2	ANALÝZA SOUČASNÉHO STAVU	23
2.1	Popis současné situace	23
2.1.1	Cestovní agentura	23
2.1.2	Cestovní kancelář	24
2.2	Popis interních procesů cestovní agentury	24
2.2.1	Proces vytvoření objednávky	25
2.2.2	Proces zpracování objednávky	25
2.2.3	Proces dokončení objednávky	26
2.3	Popis rezervačních a databázových systémů cestovních kanceláří a agentur ..	27
2.3.1	Magicware	27
2.3.2	Open Travel Network	29
2.3.3	Stovka software	29
2.3.4	CeSYS	30
2.3.5	Pear	31
2.4	Analýza problému v praxi	31
2.4.1	Komplikace při evidování nestandardních situací	32
2.4.2	Špatná informovanost	32
2.4.3	Problém s rezervacemi	32
3	Vlastní návrh řešení	33
3.1	Definice požadavků a procesů	33
3.1.1	Běžný chod společnosti – hlavní procesy	33
3.1.2	Vznik objednávky	35

3.1.3	Vyřízení doplňkových služeb, platby a smluv	36
3.2	Konceptuální návrh	38
3.2.1	Identifikace entit	38
3.2.2	Identifikace relací.....	39
3.2.3	ER Diagram.....	40
3.3	Logický návrh	41
3.3.1	Zákazník, prodejce a objednávka.....	41
3.3.2	Rezervace	42
3.3.3	Let, letiště a dopravní společnost.....	43
3.3.4	Země, lokalita, ubytování a pokoj.....	44
3.3.5	Cestovní kancelář, zájezd, pokoj a let.....	45
3.3.6	Cestující, let a typ pojištění.....	47
3.3.7	Objednávka, let, pokoj a zájezd	47
3.3.8	Objednávka a parkování.....	48
3.4	Fyzický návrh.....	49
3.4.1	Procedura – přidání zákazníka	49
3.4.2	Procedura – přidání nového letu	50
3.4.3	Pohled – Souhrn objednávky	51
	Závěr	52
	SEZNAM POUŽITÝCH ZDROJŮ	53
	SEZNAM TABULEK	55
	SEZNAM OBRÁZKŮ	55
	SEZNAM PŘÍLOH.....	56
	Příloha1 : Fyzický návrh databáze	57
	Příloha 2: Datový slovník	58
	Příloha 3: Zdrojový kód	63

ÚVOD

Cestovní agentury jsou společnosti zabývající se prodejem zájezdů, dovolených, různých typů ubytování, výletů, přepravy a jiných dalších služeb a produktů, které jsou v dnešní době prodávány v hojné míře převážně na internetu. Tyto cestovní agentury mají výhodu hlavně v tom, že prodávají služby od více cestovních kanceláří a dávají tak zákazníkovi možnost rozhlédnout se na trhu a vybrat si variantu, která je pro něj nejlepší a cenově nejvýhodnější. Cestovním agenturám tyto služby poskytují cestovní kanceláře a ty je čerpají přímo ve vybraných destinacích u jednotlivých hotelů a leteckých společností, s nimiž spolupracují. Tím pádem se z tohoto prodeje stává řetězec, kde na jednom konci hraje svoji úlohu hotel a na druhé straně zákazník, který poptává dovolenou u cestovní agentury.

Celý tento proces prodeje je zdoluhavý, jelikož některé cestovní kanceláře nemají rezervační systém, přes který by si cestovní agentury mohly ověřit informace o dostupnosti služby poptávané zákazníkem a musí tento úkol vykonávat v lepším případě telefonicky nebo mailem. Ovšem i v případě že cestovní kanceláře tento rezervační systém mají, neznamená to, že informace v něm jsou aktuální. Tímto funkčním a aktuálním systémem disponují převážně velké společnosti. Vinu na tom nenesou ale jen cestovní kanceláře, jelikož se stává, že i ty čekají na odpověď z destinace někdy i několik hodin, což může v letních měsících kdy je hlavní sezóna a rychlost hraje velkou roli společnost připravit o netrpělivé zákazníky.

Proto cílem mé bakalářské práce je navrhnout a vytvořit efektivní databázový systém, který by mohl sloužit v oblasti cestovního ruchu pro cestovní agentury, jehož by využívaly pro snadné a rychlé prověření dostupnosti ubytovacích kapacit, kapacit různých přepravních společností, tvorby rezervací a informování o nabízených službách.

CÍL A METODIKA PRÁCE

Cíl práce

Cílem mé práce je navrhnout a vytvořit databázi obsahující informace o ubytovacích zařízeních včetně počtu jednotlivých typů pokojů, jejich obsazenosti a možnosti rezervace. Dále by tato databáze sloužila k dohledání informací o přepravních kapacitách do těchto destinací a k prověření dostupnosti parkovacích míst na soukromých parkovištích přímo na letišti. Do databáze by tyto informace vkládali přímo poskytovatelé jednotlivých služeb a výstup by tak sloužil pro jednotlivé cestovní agentury k rychlému dohledání obsazenosti objektu.

Metodika práce

Postupovat při vytvoření mé práce bych začal úvodním zmapováním aktuální situace ve zmiňované oblasti, zaměřením se na problematickou část, teoretickým návrhem kterýmž by se tento stav dal řešit a nakonec vlastním návrhem pro řešení této situace s využitím dotazovacího jazyka SQL a znalostmi databázových systému a datového modelování.

1 TEORETICKÁ VÝCHODISKA PRÁCE

V této části se seznámíme s teoretickými znalostmi týkající se databází a jazyka SQL od čehož se bude dále odvíjet celá bakalářská práce.

1.1 Základní pojmy

1.1.1 Data

Nezpracovaná fakta, která mají určitou důležitost pro jednotlivce nebo pro nějakou instituci [1]. Jestliže je člověk vystaven působením zpráv a některým z nich porozumí, tak se pro něj stávají daty. Pokud se data používají k rozhodování, stávají se z nich informace, jelikož se jim přiřazuje význam a smysl [2].

1.1.2 Informace

Zpracovaná fakta, která mají určitou strukturu a význam pro jednotlivce nebo organizaci [1]. Znalosti a informace jsou dnes jediným smysluplným zdrojem, který snižuje rozhodovací neurčitost [2].

1.1.3 Entita

Množina objektů se stejnými vlastnostmi, které uživatel nebo organizace označí jako nezávisle existující objekty [1]. V podstatě se jedná o cokoliv o čem potřebujeme uchovávat nějaké informace [3].

1.1.4 Atribut

„Vlastnosti entity se nazývají atributy. Atributy představují to, co můžeme o entitách vědět“ [1, s. 159]. Jednoduchý atribut je atomický (nedělitelný) a skládá se z jedné komponenty, například jméno. V opačném případě se jedná o atribut složený z více komponent, například adresa [1, 2].

1.1.5 Relace

Relace je množina smysluplných spojení mezi zúčastněnými entitami. Každá relace by měla být v rámci množiny jedinečně identifikovatelná. Všechny relace jsou označeny jménem, které popisují jejich funkci. Relace zobrazujeme jako čáru spojující související entity a označujeme ji jménem relace. Počet atributů relace označujeme jako stupeň relace a počet entit jako kardinalitu relace [1, 2].

1.1.6 Primární klíč

Primární klíč je atribut nebo množina atributů, jejichž hodnoty jednoznačně identifikují každý z řádků relace. Každý primární klíč musí splňovat tyto vlastnosti:

- Být jednoznačný, tzn. že v relaci neexistuje druhá entita, která by pro tuto množinu atributů měla stejné hodnoty.
- Být minimální, tzn. že žádný atribut není možné vypustit, aniž bychom porušili jednoznačnost [1, 2].

1.1.7 Kandidátní klíč

Kandidátní klíč je stejný jako primární klíč se stejnými vlastnostmi, ale není vybrán jako primární klíč. V každé relaci může být více kandidátních klíčů, z nichž jeden je vybrán jako primární klíč a ostatní kandidátní klíče se nazývají alternativní [2].

1.1.8 Cizí klíč

Cizí klíč je atribut relace, který splňuje následující nezávislé vlastnosti:

- Každá hodnota je buď plně zadaná nebo plně nezadaná.
- Existuje jiná relace s takovým primárním klíčem, že každá zadaná hodnota cizího klíče je identická s hodnotou primárního klíče nějaké entity této jiné relace [2].

1.2 Databáze

Databáze je velké uložisko dat, které může být používáno více uživateli současně. Všechna data v databázi jsou integrována s minimálním množstvím duplikací [1].

1.2.1 Databázový systém

„Kolekce databázových aplikací, které integrují s databází spolu, DBMS a databáze samotná“ [1, s. 39]. Každý databázový systém se skládá ze tří částí: systému řízení bez dat (SŘBD), databázové aplikace a databáze [3].

- SŘBD

Což znamená systém řízení báze dat (neboli DBMS podle anglického Database Management System). Je to část databázového systému, která má na starost integritu s

uživatelé, databázovou aplikací a databází. Mimo jiné umožňuje uživatelům vkládat, aktualizovat, mazat a vyvolávat data z databáze [1].

- Databázová aplikace

„Počítačový program integrující s databází vyvoláním odpovídajícího požadavku (typicky jednoho nebo více příkazů SQL) pro DBMS.“ [1, s. 39] tento program umožňuje vybírat, prohlížet a aktualizovat informace uložené prostřednictvím SŘBD [1,3].

- Databáze

Databáze čili báze dat je částí databázového systému, kterou můžeme chápat jako velké úložiště uspořádaných dat. Toto úložiště může být používáno současně mnoha uživateli zároveň [1,3].

1.2.2 Datové modely

Jednotlivé datové modely jsou popisem toho, jak jsou data zpřístupňována uživateli a programátorovi [2].

- Lineární datový model je jediný datový model, který můžeme implementovat na libovolném médiu. Systém selhává v případě potřeby křížových odkazů nebo zpracování informací. Mezi jednotlivými skupinami objektů není žádná vazba [1, 2].
- Hierarchický datový model je takový, kde jsou data strukturována hierarchicky a znázorněna v podobě obráceného stromu. Vztah je v tomto modelu reprezentován termíny rodič a potomek. Uživatel pak může k záznamům přistupovat hierarchicky od kořenové tabulky, přes stromovou strukturu až ke hledaným datům [6].
- Síťový datový model je vyjádřen v pojmech uzlů a množinových struktur. Uzel představuje soubor záznamů a množinová struktura vztah v síťové databázi. Tato konstrukce vytváří vztah mezi dvěma uzly tak, že jeden uzel je definován jako vlastník a druhý jako prvek [7].
- Relační datový model je v současnosti nejpoužívanější datový model. Tento model vzniká spojením pomocí několika položek, které nazýváme relačními klíči. Relační

datový model nám umožňuje zachytit nejenom objekty ale i vzájemné vztahy těchto objektů což nám umožňuje přiblížit se více reálnému světu [2].

1.2.3 Relační databáze

Základním pojmem je relace. Tu si lze představit jako tabulku, která se skládá ze sloupců a řádků. Sloupce odpovídají jednotlivým vlastnostem a údaje v řádku zaznamenávají aktuální stav světa. Řádek této tabulky bývá označován databázovým záznamem. Celý soubor tabulek (relací) pak tvoří databázi (relační schema). Tvorba tabulek je důležitý úkol především z hlediska dlouhodobého [3].

1.2.3 Normalizace

Normalizace je činnost, která upravuje návrh datové struktury tak, aby splňovala zvolené normy. Tyto normy vycházejí z požadavku na efektivní ukládání dat a minimální redundanci. Pokud datový model porušuje některou z normálních forem, není navržen optimálně.

- První normální forma - nám říká že všechny atributy entity musí být jednoduché nikoli složené nebo vícehodnotové. Databáze je nejprůhlednější jsou-li její prvky jednoduché.
- Druhá normální forma - platí tehdy, platí-li první normální forma a všechny její atributy jsou závislé na celém kandidátním (primárním) klíči [2].
- Třetí normální forma - platí, pokud je relace v druhé normální formě, ve které jsou všechny její hodnoty ve sloupcích závislé pouze na primárním klíči a na žádném jiném sloupci závislé nejsou [1].
- Boyce – Coddova normální forma - pokud je relace v Boyce-Coddově normální formě, je také ve třetí normální formě nikoli obráceně. Boyce-Coddova normální forma je platná tehdy, pokud mezi kandidátními klíči není žádná funkční závislost a to za těchto podmínek:
 1. Relace musí mít nejméně dva kandidátní klíče.
 2. Nejméně dva z kandidátních klíčů musí být složené.
 3. Kandidátní klíče se v některých attributech musí překrývat.

- Čtvrtá normální forma - platí pokud je relace v Boyce-Coddově normální formě a všechny její vícehodnotové závislosti jsou i funkčními závislostmi z kandidátních klíčů.
- Pátá normální forma - nám vyjadřuje cyklické omezení. Mějme relaci X, relaci Y a relaci Z. Pokud je relace X spojena s relací Y, relace Y spojena s relací Z a relace Z spojena opět s relací X, pak všechny tři entity musí být součástí stejného vektoru hodnot [2].

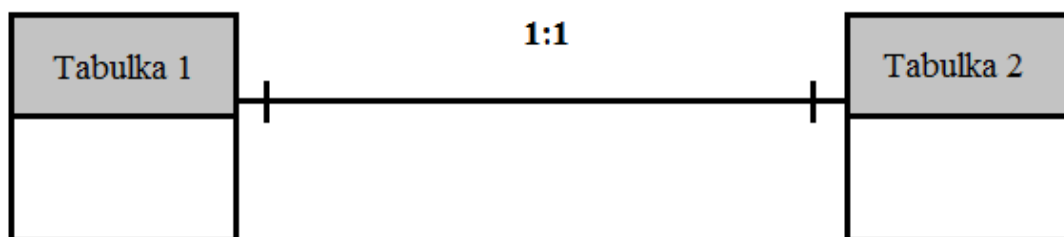
1.2.4 Entitně – relační model

Entitně relační model představuje množinu pojmů, pomocí nichž popisujeme příslušnou aplikaci za účelem následné specifikace struktury databáze. Proces návrhu system spočívá v identifikaci typů entit jako množin objektů stejného typu, v identifikaci typů vztahů a v přiřazení atributů, které blíže popisují vlastnosti jednotlivých entit a vztahů [2].

1.2.5 Kardinalita vztahů mezi entitami

Relace, které vznikají mezi tabulkami popisují vztahy mezi objekty reálného světa. Při návrhování databázových tabulek, můžeme využít několik druhů vztahů.

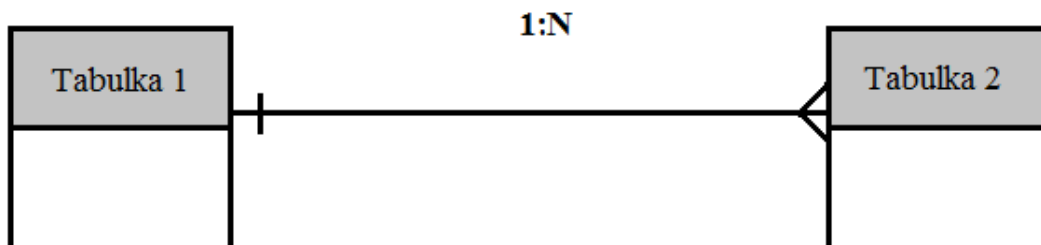
- Vztah jedna ku jedné. První entitě záznamu v tabulce odpovídá jen jeden záznam v druhé tabulce, tedy maximálně jen jedna druhá entita. Každý řádek první tabulky je tedy možné svázat jen s jedním řádkem druhé tabulky. Tuto relaci zajistíme pomocí unikátních klíčů v obou tabulkách.



Obrázek č. 1: Vztah jedna ku jedné

Zdroj: [8] s. 54

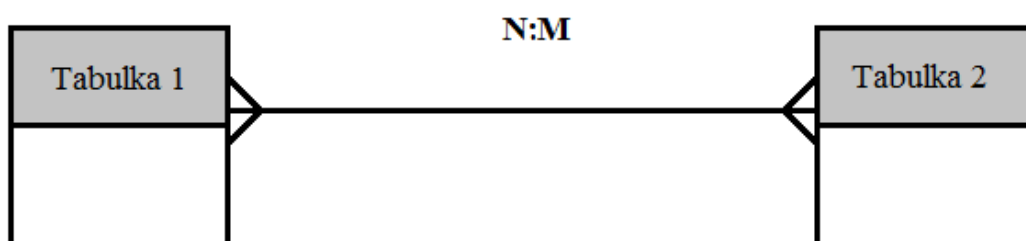
- Vztah jedna ku N. Entitě z první tabulky odpovídá jedna nebo více entit z druhé tabulky. Naopak entitě z druhé tabulky, odpovídá maximálně jedna entita z první tabulky. Při vztahu jedna ku N je velmi důležitý směr.



Obrázek č. 2: Vztah jeden k více

Zdroj: [8] s. 54

- Vztah N ku M. Entitě z první tabulky odpovídá více entit z druhé tabulky, takže i naopak, jedné entitě z druhé tabulky odpovídá více entit z první tabulky. V praxi takový vztah můžeme vypořizovat mezi výrobky a jejich vlastnostmi. Jeden výrobek může mít více vlastností a jednu vlastnost může mít zase více výrobků.

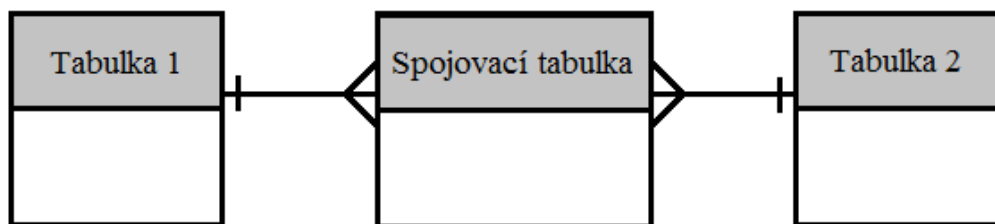


Obrázek č. 3: Vztah více ku více

Zdroj: [8] s. 55

Jelikož databázové systémy nedokážou přímo pracovat se vztahem N:M, řeší se tento problém pomocí dekompozice, což znamená že se mezi tabulku 1 a

tabulku 2 vloží spojovací tabulka a tento vztah se tak rozloží na dva vztahy typu 1:N [8].



Obrázek č. 4: Rozložení vztahu pomocí spojovací tabulky

Zdroj: [8] s. 55

1.2.6 Životní cyklus vývoje databázového systému

Jedná se o uspořádaný seznam fází, který popisuje odpovídající techniky a nástroje, jež je třeba použít při vyvíjení databázového systému. Fáze životního cyklu nenásledují striktně po sobě, ale zahrnují někdy opakování předcházejících fází na základě zpětné vazby.

- Plánování databáze
- Definice systému
- Sběr a analýza požadavků
- Návrh databáze
 - konceptuální návrh, který se pokouší identifikovat důležité objekty
 - logický návrh, během kterého se provede reprezentace těchto objektů
 - fyzický návrh, u nějž rozhodujeme o tom, jak je třeba tyto tabulky implementovat
- Výběr DBMS
- Návrh aplikací
- Vytvoření prototypu databáze
- Implementace
- Konverze a načtení dat
- Testování
- Provozní údržba [1].

1.2.7 Metodologie návrhu databáze

Při metodologii návrhu jde o strukturovaný přístup používající procedury, techniky, nástroje a dokumentaci s cílem podpořit a usnadnit proces návrhu.

- Konceptuální návrh databáze

Konceptuální návrh databáze má cíl vytvořit ER model dat, který představuje reprezentaci datových požadavků organizace. V konceptuálním návrhu se nebere v úvahu jakákoliv implementace. Tento model identifikuje důležité entity a relace, které je třeba reprezentovat v databázi.

Krok 1 vytvoření ER modelu

1. Identifikace entit
2. Identifikace relací
3. Identifikace a spojení atributů s entitami nebo relacemi
4. Určení domén atributů
5. Určení atributů, které budou kandidátními, primárními a alternativními klíči
6. Specializace / generalizace entit
7. Kontrola redundance v modelu
8. Kontrola, zda model podporuje uživatelské transakce
9. Posouzení konceptuálního návrhu databáze s uživateli [1].

- Logický návrh databáze

„Proces vytvoření modelu dat používaných organizací, který je založen na specifickém modelu dat, ale nezávislý na konkrétním DBMS a jiných úvahách o fyzické implementaci“ [1, s. 206].

V této části návrhu databáze mapujeme ER model do množiny relačních tabulek.

Krok 2 mapování ER modelu do tabulek

1. Vytvoření tabulek
2. Kontrola tabulek pomocí normalizace
3. Kontrola zda tabulky podporují uživatelské transakce

4. Kontrola integritních omezení
5. Posouzení logického návrhu databáze s uživateli

- Fyzický návrh databáze

Při fyzickém návrhu databáze se rozhodujeme, jak fyzicky implementujeme logický návrh v prostředí cílového relačního DBMS. Tato část nám umožňuje činit rozhodnutí o způsobu implementace.

Krok 3 Převod logického návrhu databáze do cílového DBMS

1. Návrh podkladových tabulek
2. Návrh reprezentace odvozených dat
3. Návrh zbývajících integritních omezení

Krok 4 Volba organizace souborů a indexů

1. Analýza transakcí
2. Volba organizace souborů
3. Volba indexů

Krok 5 Návrh uživatelských pohledů

Krok 6 Návrh bezpečnostních mechanismů

Krok 7 Zvážení zavedení kontrolované redundance

Krok 8 Monitorování a doladění systému v provozu [1].

1.3 Jazyk SQL

Zkratka SQL značí Structured Query Language. Jedná se o strukturovaný dotazovací jazyk, který v sobě zahrnuje nástroje pro tvorbu databází a další nástroje pro manipulaci s daty. [3, 4].

1.3.1 Cíle SQL

Hlavní cíle jazyka SQL jsou:

- Vytváření struktury databází a tabulek.
- Provádění údržby dat, jako například vkládání modifikace a mazání z tabulek.
- Provádění jednoduchých i složitých dotazů.

Tyto úkoly by měl jazyk SQL zvládat s minimální námahou pro uživatele a měl by odpovídat nějakému uznávanému standardu, aby bylo možné použít stejnou strukturu při přechodu mezi různými databázovými systémy.

Standard SQL ISO je rozdělen na dvě hlavní součásti:

- Data Definitiv Language (DDL) – jazyk pro definici struktury databáze a kontrolu přístupu k datům.
- Data Manipulation Language (DML) – jazyk pro vyvolání a aktualizaci dat [1].

1.3.2 Datové typy

Datové typy můžeme dělit do jednotlivých kategorií:

1. Číselné datové typy

Tab. 1: Číselné datové typy

Datový typ	Místo (bajty)
Tinyint	1
Smallint	2
Int	4
Biont	8
Decimal(n,m)	5-17
Numeric(n,m)	5-17
Smallmoney	4
Money	8
Real	4
Float(n)	4 nebo 8

Zdroj: [5]

2. Znakové (textové) datové typy

Tab. 2: Znakové (textové) datové typy

Datový typ	Místo (bajty), max.
Char(n)	1 bajt na znak, 8000
Varchar(n)	1 bajt na znak, 8000
text	1 bajt na znak, 2GB
Nchar(n)	2 bajty na znak, 4000
Nvarchar(n)	2 bajty na znak, 4000
ntext	2 bajty na znak, 2GB

Zdroj: [5]

3. Datové typy pro datum a čas

Tab. 3: Datové typy pro datum a čas

Datový typ	Přesnost	Místo (bajty)
smalldatetime	1 m	4
Datetime	0,00333 s	8
Datetime2	100 ns	6-8
Datetimeoffset	100 ns	8-10
Date	1 den	3
Time	100 ns	3-5

Zdroj: [5]

4. Binární datové typy

Tab. 4: Binární datové typy

Datový typ	Místo (bajty) – max.
bit	1 bit
binary	8000
varbinary	8000
image	2GB

Zdroj: [5]

1.3.3 Operace s tabulkami

Strukturu tabulek můžeme měnit přidáváním, modifikací nebo odebíráním sloupců. Tabulky, které už jsou vytvořeny, je možné ze systému vymazat nebo seskupovat do různých schémat (kolekcí).

1.3.4 Operace s daty

Vytvořené tabulky je potřeba naplnit daty, provádět jejich aktualizaci a popřípadě i mazat řádky [3].

Příkaz SELECT je nejdůležitějším příkazem jazyka SQL používajících se k vytváření dotazů. Účelem příkazu SELECT je vyvolání a zobrazení dat z jedné či více tabulek [1].

1.3.5 Pokročilé databázové objekty

- Pohledy

Pohled je v podstatě příkaz SELECT, který je označen názvem a uložen do databáze. Jeho hlavní výhoda spočívá v tom, že po vytvoření funguje jako tabulka, kterou mohou využívat další příkazy SELECT.

- Uložené procedury

Jakýkoliv příkaz spuštěný v systému SQL lze zapouzdřit do uložené procedury. Uložená procedura není nic jiného než dávka příkazů jazyka SQL, která je označena názvem a uložena do databáze.

- Funkce

Funkce nám umožňují provádět výpočty, jejichž výsledek lze vrátit volající aplikaci nebo integrovat do sady znaků. Funkce slouží k výpočtům, ale nemohou změnit stav databáze, instance nebo změnit data v tabulce [9].

- Spouště

Spouště neboli trigger, umožňují spouštět kód automaticky na základě určité akce.

2 ANALÝZA SOUČASNÉHO STAVU

Cílem této kapitoly je zmapovat aktuální situaci a identifikovat problém, jehož řešením se budu dále zabývat v následující části.

2.1 Popis současné situace

V této podkapitole bych se chtěl zabývat aktuální situací v oblasti cestovního ruchu a úkoly jednotlivých druhů společností, které v této oblasti figurují.

2.1.1 Cestovní agentura

Cestovní agentura je společnost, která prodává jednotlivé služby cestovních kanceláří. Jde tedy o zprostředkovatele prodeje jednotlivých služeb cestovního ruchu mezi cestovní kanceláří a zákazníkem. Cestovní agentury mají pro své zákazníky stejné ceny jako cestovní kanceláře, které danou službu nabízejí. Jejich příjmy tedy plynou z provize z prodeje, na které se dohodnou individuálně s každou cestovní kanceláří zvlášť. Nesmírnou výhodou cestovních agentur je proto široká nabídka jejich služeb. Ty největší cestovní agentury zprostředkovávají služby až více než 300 cestovních kanceláří. Jelikož cestovní kanceláře si cenu určují sami, zákazník si tak může vybrat tu nejlevnější variantu na trhu.

Rozsah oprávněných činností cestovních agentur:

- Nabídka a prodej jednotlivých služeb cestovního ruchu.
- Organizování, kombinování služeb cestovního ruchu a jejich nabídka a prodej cestovní kanceláři za účelem jejího dalšího podnikání.
- Zprostředkování a prodej jednotlivých služeb cestovního ruchu pro cestovní kanceláře nebo jinou cestovní agenturu, případně pro jiné osoby.
- Zprostředkování prodeje zájezdů cestovním kancelářím, v nichž musí být cestovní smlouva uzavřena jménem cestovní kanceláře, pro kterou je zájezd zprostředkován.
- Prodej věcí související s cestovním ruchem, např.: vstupenky, mapy, plány, jízdní řady, tištěné průvodce a upomínkové předměty [13].

2.1.2 Cestovní kancelář

Cestovní kancelář je společnost, která organizuje a zabezpečuje zájezdy a prodává jednotlivé služby cestovního ruchu zákazníkům a cestovním agenturám. Mezi hlavní služby, které cestovní kanceláře nabízejí, patří doprava, ubytování, stravování, služby delegáta v místě pobytu, zdravotní pojištění a jiné. Tyto služby si cestovní kancelář zakoupí (nasmlouvá) přímo u poskytovatele (letecké společnosti, hotelu) většinou ještě před začátkem sezóny a sestaví z nich balíček zahrnující cestu do destinace a zpět, transfer z letiště na hotel a zase zpět, ubytování, stravování, služby delegáta a další doplňkové služby. Každá cestovní kancelář musí být v dnešní době pojištěna proti úpadku.



Obrázek č. 5: Subjekty na trhu cestovního ruchu

Zdroj: [10]

2.2 Popis interních procesů cestovní agentury

V této podkapitole se zaměřím na procesy, které se odehrávají uvnitř cestovní agentury a jsou spojeny se zpracováním objednávky od jejího založení, přes zaplacení a vyplnění cestovní smlouvy až po její dokončení.

2.2.1 Proces vytvoření objednávky

Zákazník má k objednání zájezdu u cestovní agentury několik možností. Tou první je standardní objednávka přes internetovou stránku, kde si zákazník vybere zájezd, který bude v souladu s jeho požadavky. Další způsob objednání je telefonický. U tohoto způsobu si zákazník zavolá většinou na bezplatnou linku cestovní agentury, kde telefon zvedne pracovníce na recepci a pokusí se zjistit od zákazníka co nejvíce informací. Může se stát, že zákazník má pouze představu, o jaký zájezd by měl zájem a v tom případě sdělí pracovníci pouze základní kritéria, kterých by se cestovní agentura měla držet při tvorbě nabídky pro tohoto zákazníka. Dalším způsobem jak si objednat dovolenou, je zasláním emailu. Tento email je opět zpracován pracovníci na recepci, která se z něj pokusí získat základní informace. U všech výše zmiňovaných způsobů objednání, se informace o klientovi a požadavky na jeho zájezd zapíší do karty zákazníka, které se přidělí ID a zařadí se do fronty ke zpracování. Pokud si zákazník nepřeje řešit svou dovolenou telefonicky ani přes internet a dává přednost osobnímu kontaktu, má možnost navštívit pobočku cestovní agentury, kde je postup obdobný.

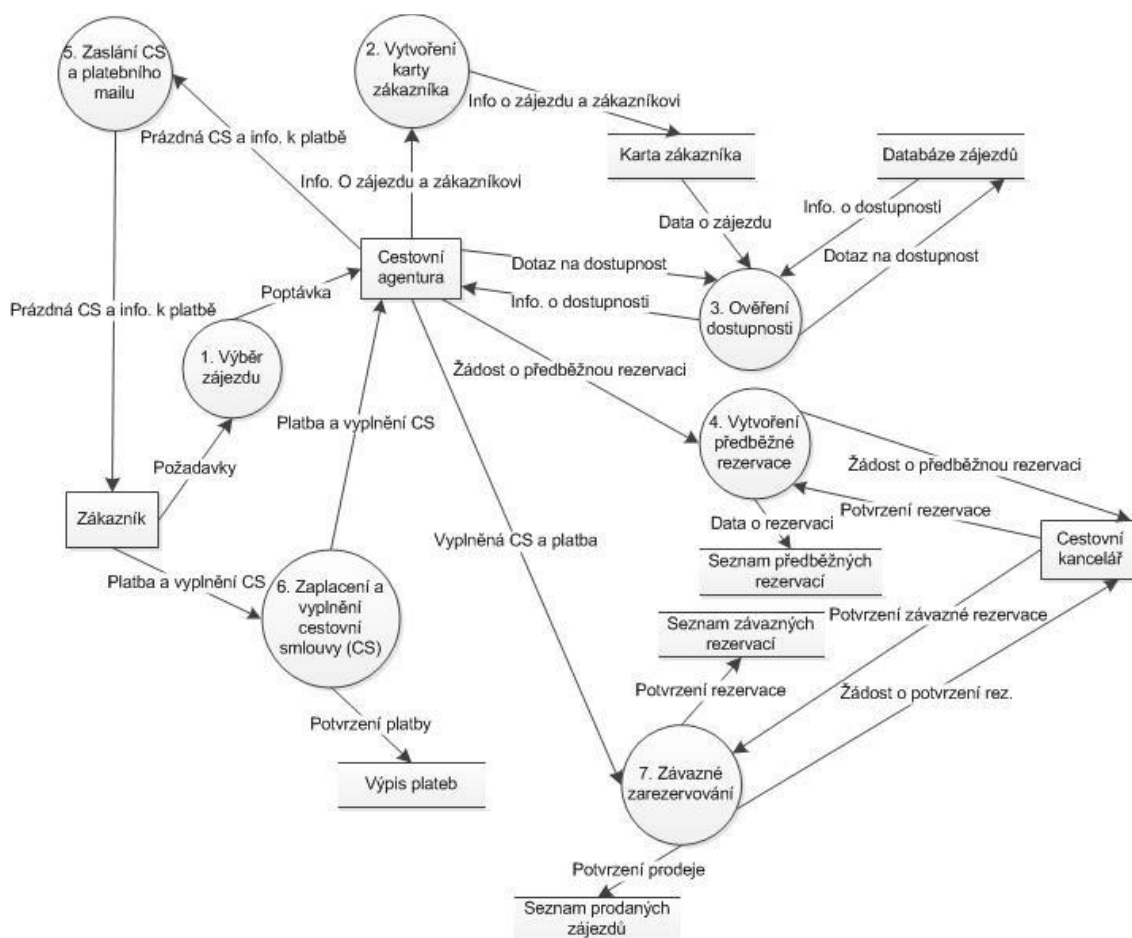
2.2.2 Proces zpracování objednávky

Vytvořená karta se zařadí do pomyslné fronty ke zpracování. Tato fronta se čistí od objednávek co nejrychleji to jde, aby v ní objednávky neležely a firma tak nepřicházela o zákazníky. Z fronty je karta náhodně přiřazena prodejci, který si zjistí, jestli se jedná o stálého nebo nového zákazníka. Pokud jde o zákazníka, který si u firmy už někdy zájezd zakoupil, je jeho objednávka přiřazena na prodejce, který s tím klientem řešil objednávku už dříve. Při zachování tohoto systému se udrží loajalita mezi zákazníkem a prodejcem a zákazník si tak vybuduje kladný vztah ke společnosti a bude se k ní i nadále vracet. Pokud je tedy vytvořená objednávka u správného prodejce, má tento prodejce za úkol kontaktovat zákazníka a ověřit u něj zda li jeho požadavky stále platí, zodpoví mu na dotazy, zjistí doplňující informace (např. jména, příjmení, adresu, data narození cestujících apod.) a zašle zákazníkovi několik nabídek. Po získání všech potřebných informací si prodejce vyhledá, která cestovní kancelář požadovaný zájezd pořádá a ověří si u ní dostupnost pro přesně určený počet osob a v termínu který klient požaduje.

Ověřování informací o dostupnosti, probíhá různými způsoby. Velké cestovní kanceláře k tomuto účelu mají rezervační systém, kde si prodejce vyhledá a ověří základní informace, které potřebuje (dostupnost, cenu, rozložení pokojů, atd.) a v případě nedostatku informací kontaktuje cestovní kancelář telefonicky. Bohužel ne všechny cestovní kanceláře mají funkční rezervační systém, na který se dá spolehnout. Existuje mnoho cestovních kanceláří, které informace o poskytovaných službách mají umístěné pouze na internetových stránkách a jsou zvyklé pouze na telefonní nebo emailovou komunikaci. Toto v hlavní sezóně prodeje zájezdů vede velice často k přetěžování těchto linek, zpomalení komunikace se zákazníkem a velmi často se vyskytují chyby způsobené lidským faktorem. Tím nejpomalejším způsobem k ověření těchto informací je emailová komunikace, na kterou jsou prodejci odkazováni při přetěžování telefonní komunikace.

2.2.3 Proces dokončení objednávky

Po získání informací o dostupnosti a úspěšné předběžné rezervaci ubytovacích a cestovních kapacit, kterou cestovní kanceláře dávají pouze na krátkou dobu v závislosti na datu začátku zájezdu (může se tedy jednat o hodinu nebo i tři dny) informuje prodejce zákazníka o době, kterou mu může poskytnout na vyplnění cestovní smlouvy a zaplacení částky za zájezd. Prodejce může ještě zákazníkovi nabídnout cestovní pojištění, parkování na letišti a další služby. Sdělí zákazníkovi platební informace a zašle mu cestovní smlouvu. Zákazník tuto smlouvu vyplní, převede peníze a na základě toho může prodejce podepsanou a vyplněnou cestovní smlouvu od zákazníka přeposlat do cestovní kanceláře, která po obdržení této smlouvy předběžnou rezervaci změní na závaznou. Posledním krokem tohoto procesu je informovat zákazníka o úspěšném dokončení jeho objednávky a zaslání cestovních pokynů, které budou obsahovat veškeré informace týkající se jeho zájezdu.



Obrázek č. 6: DFD Diagram procesu vyřízení objednávky

Zdroj: vlastní

2.3 Popis rezervačních a databázových systémů cestovních kanceláří a agentur

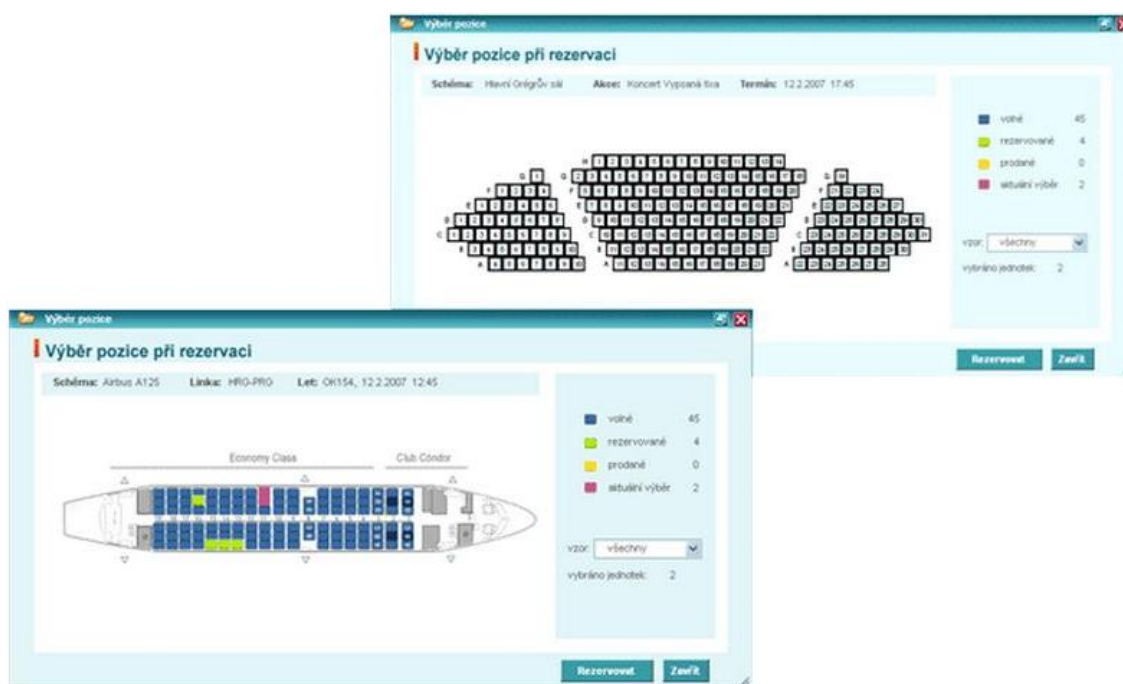
Tuto část bych chtěl věnovat krátkému představení několika společností a jejich produktů, které přicházejí s řešením, jenž se touto problematikou zabývá většinou v komplexnější rovině.

2.3.1 Magicware

Produktem společnosti magicware je rezervační systém pro cestovní kanceláře a agentury is-tour. Systém is-tour je dynamický díky rozdělení do několika základních nezávislých modulů, kterými jsou: evidence osob, evidence organizací, správa

uživatelů, role a soubory a obrázky. Ty obsahují různé agendy a nástroje pro práci s nimi. Systém se dělí na několik částí mezi něž patří:

- Rezervační systém
- Tvorba produktů - což se dá jinými slovy nazvat dynamické skládání balíčků.
- Výstupní sestavy, což jsou analýzy uplynulého prodeje, které slouží managementu firmy jako podklady pro důležité rozhodování týkající se budoucího modelu prodeje.
- Inventurní systém, který obsahuje evidence nakoupených kapacit, zboží a služeb. K těmto nakoupeným kontingentům jsou evidovány další podmínky nákupu a cena. Jednotlivým umístěním je možné přiřadit grafické schéma umístění jednotek v zařízení jako např. sedadla v letadlech, plány sedadel v sálech a plány podlaží v hotelech [11].



Obrázek č. 7: Grafické schéma umístění jednotek v zařízení

Zdroj: [11]

2.3.2 Open Travel Network

Open Travel Network je systém poskytující nástroje pro výstavbu webových stránek, nástroje pro jejich další optimalizaci a nástroje pro komunikaci s partnery a zákazníky. Avšak základem jejich systému je databáze zájezdů používaná 250 provizními prodejci, do které se importují data ve formátu XML.

Produkty této společnosti jsou rozděleny do tří kategorií. WEB plus, WEB Business a XML export. WEB plus je produkt, který stojí 390 Kč měsíčně bez DPH a obsahuje webové stránky pro cestovní agentury s online katalogem zájezdů. Dalším produktem, který tato firma nabízí je WEB Business za cenu 990 Kč měsíčně bez DPH, což je web pro cestovní kanceláře. XML export má za úkol vyexportovat zájezdy prodejcům a na různé portály v jednotném formátu za cenu od 250 Kč měsíčně za množství do 50 zájezdů a až za 1500 Kč měsíčně za množství 2000 zájezdů.

Do databáze se data importují v pravidelných intervalech minimálně jednou za den. V sezóně tato databáze obsahuje téměř dva miliony odjezdových termínů [14].

2.3.3 Stovka software

Stovka software je společnost, která se specializuje na spolupráci se středními a menšími cestovními kancelářemi. Jejich snahou je propojení informačních systémů, internetového marketingu a prodeje, firemní strategii a vizi do jednoho celku, kterému říkají Stovka vision. Stovka vision je další komplexní řešení nabízené na našem trhu a jeho funkce jsou rozděleny na:

- Rezervační systém, jenž má netradiční funkci rozlišovat dvojí charakter u rezervace pokojů. Jsou to pokoje kontingentní, které mají cestovní kanceláře nasmlouvané a pokoje na vyžádání, u kterých se musí prověřovat dostupnost přímo u majitele.
- V ekonomickém systému se podařilo zachytit problematiku faktur, ovšem se nejedná o plnohodnotný účetní software. Na tento systém se však standardně dá připojit ekonomický systém pohoda.
- CRM – podpora prodeje zahrnuje databázi klientů a účastníků zájezdu. Dále se evidují obchodní partneři, prodejci a dodavatelé. Další možností je hromadné

rozesílání nabídek přes e-mail a export dat z rezervačních systémů ve formátu XML pro další internetové prodejce.

- Dalším modulem, který firma nabízí, jsou WWW stránky + online objednávky. Ty prošly v roce 2010 optimalizací SEO. Součástí je i jednoduchý redakční systém.

Ceny těchto produktů se dělí do jednotlivých cenových programů web, standart, profi a vision [12].

Tab. 5: Základní cenové programy

WEB	STANDART	PROFI	VISION
XML report a webové stránky s kalkulací a objednávkou	Běžná agenda cestovní kanceláře včetně plánování dopravy a fakturace.	Napojení na účetní systémy, modul pro rozesílání newsletterů, stop boxing a online objednávky.	Analýza procesu firmy a návrh řešení s důrazem na efektivitu dané cestovní kanceláře.
1900 Kč za měsíc	3 900 Kč za měsíc	5 900 Kč za měsíc	individuálně

Zdroj: [12]

2.3.4 CeSYS

CeSYS je dalším řešením pro cestovní kanceláře a agentury obsahující zájezdy nejen českých ale i německých cestovních kanceláří a jejich vyhledávačů. CeSYS poskytuje webové stránky s redakčním systémem SEO optimalizací a hotline. Celý systém je rozdělen na dvě hlavní části na webové stránky a administraci.

Modul webové stránky má několik funkcí. Mezi ně se řadí vyhledávání zájezdů, poskytování informací o těchto zájezdech a objednání zájezdů.

Do modulu administrace byl zahrnut redakční systém, který slouží k práci s přijatými poptávkami a tvorbě vlastních nabídek. Další částí tohoto modulu je funkce rezervace a smlouvy, která eviduje obchodní případy od evidence klientů až po kontaktní údaje cestovních kanceláří. Funkce uživatelé, obsahuje správu uživatelů, kteří do tohoto systému mají přístup.

Ceník jednotlivých verzí CeSYSu se skládá z pěti verzí:

- CeSYS Vyhledávač za cenu 499 Kč měsíčně bez poplatku za nasazení
- Mini za cenu 499 Kč měsíčně s pořízovací cenou 4 999 Kč
- Standart má cenu za měsíční licenci 799 Kč a poplatkem za nasazení 4 999 Kč
- Profi s měsíční cenou 1 499 Kč a poplatkem za nasazení 4 999 Kč
- Frame s licenci za 799 Kč měsíčně a nasazením za 4 999 Kč [15].

2.3.5 Pear

Firma Pear s.r.o. dodává rezervační systémy pro cestovní kanceláře. Tyto systémy zahrnují problematiku dispečinku, call centra, tour operátora a poboček provizních prodejců. Mezi produkty nabízené touto společností patří:

- Rezervační terminál, což je windows aplikace, která zahrnuje kroky vyřízení zákazníka od objednávky až po fakturaci.
- WinCK poskytuje kalkulační model založený na definovaných smlouvách, jehož výstupem jsou koncové ceny.
- Datová příprava evidující správu adres, země, zařízení, pojištění, trasy a nástupní místa.
- Redakční systém, který shromažďuje informace o hotelech a destinacích. Tyto informace slouží následně jako podklady pro webové stránky.
- Modul CRM sloužící k obstarání informací o zákaznících, kteří mají v systému svou historii smluv a rezervačních čísel.
- Statis je modul zahrnující statistiky rezervací, zákazníků, roamingů, odjezdových řádů a odletů.
- A v neposlední řadě je v rezervačním systému obsažen ještě modul účetnictví, pokrývající účetní deník, hlavní knihu, generované sestavy, analýzy nákladů a prvotní doklady [16].

2.4 Analýza problému v praxi

V této podkapitole bych se chtěl zaměřit na několik nedostatků, které výše zmiňované systémy obsahují. Některým z nich by se dalo předejít vytvořením jednotné databáze, která by obsahovala aktuální informace o dostupnosti a možnosti rezervace služeb v cestovním ruchu.

2.4.1 Komplikace při evidování nestandardních situací

Pokud se prodejce dostane do situace, kdy si zákazník poptá dvě různé služby (např. samostatný let a jiný zájezd) od různých cestovních kanceláří, nebylo možné tuto objednávku zaznamenat do systému a museli se kvůli tomu vytvářet a vyplňovat dvě různé karty. Tyto karty se potom museli všechny dohledat a přiřadit na stejného prodejce. Tímto způsobem se dostaneme občas do situace, kdy objednávku jednoho zákazníka řeší dva prodejci.

2.4.2 Špatná informovanost

Další situace, která se může naskytnout, je nedostatečná nebo špatná informovanost cestovních kanceláří o stavu obsazení ubytovacích kapacit. To může být způsobeno chybami v systému nebo nepozorností pracovníků ale i neevidováním všech informací. Některé cestovní kanceláře informace o dostupnosti a o ubytování samotném zjišťují přímo v destinaci. Proto se může stát, že se čeká na odpověď z destinace i několik hodin v horších případech i dnů. Je jasné, že tomuto způsobu se zkrátka někdy nevyhneme, ale určitě by se práce usnadnila, kdyby data o aktuální obsazenosti byla načítána přímo z databáze hotelu včetně všech informací, které hotel o daném ubytování může poskytnout. Momentálně se v některých systémech situace řeší pouze záznamem o typu pokoje a obecnými informacemi o hotelu. Zákazník tak například neví, v jakém podlaží bude ubytován, jak je pokoj vybaven nebo v jaké části hotelu se nachází.

2.4.3 Problém s rezervacemi

Při vytváření předběžných rezervací v různých rezervačních systémech se chybou prodejce může stát, že vytvořenou rezervaci nevloží do databáze cestovní agentury (nebo ji vloží špatně vyplněnou). Tím se dostáváme do situace, kdy je v cestovní kanceláři vytvořena rezervace, která nemá žádnou vazbu na zákazníka nebo v horším případě na prodejce. Takovou rezervaci potom není možné dohledat a v případě že se jedná o poslední místa, tím omezuje možnost zarezervovat zájezd jiným prodejcům, kteří mají zákazníky, kteří by o tuto rezervaci měli zájem. Problém s vazbou mezi rezervací a prodejcem tedy pocítují především cestovní kanceláře, kterým v těchto případech zůstane v systému zarezervován zájezd bez možnosti dohledání na prodejce. Řešením pro tuto situaci by bylo evidovat jednotlivě všechny pracovníky společnosti a

pro každého vytvořit účet. Tento účet by se pak vázal automaticky na každou rezervaci, kterou pracovník vytvoří.

3 Vlastní návrh řešení

Po provedení analýzy současného stavu a shrnutí základních nedostatků můžu pokračit a přejít k samotnému návrhu databáze. Tímto návrhem bych se chtěl pokusit alespoň z části předejít výše zmiňovaným problémům.

3.1 Definice požadavků a procesů

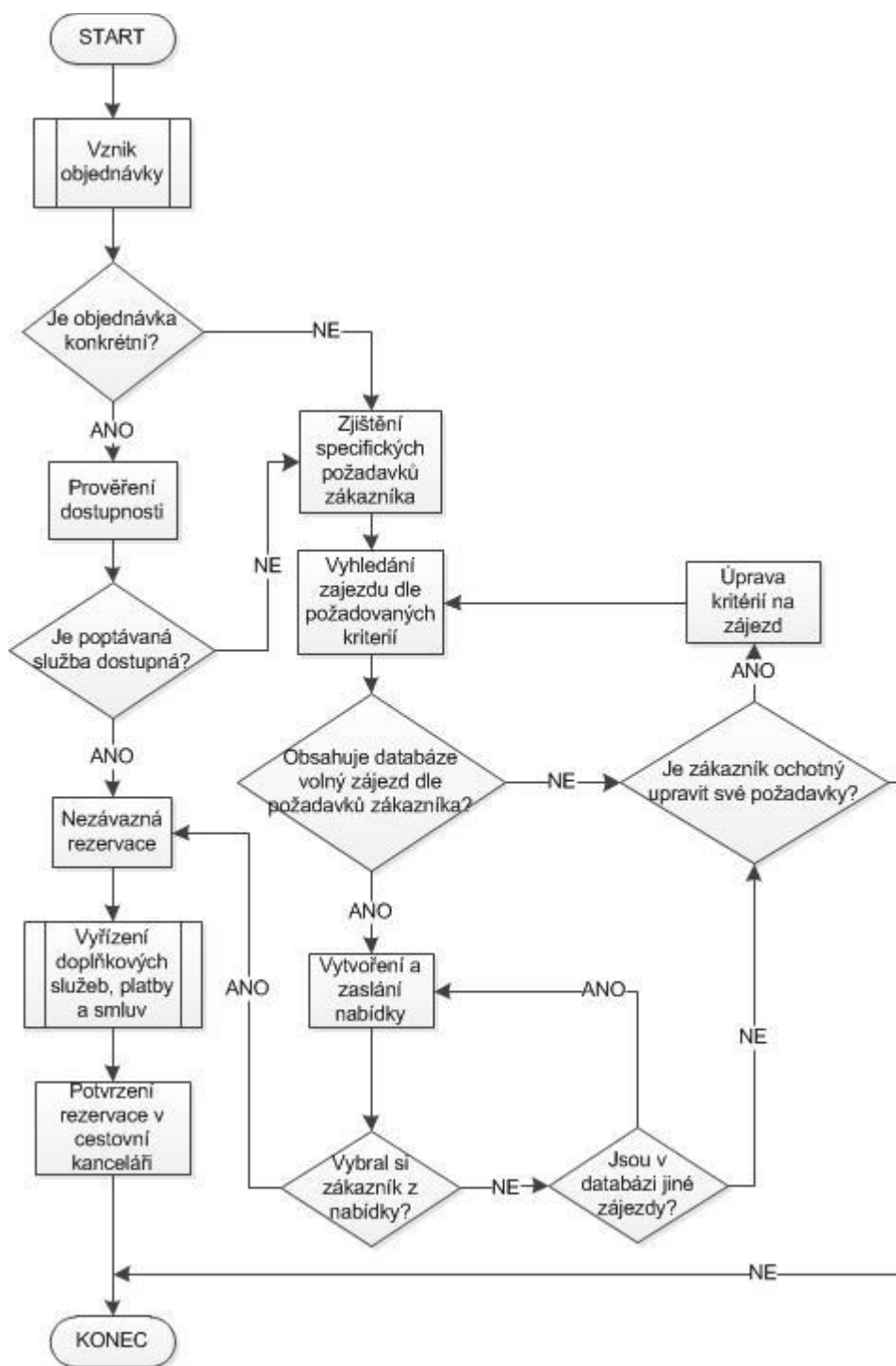
Tuto část bych chtěl věnovat popisu jednotlivým procesům rozděleným na hlavní procesy, které slouží k běžnému chodu společnosti a subprocessy.

Databáze by zahrnovala sepisování nových objednávek, informace o ubytovacích kapacitách, záznamy o zákaznících, všech objednávkách, prodejcích, tvorbu rezervací nejen ubytovacích ale i přepravních kapacit a další procesy sloužící ke každodennímu chodu společnosti.

3.1.1 Běžný chod společnosti – hlavní procesy

Mezi hlavní procesy společnosti jsem se snažil zachytit ty procesy, které se používají při běžném prodeji zájezdů. To zahrnuje zjištění požadavků zákazníka, prověření dostupnosti, vytvoření předběžné rezervace, tvorba nabídky, dokončení objednávky aj.

Mimo to si zákazník může pořídit doplňkové služby. Což nám tento seznam může rozšířit o subprocessy: sjednání cestovního pojištění, rezervaci parkování na letišti nebo přidělení objednávky prodejci což probíhá podle určitých pravidel. U zákazníka se tak po dobu zpracování jeho objednávky mění stav této objednávky, podle toho v jaké fázi se zrovna nachází. Stejný postup se aplikuje i při prodeji samostatné letenky nebo samostatného ubytování.

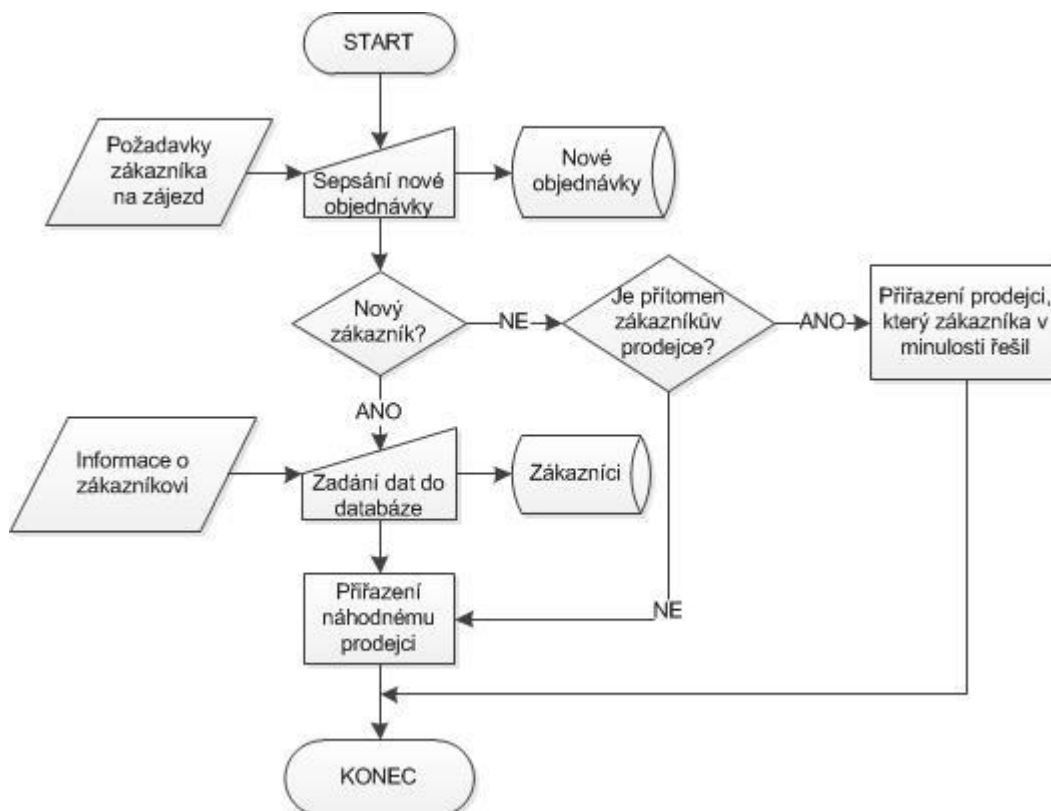


Obrázek č. 8: Vývojový diagram – běžný chod společnosti

Zdroj: vlastní

3.1.2 Vznik objednávky

Subproces vzniku objednávky je založen na několika pravidlech, které rozhodují o tom, kdo bude vzniklou objednávkou řešit. Po uložení nově vzniklé objednávky do databáze, se rozhoduje o tom, jestli se jedná o nového zákazníka či nikoli. Je-li zákazník nový, přidělí se jeho objednávka náhodnému prodejci. Pokud ovšem zákazník už v naší databázi je, je jeho objednávka přidělena prodejci, který zákazníka řešil jako poslední a zároveň je přítomen.

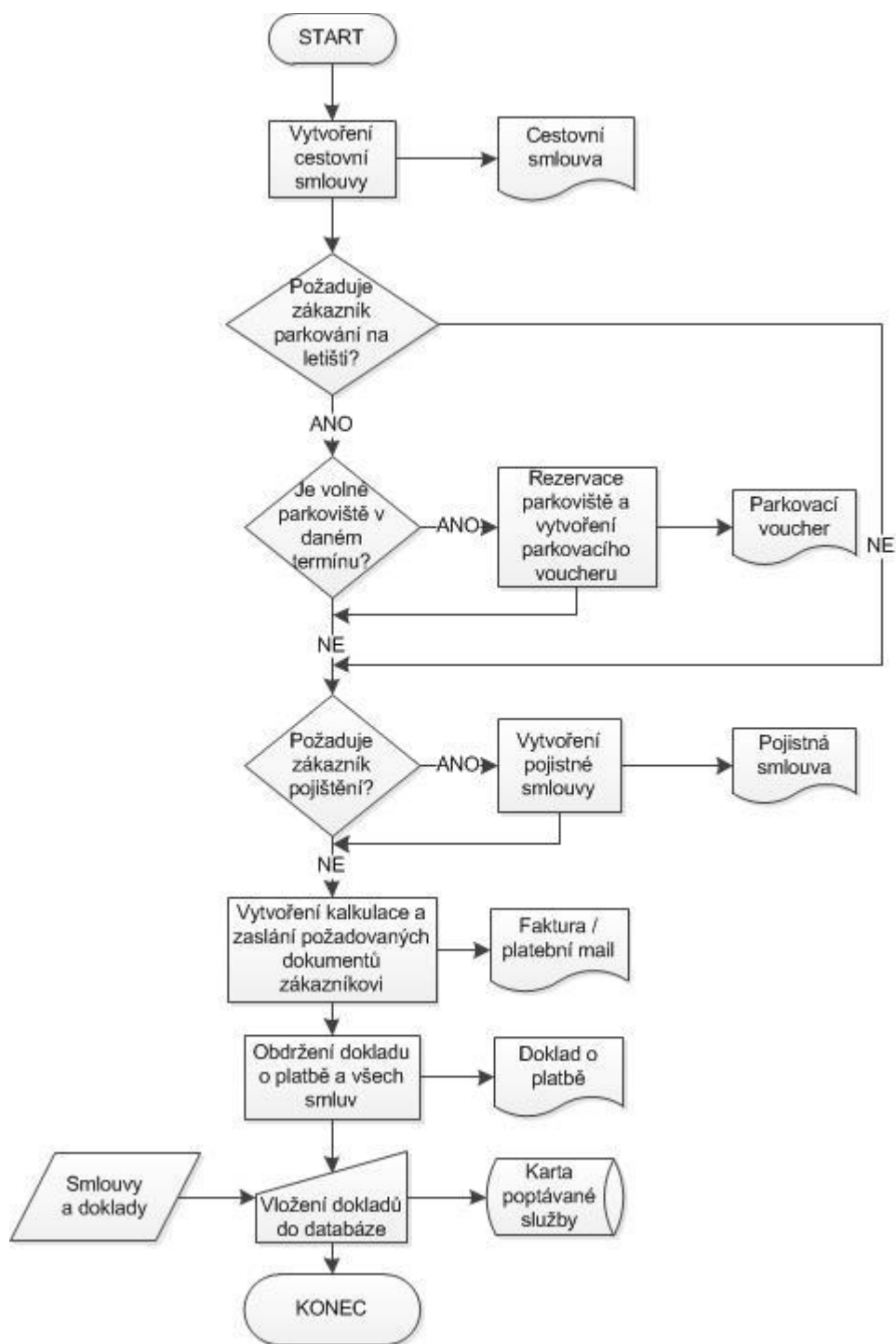


Obrázek č. 9: Vývojový diagram – vznik objednávky

Zdroj: vlastní

3.1.3 Vyřízení doplňkových služeb, platby a smluv

Tento subproces je založen na zjištění, co zákazník bude ke své objednávce požadovat za doplňkové služby, které bychom mu mohli nabídnout. Jedná se konkrétně o parkování na letišti, ze kterého odlétá, k čemuž patří vygenerování parkovacího voucheru. Další nabízenou službou je komplexní pojištění. To je samozřejmě rozděleno na jednotlivé sazby, podle toho o jaký druh pojištění je zájem. K pojištění se generuje pojistná smlouva obsahující všechny náležitosti. K těmto službám se generuje cestovní smlouva a platební mail, kde jsou uvedeny všechny informace, které zákazník potřebuje k vyplnění dokumentu a zaslání zpět spolu s platbou. Zákazník k tomu všemu zašle doklad o provedené platbě, na základě čehož se cestovní smlouva přepošle do cestovní kanceláře a zájezd se tím závazně zarezervuje. Všechny dokumenty se ukládají do karty zákazníka, aby bylo možné je zpětně kdykoli dohledat.



Obrázek č. 10: Vývojový diagram – vyřízení doplňkových služeb, platby a smluv

Zdroj: vlastní

3.2 Konceptuální návrh

Konceptuální návrh je prvním ze tří kroků vedoucích k celkovému návrhu databáze. V této části identifikujeme důležité entity a relace, které je třeba reprezentovat v databázi. Posledním krokem konceptuálního návrhu je vytvoření ER diagramu, který nám poslouží při vytváření logického návrhu.

3.2.1 Identifikace entit

Identifikace entit je podkapitola, ve které se identifikují základní entity databáze. Nesmíme zapomenout, že nejde o všechny entity ale jen o pouhý návrh objektů, protože konceptuální návrh databáze nezohledňuje dekompozici ani normalizaci.

Tab. 6: Seznam entit

Entita	Popis entity
cestovní kancelář	Společnost prodávající zájezdy přímo od poskytovatele služeb
cestovní agentura	Společnost prodávající zájezdy od cestovních kanceláří
dopravní společnost	Společnost poskytující dopravu cestovním kancelářím
prodejce	Osoba prodávající zájezdy v cestovní agentuře
zákazník	Osoba kupující zájezdy v cestovní agentuře
objednávka	Souhrn všech zájezdů a dílčích služeb poptaných zákazníkem
rezervace	Nezávazné zamluvení poptávané služby prodejcem na určitou dobu
země	Geografické území (stát)
lokalita	Část státu
ubytování	Hotel, apartmán nebo jiné ubytovací zařízení
let	Služba zajišťující přepravu cestujících
letišť	Místo odletu nebo příletu
pokoj	Ubytovací jednotka v hotelu nebo jiném ubytovacím zařízení
parkování	Doplňková služba zajišťující parkovací místo na letišti
zájezd	Balíček služeb připravený cestovní kanceláří zajišťující dopravu, ubytování a další služby
cestující	Osoby, které budou využívat zakoupených služeb

typ pojištění	Typ zdravotního pojištění pro cestující
---------------	---

Zdroj: vlastní

3.2.2 Identifikace relací

V této podkapitole si určíme vztahy mezi entitami, což nám později pomůže s dekompozicí tabulek.

Tab. 7: Identifikace relací mezi entitami

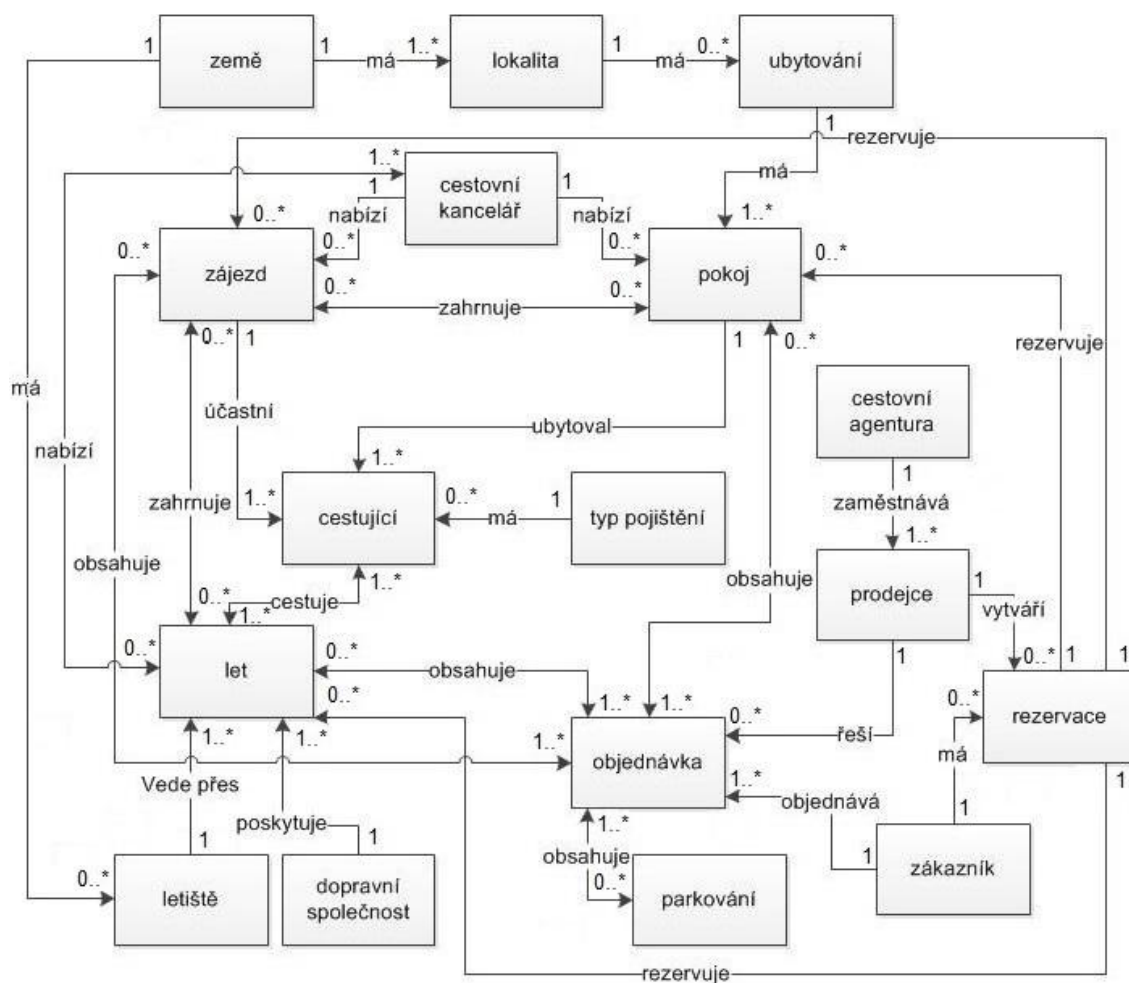
Entity	Typ relace	Popis
prodejce - objednávka	1:N	Prodejce může řešit N objednávek a objednávku řeší pouze jeden prodejce
objednávka - zájezd	M:N	Objednávka může obsahovat N zájezdů a zájezd může být obsažen v M objednávkách
země - lokalita	1:N	Země může mít N lokalit a lokalita patří pouze do jedné země
zákazník - objednávka	1:N	Zákazník si může objednat N objednávek a objednávku si může objednat pouze jeden zákazník
cestovní kancelář – zájezd	1:N	Cestovní kancelář nabízí N zájezdů a zájezd je nabízen pouze jednou cestovní kanceláří
cestovní kancelář – pokoj	1:N	Cestovní kancelář nabízí N pokojů a pokoj je nabízen pouze jednou cestovní kanceláří
cestovní kancelář - let	M:N	Cestovní kancelář nabízí N letů a let je nabízen M cestovními kancelářemi
cestovní agentura - prodejce	1:N	Cestovní agentura zaměstnává N prodejců a prodejce je zaměstnán pouze v jedné cestovní agentuře
zájezd - cestující	1:N	Zájezdu se může účastnit N cestujících a jeden cestující se účastní pouze jednoho zájezdu
let - cestující	M:N	Jedním letem může cestovat N cestujících a jeden cestující může cestovat M lety
pokoj - cestující	1:N	Pokoj je pro N cestujících a jeden cestující má pouze jeden pokoj
lokalita - ubytování	1:N	V jedné lokalitě se nachází N ubytování a jedno ubytování je pouze v jedné lokalitě
prodejce - rezervace	1:N	Prodejce může vytvořit N rezervací a rezervaci může vytvořit pouze jeden prodejce
zákazník - rezervace	1:N	Zákazník může mít N rezervací a rezervace musí mít jednoho zákazníka
ubytování - pokoj	1:N	Ubytování má N pokojů a jeden pokoj je pouze v jednom ubytování
objednávka - parkování	M:N	V objednávce může být N parkování a parkování může být v M objednávkách
let - letiště	1:N	jeden let vede z jednoho letiště a přes jedno letiště vede N letů

typ pojištění - cestující	1:N	Jeden typ pojištění může mít N cestujících a jeden cestující má jen jeden typ pojištění
dopravní společnost - let	1:N	Dopravní společnost poskytuje N letů a let může být poskytován jen jednou dopravní společností
zájezd - pokoj	M:N	Zájezd může zahrnovat N pokojů a pokoj může být zahrnut v M zájezdech
zájezd - let	M:N	Zájezd může zahrnovat N letů a let může obsahovat M zájezdů
objednávka - let	M:N	Objednávka může obsahovat N letů a let může být obsažen v M objednávkách
objednávka - pokoj	M:N	Objednávka může obsahovat N pokojů a pokoj může být obsažen v M objednávkách
rezervace - let	1:N	Pod jednou rezervací může být zarezervováno N letů a jeden let musí být zarezervován pouze pod jednou rezervací
rezervace - pokoj	1:1	Pod jednou rezervací může být zarezervováno N pokojů a pokoj musí být zarezervován pouze pod jednou rezervací
rezervace - zájezd	1:N	Pod jednou rezervací může být zarezervováno N zájezdů a zájezd musí být zarezervován pouze pod jednou rezervací
země - letiště	1:N	Jedna země může mít N letišť a letiště patří pouze do jedné země

Zdroj: vlastní

3.2.3 ER Diagram

V tomto kroku jsme vytvořili ER diagram, což nám usnadňuje reprezentovat entity a jejich vztahy k ostatním entitám a pomůže nám při tvorbě logického návrhu.



Obrázek č. 11: ER Diagram

Zdroj: vlastní

3.3 Logický návrh

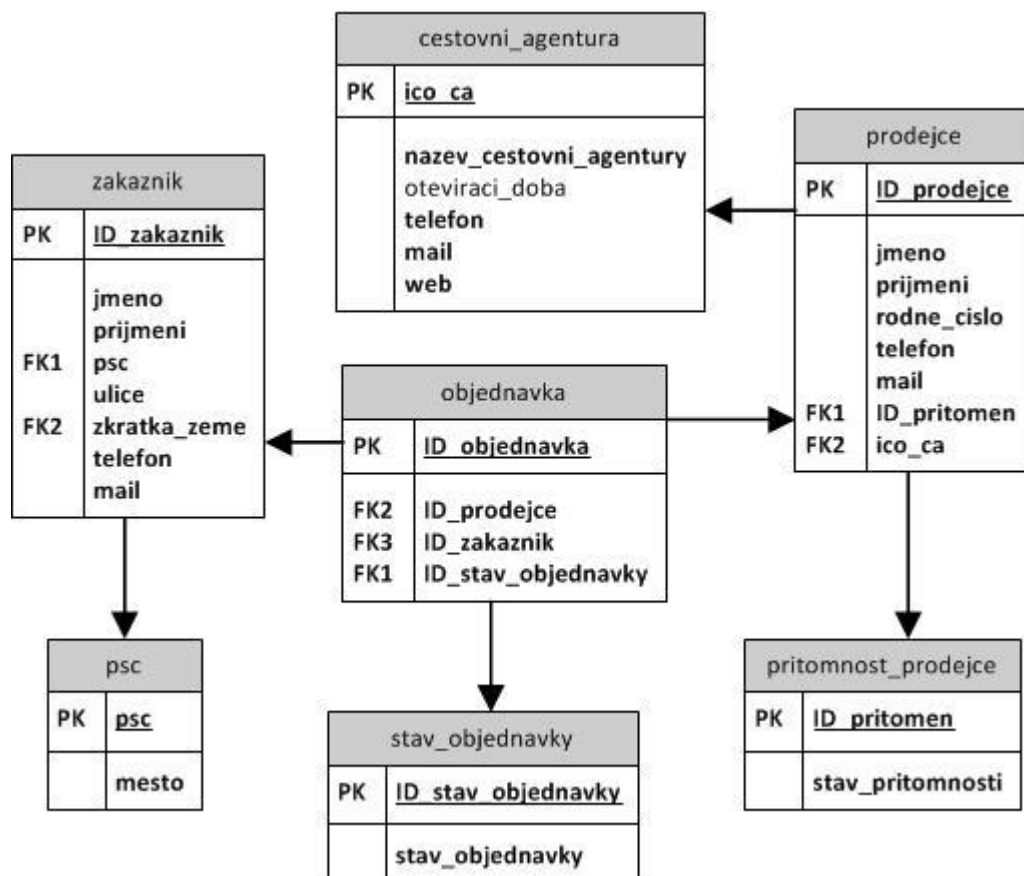
Logický návrh slouží k vytvoření tabulek, ověření integrity a normalizace. Tentokrát jsou v něm obsaženy všechny tabulky, jejich atributy, primární a cizí klíče a relace mezi nimi.

3.3.1 Zákazník, prodejce a objednávka

V tabulce *zakaznik* je primárním klíčem identifikační číslo zákazníka. V tabulce ho reprezentuje atribut *ID_zakaznik*. Tabulka obsahuje základní údaje o zákazníkovi, tedy především jeho adresu a kontaktní údaje.

Tabulka *prodejce* využívá jako primární klíč identifikační číslo prodejce a jsou v ní uvedeny kontaktní údaje na prodejce cestovní agentury. Jde především o pracovní telefon a mail, který se využívá nejvíc, dále tabulka obsahuje údaje o tom, jestli je prodejce přítomen v práci a pro jakou cestovní agenturu pracuje.

Tabulka *objednavka* je určena jedinečným identifikátorem ID_objednavka a eviduje všechny objednávky v cestovní agentuře, který zákazník si danou objednávku poptal, který prodejce danou objednávku vyřizuje a v jakém stavu se objednávka nachází.



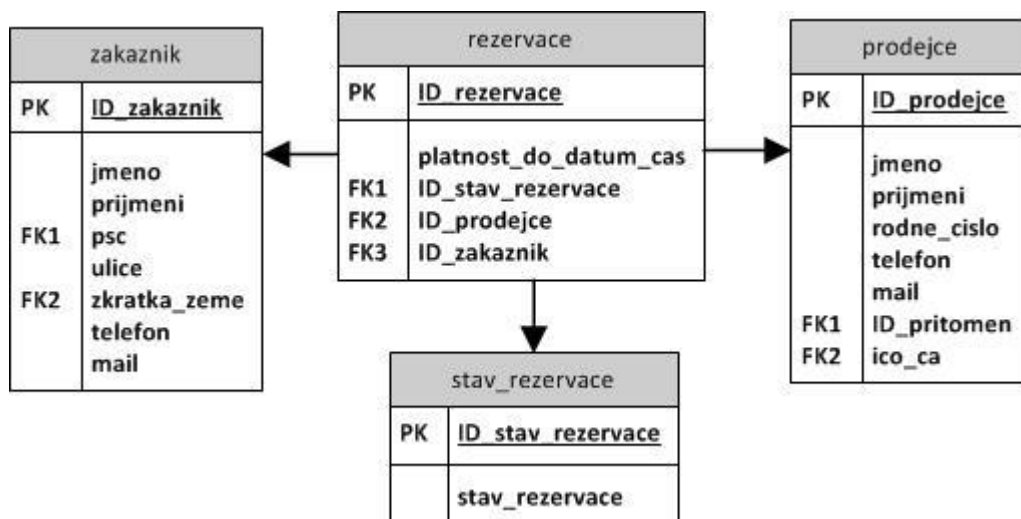
Obrázek č. 12: Zákazník, prodejce a objednávka

Zdroj: vlastní

3.3.2 Rezervace

Tabulka *rezervace* používá jako primární klíč identifikační číslo rezervace neboli rezervační číslo. Každá rezervace je platná jen po určitou dobu a proto je v tabulce atribut určující dobu expirace každé rezervace. V tabulce se kromě toho ještě eviduje zákazník, pro kterého byla rezervace vytvořena, v jakém stavu se rezervace zrovna

nachází a prodejce, který rezervaci vytvořil, aby bylo vždy možné dohledat, kdo jakou službu blokuje.



Obrázek č. 13: Rezervace

Zdroj: vlastní

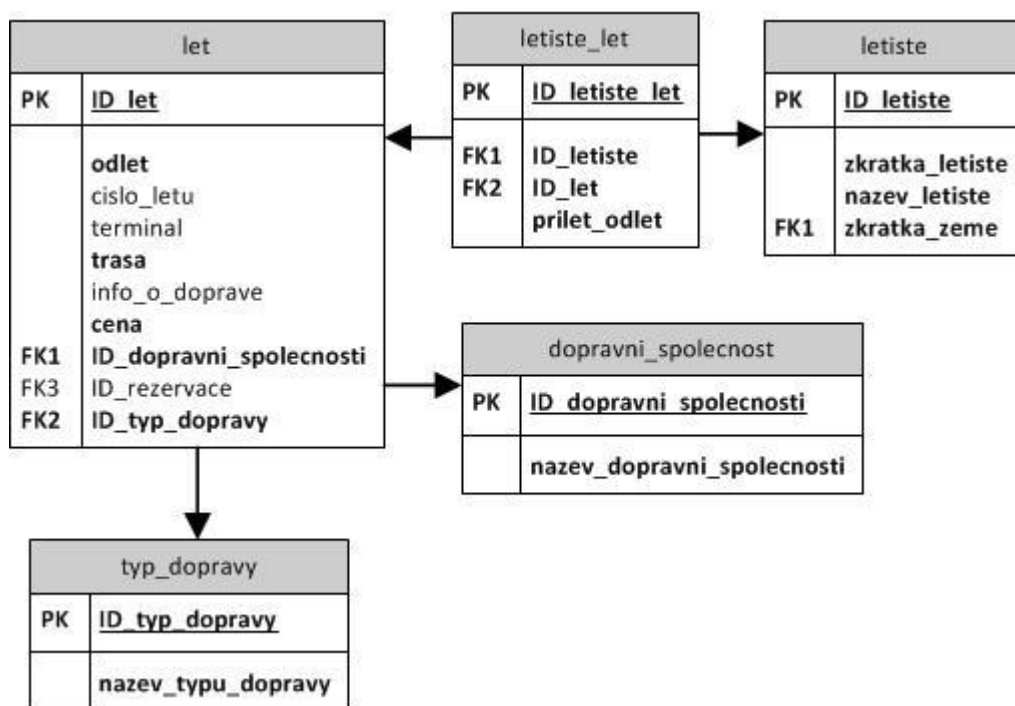
3.3.3 Let, letiště a dopravní společnost

V tabulce *letiste* je určen primární klíč identifikačním číslem letiště a hlavní účel této tabulky je identifikovat letiště, z kterého nebo na které daný let letí. Proto tabulka obsahuje pouze základní informace o letišti.

Jelikož zákazníci mají často dotaz, s jakou leteckou společností na dovolenou poletí, rozhodl jsem se, vytvořit tabulku *dopravni_spolecnost*, která nám bude dávat informace která doprava je kým organizována.

Tabulka *let* obsahuje informace o službě zajišťující dopravu. Její primární klíč je určen identifikačním číslem letu, který je automaticky generován. Mezi atributy této tabulky je zahrnuto o jaký typ dopravy se jedná, dobu odletu, jaká dopravní společnost tuto službu poskytuje a jestli byl let už někým rezervován.

Dekompoziční tabulka *letiste_let* shromažďuje informace o tom, který let odlétá z kterého letiště a kam přilétá.



Obrázek č. 14: Let, letiště a dopravní společnost

Zdroj: vlastní

3.3.4 Země, lokalita, ubytování a pokoj

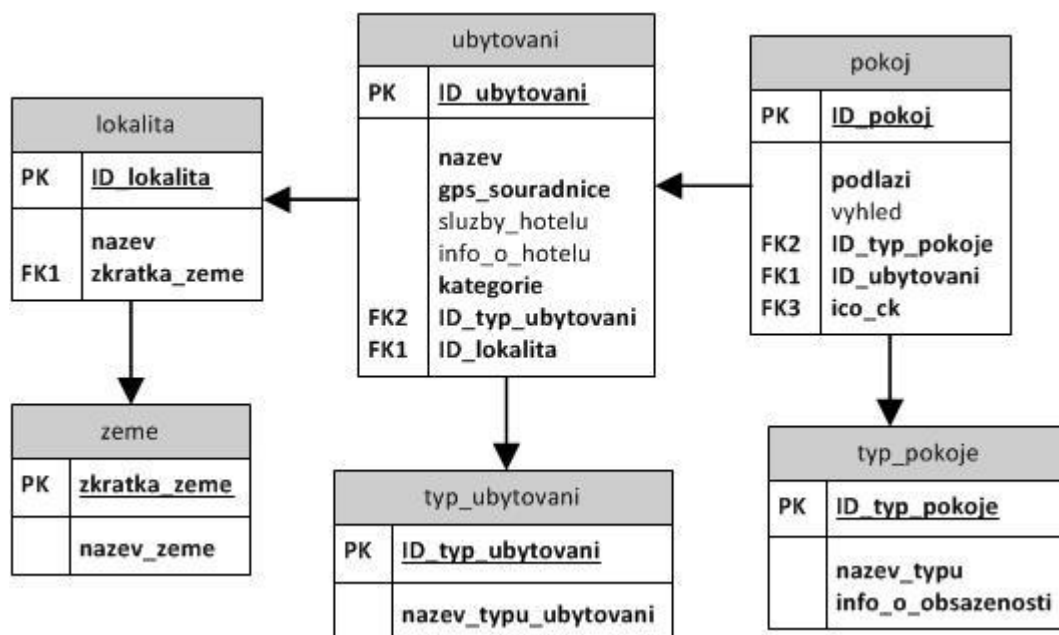
V tabulce *zeme* je primární klíč určen třímístnou zkratkou země. Kromě toho tabulka ještě obsahuje název země.

V tabulce *lokalita* je jako primární klíč použito identifikační číslo a mezi její atributy patří název lokality a země, v které se lokalita nachází.

Tabulka *ubytovani* nám dává informace o hotelu nebo jiném ubytovacím zařízení. Tyto informace spočívají v identifikačním čísle hotelu, které zároveň slouží jako primární klíč, názvu ubytování a GPS souřadnicích, jenž udávají přesnou polohu hotelu. Dále se z této tabulky můžeme dozvědět informace o hotelových službách, kolika hvězdami byl hotel ohodnocen, o jaký typ ubytování se přesně jedná a ve které lokalitě se ubytování nachází.

Na tabulku ubytování se váže tabulka *pokoj*. Jako primární klíč jsem v této tabulce zvolil identifikační číslo pokoje a uvedl v ní co nejvíce informací týkající se této ubytovací jednotky. Mezi nimi je zahrnuto, na kterém podlaží se ubytovací jednotka

nachází, jaký výhled pokoj nabízí, o jaký typ pokoje jde, v jakém hotelu se jednotka nachází a jaká cestovní kancelář ji nabízí.



Obrázek č. 15: Země, lokalita, ubytování a pokoj

Zdroj: vlastní

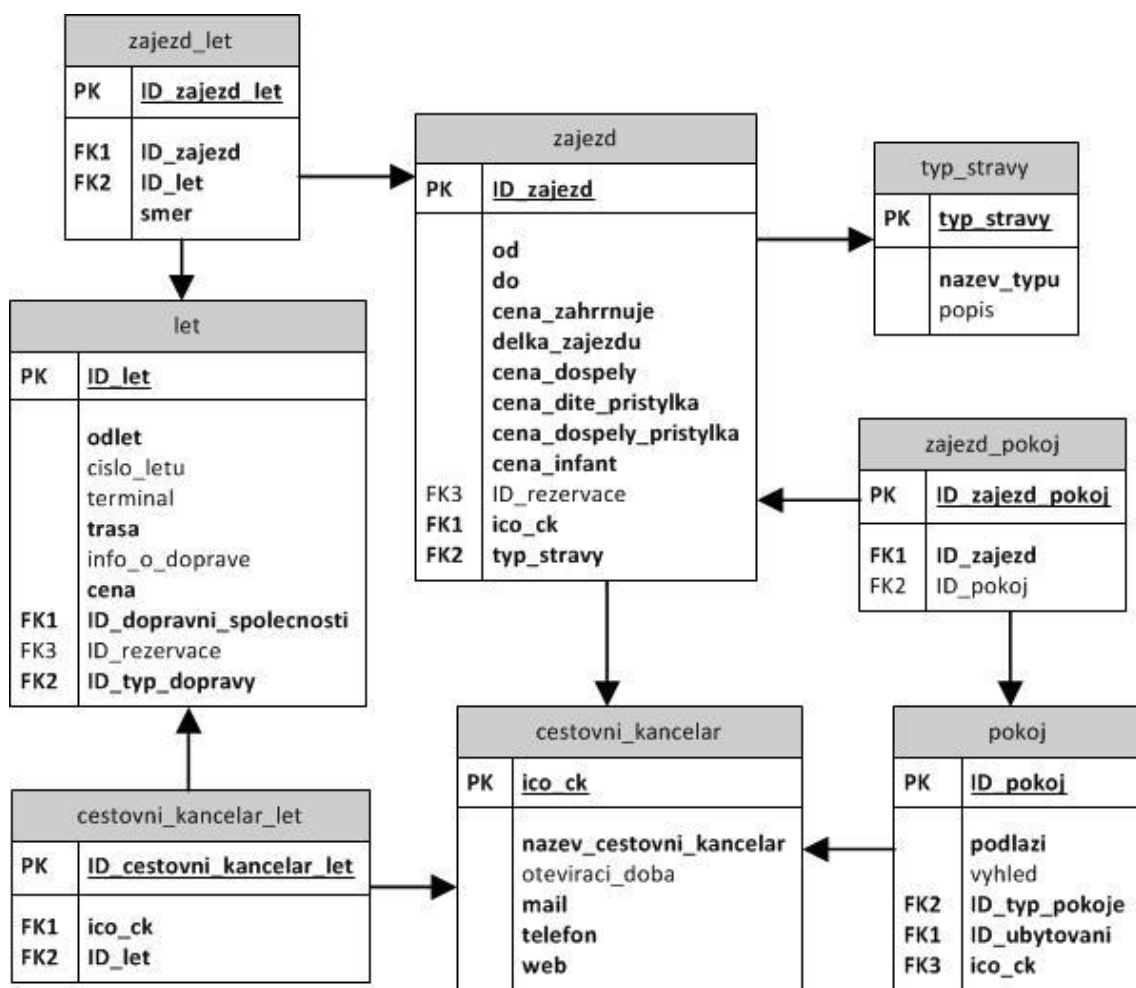
3.3.5 Cestovní kancelář, zájezd, pokoj a let

Cestovni_kancelar je tabulka evidující všechny cestovní kanceláře, s kterými cestovní agentura spolupracuje. V této tabulce tvoří primární klíč IČO společnosti a obsahuje základní údaje, které prodejce při prodeji zájezdů využije. Jedná se o kontakt na cestovní kancelář, otevírací dobu a web společnosti.

Tabulka cestovní kanceláře je napojena na tabulku *zajezd*, *pokoj* a *cestovni_kancelar LET*. První zmiňovanou tabulkou *zajezd* je myšlen balíček služeb. Proto, tato tabulka eviduje informace týkající se termínu zájezdu, jeho cenu, co všechno je v něm zahrnuto a jaká cestovní kancelář ho nabízí. Jako primární klíč jsem v této tabulce zvolil identifikační číslo zájezdu.

Jelikož mezi tabulkou *zajezd* a *pokoj* je vazba M:N musel jsem mezi ně vložit tabulku *zajezd_pokoj*. Tato tabulka nám říká, který pokoj je zahrnut v kterém zájezdu, jelikož jeden pokoj bývá využíván pro více zájezdů.

V tabulce *cestovni_kancelar лет* se ukládají informace o tom, která cestovní kancelář nabízí který let. Jelikož místa v jednom letadle nabízí více cestovních kancelářů, musí se tato situace řešit přes samostatnou tabulku.



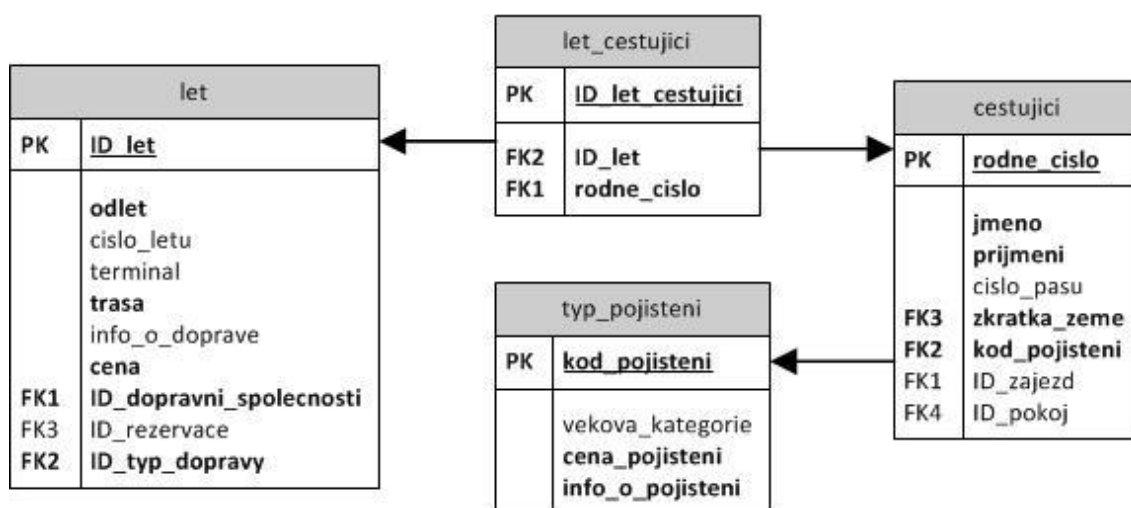
Zdroj: vlastní

3.3.6 Cestující, let a typ pojištění

V tabulce *cestující* tvoří primární klíč atribut *rodne_cislo*. Kromě toho se v tabulce zaznamenávají údaje o cestujícím, kterého zájezdu se účastní, jaký pokoj si pronajal a jaký typ cestovního pojištění si zvolil.

Tabulka *cestujici* je spojena s tabulkou *let* pomocí dekompoziční tabulky *cestujici_let*. Tato tabulka nám dává informace o tom, jakou dopravou jel jaký cestující.

V tabulce *typ_pojisteni* jsou obsaženy informace o jednotlivých typech pojištění a jejich ceně. Tato tabulka je propojena s tabulkou *cestujici* pomocí primárního klíče *kod_pojisteni*.



Obrázek č. 17: Cestující, let a typ pojištění

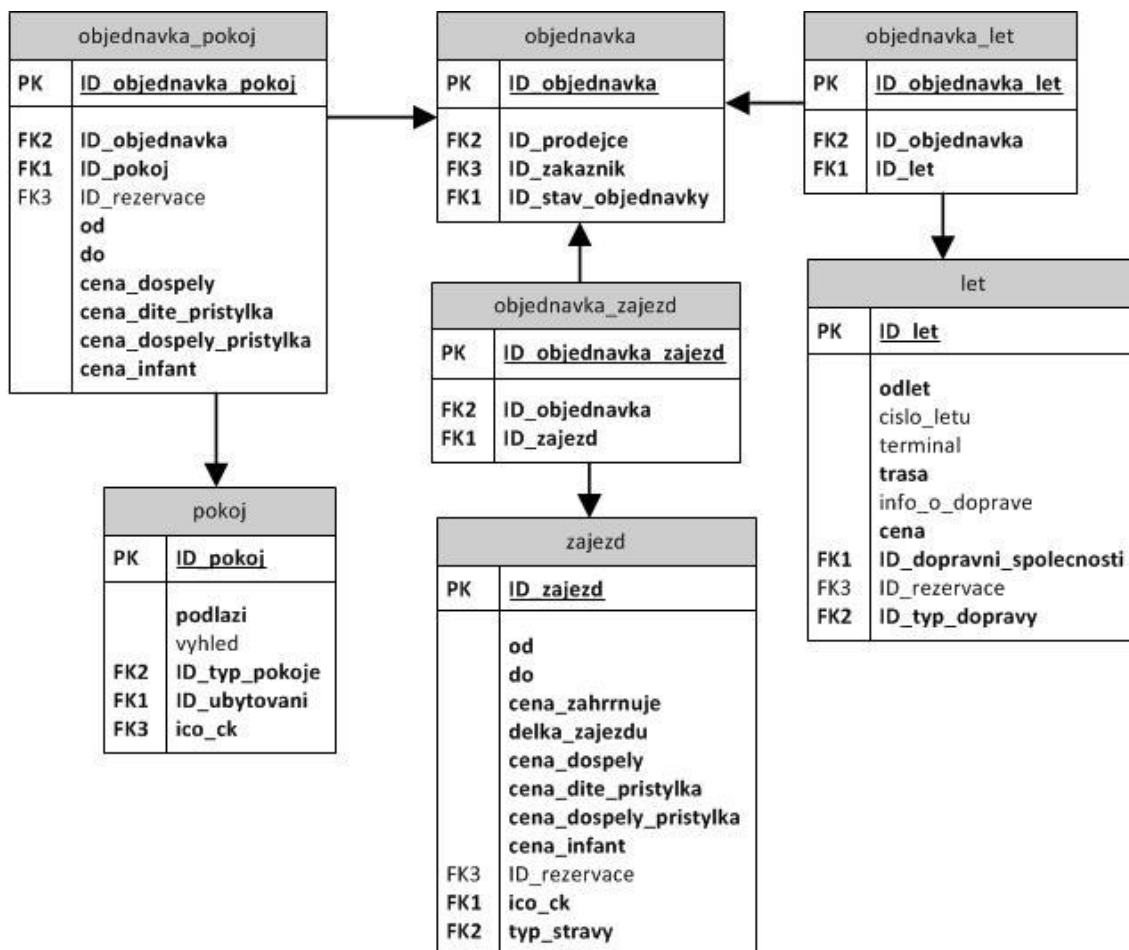
Zdroj: vlastní

3.3.7 Objednávka, let, pokoj a zájezd

Mezi tabulkou *objednavka* a *let* se nachází tabulka *objednavka_let*. Tato tabulka uchovává informace o tom, kolik si zákazník objednal služeb zajišťujících dopravu.

Podobnou tabulku jsem vytvořil i mezi tabulkou *pokoj* a *objednavka*. V této tabulce evidujeme údaje o tom, který pokoj patří do které objednávky. Dále zde zaznamenáváme termín pobytu v daném pokoji, cenu a informace o rezervaci jeli zde už nějaká vytvořena.

Jelikož zákazník si může objednat více zájezdů naráz, vytvořil jsem tabulku *objednavka_zajezd*. Ta nás informuje o tom, jaký zájezd patří do jaké objednávky.



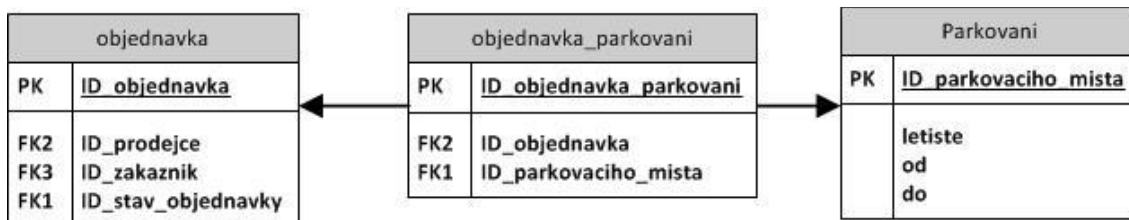
Obrázek č. 18: Objednávka, let, pokoj a zájezd

Zdroj: vlastní

3.3.8 Objednávka a parkování

Tabulka *parkovani* je určena identifikačním číslem parkovacího místa, které je automaticky generováno. Jelikož cestovní agentury nabízejí parkovací místa pouze na českých letištích, nespojoval jsem tuto tabulku s tabulkou *letiste* ale vložil pouze atribut s datovým typem varchar kde se udává název letiště na kterém je parkovací místo objednáno. Dále z tabulky můžeme vyčíst termín zamluvení parkovacího místa.

Mezi tabulkou *parkovani* a *objednavka* se nachází dekompoziční tabulka *objednavka_parkovani*, která nám říká, kolik a jakých parkovacích míst je v objednávce požadováno.



Obrázek č. 19: Objednávka a parkování

Zdroj: vlastní

3.4 Fyzický návrh

Ve fyzickém návrhu databáze bych se chtěl věnovat některým pohledům a procedurám, které jsou v databázi použity. Celý fyzický návrh databáze obsahující všechny tabulky a datové typy je v příloze č. 1. V příloze č. 2 se nachází datový slovník a zdrojový kód včetně insertů, pohledů a procedur je v příloze č. 3.

3.4.1 Procedura – přidání zákazníka

Procedura přidání zákazníka slouží pro přidání nového zákazníka do databáze. Jako vstupní parametry slouží *jmeno*, *prijmeni*, *psc*, *ulice*, *zkratka_zeme*, *telefon* a *mail*. Procedura nejprve ověří na základě telefonního čísla, jestli se v databázi už nenachází zákazník se stejným telefonním číslem. V případě shody procedura vypíše, že tento zákazník už v databázi je a v opačném případě, procedura přidá zákazníka do databáze.

```
create procedure novy_zakaznik
@jmeno varchar(20),
@prijmeni varchar(30),
@psc numeric(5),
@ulice varchar(50),
@zkratka_zeme varchar(3),
@telefon numeric(9),
@mail varchar(60)

as
begin
if not exists (select * from zakaznik.telefon where @telefon =
zakaznik.telefon)
insert into zakaznik (jmeno, prameni, psc, ulice, zkratka_zeme,
telefon, mail)
```

```

        values (@jmeno,@prijmeni,@psc, @ulice, @zkratka_zeme, @telefon,
        @mail)
else
    print ('Zákazník s tímto telefonním číslem už je v databázi')
end

```

Příklad dotazu na vyvolání procedury

```

execute novy_zakaznik
'Tomáš', 'Jedno', 76005, 'Horáková', 'CZE', 732449331, 'tomas12@seznam.cz'

```

3.4.2 Procedura – přidání nového letu

Procedura nový let se využívá pro přidání nového letu do databáze. Mezi vstupní parametry patří datum a čas odletu, trasa letu, cena, s jakou dopravní společností je let uskutečněn, letiště odkud je odlet a kam je přilet. Data se zapisují do dvou tabulek *let* a *letiste_let*.

```

create procedure novy_let
@odlet smalldatetime,
@trasa varchar(25),
@cena int,
@ID_dopravni_spolecnosti int,
@ID_typ_dopravy int,
@ID_letiste_odlet int,
@ID_letiste_prilet int,
@ID_let int,
@odlet varchar (10),
@prilet varchar (10)

as
begin
insert into let (odlet, trasa, cena, ID_dopravni_spolecnosti,
ID_typ_dopravy)
values (@odlet, @trasa, @cena, @ID_dopravni_spolecnosti,
@ID_typ_dopravy)

insert into letiste_let (ID_letiste, ID_let, odlet_prilet)
values (@ID_letiste_odlet, @ID_let, @odlet)

insert into letiste_let (ID_letiste, ID_let, odlet_prilet)
values (@ID_letiste_prilet, @ID_let, @prilet)
end

```

Příklad dotazu na vyvolání procedury

```

execute novy_let
'2014-07-21 22:00:00', 'BRN - PMI', 4800, 2, 1, 1, 2, 9, 'odlet', 'prilet'

```

3.4.3 Pohled – Souhrn objednávky

Pohled souhrn objednávky nám zobrazuje zákazníky, kteří si objednali zájezd, dobu trvání těchto zájezdů, název hotelů, typ pokojů, ve kterých budou ubytováni a jména a příjmení prodejců, kteří jim tyto zájezdy prodali. Jde o propojení devíti tabulek, podávajících základní informace o tom, kdo, komu a co prodal.

```
create view souhrn_objednavky as
select
zakaznik.jmeno, zakaznik.prijmeni, ubytovani.nazev,
zajezd.delka_zajezdu, prodejce.jmeno, prodejce.prijmeni,
typ_pokoje.nazev_typu
From zakaznik, objednavka, prodejce, objednavka_zajezd, zajezd,
zajezd_pokoj, pokoj, ubytovani, typ_pokoje
where zakaznik.ID_zakaznik = objednavka.ID_zakaznik and
prodejce.ID_prodejce = objednavka.ID_prodejce and
objednavka.ID_objednavka = objednavka_zajezd.ID_objednavka and
objednavka_zajezd.ID_zajezd = zajezd.ID_zajezd and zajezd.ID_zajezd =
zajezd_pokoj.ID_zajezd and zajezd_pokoj.ID_pokoj = pokoj.ID_pokoj and
pokoj.ID_ubytovani = ubytovani.ID_ubytovani and pokoj.ID_typ_pokoje =
typ_pokoje.ID_typ_pokoje
```

Závěr

Jako cíl své bakalářské práce jsem si určil navrhnout databázi pro cestovní agenturu. V databázi jsem chtěl zachytit evidenci všech zájezdů, které cestovní agentura nabízí s tím, že se zaměřím na jednotlivé služby, z kterých se zájezd skládá. Tím jsem chtěl docílit toho, že by se jednotlivé ubytovací kapacity neevidovaly po hotelech ale po jednotlivých pokojích a zákazník tak měl možnost nabízený zájezd upravit, nebo si požádat pouze o samostatnou službu, která k zájezdu patří bez všeho ostatního. Dále jsem zde chtěl zachytit data o cestujících, prodejcích a všech zákaznících cestovní agentury.

Díky šestiměsíční pracovní zkušenosti v cestovní agentuře, zmapování současných řešení a analýze současného stavu obsahující všechny interní procesy, jsem ve vlastním návrhu navrhl databázi, která by měla lépe informovat zaměstnance o nabízených službách a přinést menší chybovost při prodeji těchto služeb.

Dalším krokem k efektivnímu použití této databáze by bylo vytvoření informačního systému, přes který by se dalo s daty v databázi lépe manipulovat.

SEZNAM POUŽITÝCH ZDROJŮ

- [1] CONOLLY, T., C. BEGG a R. HOLOWCZAK. *Mistrovství – databáze*. Brno: Computer Press, 2009. ISBN 978-80-251-2328-7.
- [2] KOCH, M. a B. NEUWIRTH. *Datové a funkční modelování*. 4. rozš. vyd. Brno: Akademické nakladatelství CERM, 2008. ISBN 978-80-214-4125-5.
- [3] KŘÍŽ, J. a P. DOSTÁL. *Databázové systémy*. Brno: Akademické nakladatelství CERM, 2005. ISBN 80-214-3064-8.
- [4] SKŘIVAN J. Databáze a jazyk SQL. *Interval.cz* [online]. 2000 [2013-11-28]. Dostupné z: <http://interval.cz/clanky/databaze-a-jazyk-sql/>
- [5] KŘÍŽ, J. *Databázové systémy*. Přednáška. Brno: VUT, 20.11.2011.
- [6] MANAGED DEDICATED SERVERY. *Hierarchický databázový model* [online]. © 2009 [cit. 2013-12-23]. Dostupné z: <http://www.managed-dedicated-servery.net/hierarchicky-databazovy-model.html>
- [7] MANAGED DEDICATED SERVERY. *Síťový databázový model* [online]. © 2009 [cit. 2013-12-23]. Dostupné z: <http://www.managed-dedicated-servery.net/sitovy-databazovy-model.html>
- [8] LACKO, L. *1001 tipů a triků pro SQL: Sbíрка nejlepších programátorských postupů a řešení*. Brno: ComputerPress, 2008. ISBN 978-80-251-2329-4.
- [9] HOTEK, M. *Microsoft SQL Server 2008: Krok za krokem*. Brno: ComputerPress, 2009. ISBN 978-80-251-2466-6.
- [10] MASARYKOVA UNIVERZITA. *Provoz a činnost cestovních kanceláří* [online]. © 2005 [cit. 2014-03-15]. Dostupné z: http://is.muni.cz/do/econ/soubory/katedry/kres/3910085/MSMT_PCR.pdf
- [11] MAGICWARE. *Představení produktu* [online]. © 1994 - 2014 [cit. 2014-03-16]. Dostupné z: <http://www.istour.cz/predstaveni-produktu/>
- [12] STOVKA SOFTWARE. *Software pro cestovní kanceláře* [online]. [cit. 2014-03-17]. Dostupné z: <http://www.stovka.cz/>
- [13] SVAZ OBCHODU A CESTOVNÍHO RUCHU ČR. *Aktualizace oborových příruček pro provozování cestovní agentury* [online]. © 2013 [cit. 2014-03-23]. Dostupné z: http://www.socr.cz/file/273/22_op_provozovani_cestovni_agentury.pdf
- [14] OPEN TRAVEL. *Řešení pro cestovní kanceláře a agentury* [online]. © 2014 [cit. 2014-03-16]. Dostupné z: <http://www.opentravel.cz/>

[15] DARKMAY. *CeSYS - Řešení pro cestovní kanceláře a agentury* [online]. © 2014 [cit. 2014-03-16]. Dostupné z: <http://www.cestovnisystem.cz/>

[16] PEAR. *Produkty* [online]. © 2011 [cit. 2014-03-17]. Dostupné z: <http://www.codoma.cz/?menu1=produkty>

SEZNAM TABULEK

Tab. 1: Číselné datové typy.....	20
Tab. 2: Znakové (textové) datové typy.....	21
Tab. 3: Datové typy pro datum a čas	21
Tab. 4: Binární datové typy.....	21
Tab. 5: Základní cenové programy.....	30
Tab. 6: Seznam entit.....	38
Tab. 7: Identifikace relací mezi entitami.....	39

SEZNAM OBRÁZKŮ

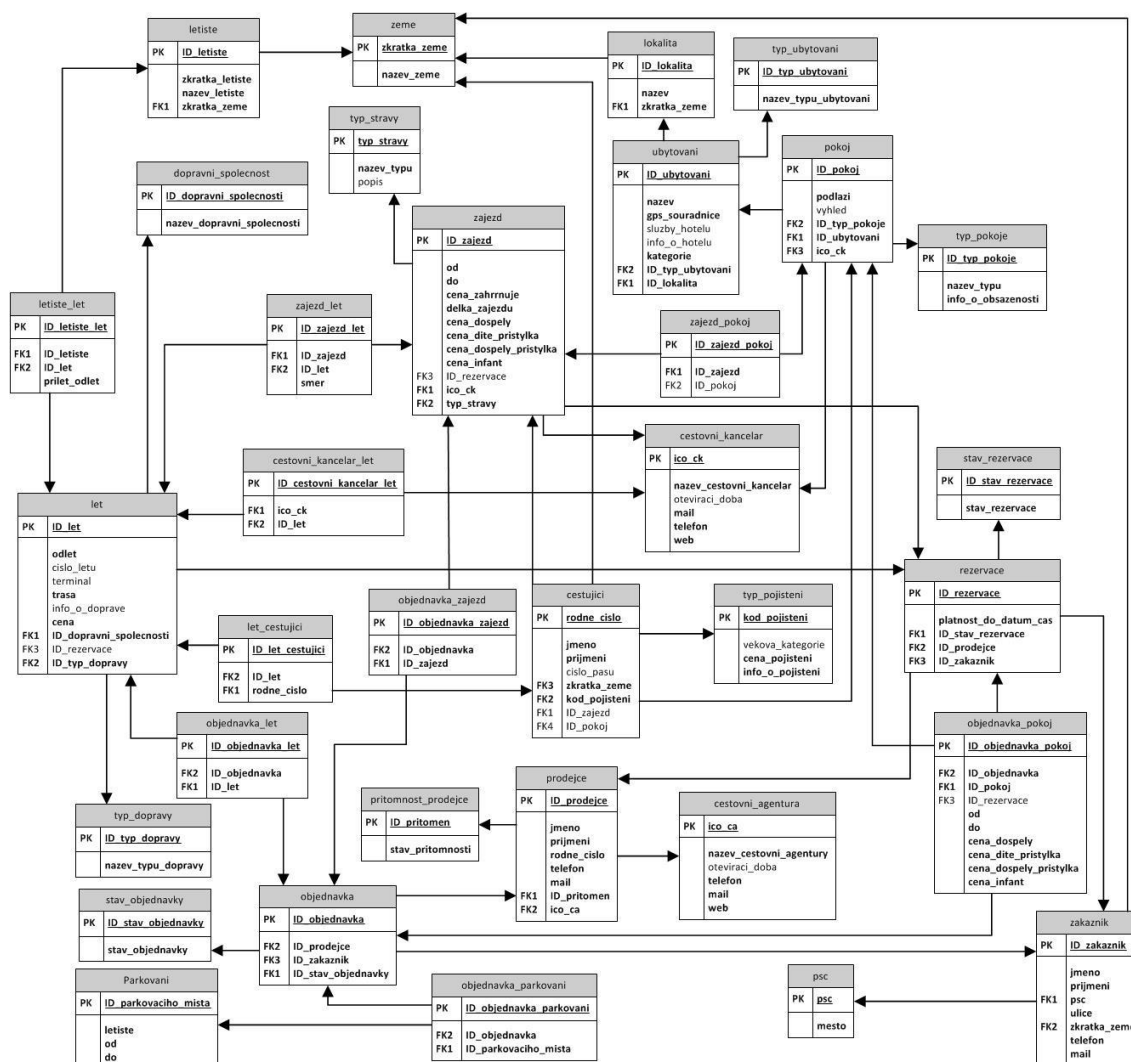
Obrázek č. 1: Vztah jedna ku jedné.....	15
Obrázek č. 2: Vztah jeden k více.....	16
Obrázek č. 3: Vztah více ku více.....	16
Obrázek č. 4: Rozložení vztahu pomocí spojovací tabulky.....	17
Obrázek č. 5: Subjekty na trhu cestovního ruchu.....	24
Obrázek č. 6: DFD Diagram procesu vyřízení objednávky.....	27
Obrázek č. 7: Grafické schéma umístění jednotek v zařízení.....	28
Obrázek č. 8: Vývojový diagram – běžný chod společnosti.....	34
Obrázek č. 9: Vývojový diagram – vznik objednávky.....	35
Obrázek č. 10: Vývojový diagram – vyřízení doplňkových služeb, platby a smluv.....	37
Obrázek č. 11: ER Diagram.....	41
Obrázek č. 12: Zákazník, prodejce a objednávka.....	42
Obrázek č. 13: Rezervace.....	43

Obrázek č. 14: Let, letiště a dopravní společnost.....	44
Obrázek č. 15: Země, lokalita, ubytování a pokoj.....	45
Obrázek č. 16: Cestovní kancelář, zájezd, pokoj a let.....	46
Obrázek č. 17: Cestující, let a typ pojištění.....	47
Obrázek č. 18: Objednávka, let, pokoj a zájezd.....	48
Obrázek č. 19: Objednávka a parkování.....	49

SEZNAM PŘÍLOH

Příloha 1: Fyzický návrh databáze.....	57
Příloha 2: Datový slovník.....	58
Příloha 3: Zdrojový kód.....	63

Příloha1: Fyzický návrh databáze



Příloha 2: Datový slovník

Tabulka	Atributy	Typ	Délka	PK, FK	NOT NULL
cestující	rodne_cislo	numeric	10	PK	X
	jmeno	varchar	20		X
	prijmeni	varchar	30		X
	cislo_pasu	numeric	8;0		
	zkratka_zeme	varchar	3	FK	X
	kod_pojisteni	varchar	3	FK	X
	ID_zajezd	int		FK	
	ID_pokoj	int		FK	
zeme	zkratka_zeme	varchar	3	PK	X
	nazev_zeme	varchar	50		X
lokalita	ID_lokalita	int		PK	X
	nazev	varchar	40		X
	zkratka_zeme	varchar	3	FK	X
ubytovani	ID_ubytovani	int		PK	X
	nazev	varchar	50		X
	gps_souradnice	varchar	40		X
	sluzby_hotelu	varchar	300		
	info_o_hotelu	varchar	300		
	kategorie	varchar	5		X
	ID_typ_ubytovani	tinyint		FK	X
	ID_lokalita	int		FK	X
pokoj	ID_pokoj	int		PK	X
	podlazi	tinyint			X
	vyhled	varchar	20		
	ID_typ_pokoje	tinyint		FK	X
	ID_ubytovani	int		FK	X
	ico_ck	numeric	8;0	FK	X
typ_pokoje	ID_typ_pokoje	tinyint		PK	X
	nazev_typu	varchar	5		X
	info_o_obsazenosti	varchar	100		X

cestovni_kancelar	ico_ck	numeric	8;0	PK	X
	nazev_cestovni_kancelar	varchar	30		X
	otevírací_doba	varchar	100		
	mail	varchar	60		X
	telefon	numeric	9;0		X
	web	varchar	40		X
zajezd	ID_zajezd	int		PK	X
	od	date			X
	do	date			X
	cena_zahrnuje	varchar	300		X
	delka_zajezdu	tinyint			X
	cena_dospely	int			X
	cena_dite_pristylka	int			X
	cena_dospely_pristylka	int			X
	cena_infant	int			X
	typ_stravy	varchar	4	FK	X
	ID_rezervace	int		FK	
	ico_ck	numeric	8;0	FK	X
typ_stravy	typ_stravy	varchar	4	PK	X
	nazev_typu	varchar	30		X
	popis	varchar	200		
typ_ubytovani	ID_typ_ubytovani	tinyint		PK	X
	nazev_typu_ubytovani	varchar	20		X
letiste	ID_letiste	int		PK	X
	zkratka_letiste	varchar	3		X
	nazev_letiste	varchar	40		X
	zkratka_zeme	varchar	3	FK	X
let	ID_let	int		PK	X
	odlet	date			X
	cislo_letu	varchar	6		
	terminal	varchar	15		
	trasa	varchar	25		X
	info_o_doprave	varchar	200		

	cena	int			X
	ID_dopravni_spolecnosti	int		FK	X
	ID_rezervace	int		FK	
	ID_typ_dopravy	tinyint		FK	X
dopravni_spolecnost	ID_dopravni_spolecnosti	int		PK	X
	nazev_dopravni_spolecnosti	varchar	30		X
typ_pojisteni	kod_pojisteni	varchar	3	PK	X
	vekova_kategorie	varchar	20		
	cena_pojisteni	int			X
	info_o_pojisteni	varchar	200		X
cestovni_agentura	ico_ca	numeric	8;0	PK	X
	nazev_cestovni_agentury	varchar	30		X
	oteviraci_doba	varchar	100		
	telefon	numeric	9;0		X
	mail	varchar	60		X
	web	varchar	40		X
typ_dopravy	ID_typ_dopravy	tinyint		PK	X
	nazev_typu_dopravy	varchar	15		X
prodejce	ID_prodejce	int		PK	X
	jmeno	varchar	20		X
	prijmeni	varchar	30		X
	rodne_cislo	numeric	10;0		X
	telefon	numeric	9;0		X
	mail	varchar	60		X
	ID_pritomen	tinyint		FK	X
	ico_ca	numeric	8;0	FK	X
objednavka	ID_objednavka	int		PK	X
	ID_stav_objednavky	tinyint		FK	X
	ID_prodejce	int		FK	X
	ID_zakaznik	int		FK	X
stav_rezervace	ID_stav_rezervace	tinyint		PK	X
	stav_rezervace	varchar	30		X
stav_objednavky	ID_stav_objednavky	tinyint		PK	X
	stav_objednavky	varchar	20		X

pritomnost_prodejce	ID_pritomen	tinyint		PK	X
	stav_pritomnosti	varchar	20		X
parkovani	ID_parkovaciho_mista	int		PK	X
	letiste	varchar	15		X
	od	date			X
	do	date			X
zakaznik	ID_zakaznik	int		PK	X
	jmeno	varchar	20		X
	prijmeni	varchar	30		X
	psc	numeric	5;0	FK	X
	ulice	varchar	50		X
	zkratka_zeme	varchar	3	FK	X
	telefon	numeric	9;0		X
	mail	varchar	60		X
psc	psc	numeric	5;0	PK	X
	mesto	varchar	40		X
rezervace	ID_rezervace	int		PK	X
	platnost_do_datum_cas	date			X
	ID_stav_rezervace	tinyint		FK	X
	ID_prodejce	int		FK	X
	ID_zakaznik	int		FK	X
objednavka_zajezd	ID_objednavka_zajezd	int		PK	X
	ID_objednavka	int		FK	X
	ID_zajezd	int		FK	X
cestovni_kancelar_let	ID_cestovni_kancelar_let	int		PK	X
	ico_ck	numeric	8;0	FK	X
	ID_let	int		FK	X
let_cestujici	ID_let_cestujici	int		PK	X
	ID_let	int		FK	X
	rodne_cislo	numeric	10	FK	X
objednavka_parkovani	ID_objednavka_parkovani	int		PK	X
	ID_objednavka	int		FK	X
	ID_parkovani	int		FK	X

zajezd_pokoj	ID_zajezd_pokoj	int		PK	X
	ID_zajezd	int		FK	X
	ID_pokoj	int		FK	X
zajezd_let	ID_zajezd_let	int		PK	X
	ID_zajezd	int		FK	X
	ID_let	int		FK	X
	smer	varchar	10		X
objednavka_let	ID_objednavka_let	int		PK	X
	ID_objednavka	int		FK	X
	ID_let	int		FK	X
objednavka_pokoj	ID_objednavka_pokoj	int		PK	X
	ID_objednavka	int		FK	X
	ID_pokoj	int		FK	X
	od	date			X
	cena_dospely	int			X
	cena_dite_pristylka	int			X
	cena_dospely_pristylka	int			X
	cena_infant	int			X
	do	date			X
	ID_rezervace	int		FK	
letiste_let	ID_letiste_let	int		PK	X
	ID_letiste	int		FK	X
	ID_let	int		FK	X
	odlet_prilet	varchar	10		X

Příloha 3: Zdrojový kód

```
create table zeme
(
zkratka_zeme varchar (3) primary key NOT NULL,
nazev_zeme varchar (50) NOT NULL
)

create table typ_pokoje
(
ID_typ_pokoje tinyint identity (1,1) primary key NOT NULL,
nazev_typu varchar (5) NOT NULL,
info_o_obsazenosti varchar (100) NOT NULL
)

create table stav_rezervace
(
ID_stav_rezervace tinyint identity (1,1) primary key NOT NULL,
stav_rezervace varchar (30) NOT NULL
)

create table cestovni_kancelar
(
ico_ck numeric (8) primary key NOT NULL,
nazev_cestovni_kancelar varchar (30) NOT NULL,
oteviraci_doba varchar (100),
mail varchar (60) NOT NULL,
telefon numeric (9) NOT NULL,
web varchar (40) NOT NULL
)

create table stav_objednavky
(
ID_stav_objednavky tinyint identity (1,1) primary key NOT NULL,
stav_objednavky varchar (20) NOT NULL
)

create table typ_stravy
(
typ_stravy varchar (4) primary key NOT NULL,
nazev_typu varchar (30) NOT NULL,
popis varchar (200)
)

create table typ_ubytovani
(
ID_typ_ubytovani tinyint identity (1,1) primary key NOT NULL,
nazev_typu_ubytovani varchar (20) NOT NULL
)

create table dopravni_spolecnost
(
ID_dopravni_spolecnosti int identity (1,1) primary key NOT NULL,
nazev_dopravni_spolecnosti varchar (30) NOT NULL
)

create table typ_pojisteni
(
kod_pojisteni varchar (3) primary key NOT NULL,
```

```

vekova_kategorie varchar (20),
cena_pojisteni int NOT NULL,
info_o_pojisteni varchar (200) NOT NULL
)

create table cestovni_agentura
(
ico_ca numeric (8) primary key NOT NULL,
nazev_cestovni_agentury varchar (30) NOT NULL,
oteviraci_doba varchar (100),
telefon numeric (9) NOT NULL,
mail varchar (60) NOT NULL,
web varchar (40) NOT NULL
)

create table typ_dopravy
(
ID_typ_dopravy tinyint identity (1,1) primary key NOT NULL,
nazev_typu_dopravy varchar (15) NOT NULL
)

create table pritomnost_prodejce
(
ID_pritomen tinyint identity (1,1) primary key NOT NULL,
stav_pritomnosti varchar (20) NOT NULL
)

create table parkovani
(
ID_parkovaciho_mista int identity (1,1) primary key NOT NULL,
letiste varchar (15) NOT NULL,
od smalldatetime NOT NULL,
do smalldatetime NOT NULL
)

create table psc
(
psc numeric (5) primary key NOT NULL,
mesto varchar (40) NOT NULL
)

create table lokalita
(
ID_lokalita int identity (1,1) primary key NOT NULL,
nazev varchar (40) NOT NULL,
zkratka_zeme varchar (3) foreign key references zeme (zkratka_zeme)
NOT NULL
)

create table ubytovani
(
ID_ubytovani int identity (1,1) primary key NOT NULL,
nazev varchar (50) NOT NULL,
gps_souradnice varchar (40) NOT NULL,
sluzby_hotelu varchar (300),
info_o_hotelu varchar (300),
kategorie varchar (5) NOT NULL,
ID_typ_ubytovani tinyint foreign key references typ_ubytovani
(ID_typ_ubytovani) NOT NULL,

```

```

ID_lokalita int foreign key references lokalita (ID_lokalita) NOT NULL
)

create table letiste
(
ID_letiste int identity (1,1) primary key NOT NULL,
zkratka_letiste varchar (3) NOT NULL,
nazev_letiste varchar (40) NOT NULL,
zkratka_zeme varchar (3) foreign key references zeme (zkratka_zeme)
NOT NULL
)

create table prodejce
(
ID_prodejce int identity (1,1) primary key NOT NULL,
jmeno varchar (20) NOT NULL,
prijmeni varchar (30) NOT NULL,
rodne_cislo numeric (10) NOT NULL,
telefon numeric (9) NOT NULL,
mail varchar (60) NOT NULL,
ID_pritomen tinyint foreign key references pritomnost_prodejce
(ID_pritomen) NOT NULL,
ico_ca numeric (8) foreign key references cestovni_agentura (ico_ca)
NOT NULL
)

create table zakaznik
(
ID_zakaznik int identity (1,1) primary key NOT NULL,
jmeno varchar (20) NOT NULL,
prijmeni varchar (20) NOT NULL,
psc numeric (5) foreign key references psc (psc) NOT NULL,
ulice varchar (50) NOT NULL,
zkratka_zeme varchar (3) foreign key references zeme (zkratka_zeme)
NOT NULL,
telefon numeric (9) NOT NULL,
mail varchar (60) NOT NULL
)

create table rezervace
(
ID_rezervace int identity (1,1) primary key NOT NULL,
platnost_do_datum_cas smalldatetime NOT NULL,
ID_stav_rezervace tinyint foreign key references stav_rezervace
(ID_stav_rezervace) NOT NULL,
ID_prodejce int foreign key references prodejce (ID_prodejce) NOT
NULL,
ID_zakaznik int foreign key references zakaznik (ID_zakaznik) NOT NULL
)

create table pokoj
(
ID_pokoj int identity (1,1) primary key NOT NULL,
podlazi tinyint NOT NULL,
vyhled varchar (20),
ID_typ_pokoje tinyint foreign key references typ_pokoje
(ID_typ_pokoje) NOT NULL,

```

```

ID_ubytovani int foreign key references ubytovani (ID_ubytovani) NOT
NULL,
ico_ck numeric (8) foreign key references cestovni_kancelar (ico_ck)
NOT NULL
)

create table zajezd
(
ID_zajezd int identity (1,1) primary key NOT NULL,
od smalldatetime NOT NULL,
do smalldatetime NOT NULL,
cena_zahrnuje varchar (300) NOT NULL,
delka_zajezdu tinyint NOT NULL,
cena_dospely int NOT NULL,
cena_dite_pristylka int NOT NULL,
cena_dospely_pristylka int NOT NULL,
cena_infant int NOT NULL,
typ_stravy varchar (4) foreign key references typ_stravy (typ_stravy)
NOT NULL,
ID_rezervace int foreign key references rezervace (ID_rezervace),
ico_ck numeric (8) foreign key references cestovni_kancelar (ico_ck)
NOT NULL
)

create table let
(
ID_let int identity (1,1) primary key NOT NULL,
odlet smalldatetime NOT NULL,
cislo_letu varchar (6),
terminal varchar (15),
trasa varchar (25) NOT NULL,
info_o_doprave varchar (200),
cena int NOT NULL,
ID_dopravni_spolecnosti int foreign key references dopravni_spolecnost
(ID_dopravni_spolecnosti) NOT NULL,
ID_rezervace int foreign key references rezervace (ID_rezervace),
ID_typ_dopravy tinyint foreign key references typ_dopravy
(ID_typ_dopravy) NOT NULL
)

create table objednavka
(
ID_objednavka int identity (1,1) primary key NOT NULL,
ID_stav_objednavky tinyint foreign key references stav_objednavky
(ID_stav_objednavky) NOT NULL,
ID_prodejce int foreign key references prodejce (ID_prodejce) NOT
NULL,
ID_zakaznik int foreign key references zakaznik (ID_zakaznik) NOT NULL
)

create table cestujici
(
rodne_cislo numeric (10) primary key NOT NULL,
jmeno varchar (20) NOT NULL,
prijmeni varchar (30) NOT NULL,
cislo_pasu numeric (8),
zkratka_zeme varchar (3) foreign key references zeme (zkratka_zeme)
NOT NULL,

```

```

kod_pojisteni varchar (3) foreign key references typ_pojisteni
(kod_pojisteni) NOT NULL,
ID_zajezd int foreign key references zajezd (ID_zajezd),
ID_pokoj int foreign key references pokoj (ID_pokoj)
)

create table objednavka_zajezd
(
ID_objednavka_zajezd int identity (1,1) primary key NOT NULL,
ID_objednavka int foreign key references objednavka (ID_objednavka)
NOT NULL,
ID_zajezd int foreign key references zajezd (ID_zajezd) NOT NULL
)

create table cestovni_kancelar_let
(
ID_cestovni_kancelar_let int identity (1,1) primary key NOT NULL,
ico_ck numeric (8) foreign key references cestovni_kancelar (ico_ck)
NOT NULL,
ID_let int foreign key references let (ID_let) NOT NULL
)

create table let_cestujici
(
ID_let_cestujici int identity (1,1) primary key NOT NULL,
ID_let int foreign key references let (ID_let) NOT NULL,
rodne_cislo numeric (10) foreign key references cestujici
(rodne_cislo) NOT NULL
)

create table objednavka_parkovani
(
ID_objednavka_parkovani int identity (1,1) primary key NOT NULL,
ID_objednavka int foreign key references objednavka (ID_objednavka)
NOT NULL,
ID_parkovani int foreign key references parkovani
(ID_parkovaciho_mista) NOT NULL
)

create table zajezd_pokoj
(
ID_zajezd_pokoj int identity (1,1) primary key NOT NULL,
ID_zajezd int foreign key references zajezd (ID_zajezd) NOT NULL,
ID_pokoj int foreign key references pokoj (ID_pokoj) NOT NULL
)

create table zajezd_let
(
ID_zajezd_let int identity (1,1) primary key NOT NULL,
ID_zajezd int foreign key references zajezd (ID_zajezd) NOT NULL,
ID_let int foreign key references let (ID_let) NOT NULL,
smer varchar (10) NOT NULL
)

create table objednavka_let
(
ID_objednavka_let int identity (1,1) primary key NOT NULL,
ID_objednavka int foreign key references objednavka (ID_objednavka)
NOT NULL,

```

```

ID_let int foreign key references let (ID_let) NOT NULL
)

create table objednavka_pokoj
(
ID_objednavka_pokoj int identity (1,1) primary key NOT NULL,
ID_objednavka int foreign key references objednavka (ID_objednavka)
NOT NULL,
ID_pokoj int foreign key references pokoj (ID_pokoj) NOT NULL,
ID_rezervace int foreign key references rezervace (ID_rezervace),
od smalldatetime NOT NULL,
do smalldatetime NOT NULL,
cena_dospely int NOT NULL,
cena_dite_pristylka int NOT NULL,
cena_dospely_pristylka int NOT NULL,
cena_infant int NOT NULL
)
create table letiste_let
(
ID_letiste_let int identity (1,1) primary key NOT NULL,
ID_letiste int foreign key references letiste (ID_letiste) NOT NULL,
ID_let int foreign key references let (ID_let) NOT NULL,
odlet_prilet varchar (10) NOT NULL
)

insert into zeme (zkratka_zeme, nazev_zeme) values ('CZE', 'Česká
republika')
insert into zeme (zkratka_zeme, nazev_zeme) values ('ESP',
'Spanělsko')
insert into zeme (zkratka_zeme, nazev_zeme) values ('ITA', 'Itálie')
insert into zeme (zkratka_zeme, nazev_zeme) values ('BGR',
'Bulharsko')

insert into typ_pokoje (nazev_typu, info_o_obsazenosti)
values ('dbl', 'Dvě pevné lůžka pro dospělé')
insert into typ_pokoje (nazev_typu, info_o_obsazenosti)
values ('dbl+1', 'Dvě pevné lůžka pro dospělé + přistýlka')
insert into typ_pokoje (nazev_typu, info_o_obsazenosti)
values ('dbl+2', 'Dvě pevné lůžka pro dospělé + dvě přistýlky')
insert into typ_pokoje (nazev_typu, info_o_obsazenosti)
values ('trp', 'Tři pevné lůžka pro dospělé')

insert into stav_rezervace (stav_rezervace) values ('Bez rezervace')
insert into stav_rezervace (stav_rezervace) values ('Nezávazně
zarezervováno')
insert into stav_rezervace (stav_rezervace) values ('Závazně
zarezervováno')

insert into cestovni_kancelar (ico_ck, nazev_cestovni_kancelar,
oteviraci_doba, mail, telefon, web)
values (45312974, 'Exim tours a.s.', 'Po - Ne 8:00 -
20:00', 'online@eximtours.cz', 841115115, 'http://www.eximtours.cz/')
insert into cestovni_kancelar (ico_ck, nazev_cestovni_kancelar,
oteviraci_doba, mail, telefon, web)
values (27869237, 'FIRO-tour a.s.', 'Po - So 9:00 -
18:00', 'firo@firotour.cz', 210000210, 'http://www.firotour.cz/')
insert into cestovni_kancelar (ico_ck, nazev_cestovni_kancelar,
oteviraci_doba, mail, telefon, web)

```

```

values (26141647, 'FISCHER, a.s.', 'Po - Ne 7:00 -
21:00', 'info@fischer.cz', 800121010, 'http://www.fischer.cz/')
insert into cestovni_kancelar (ico_ck, nazev_cestovni_kancelar,
oteviraci_doba, mail, telefon, web)
values (60192755, 'Čedok a.s.', 'Po - So 9:00 -
19:30', 'info@cedok.cz', 800112112, 'http://www.cedok.cz/')

insert into stav_objednavky (stav_objednavky) values ('Nová obj.')
insert into stav_objednavky (stav_objednavky) values ('Obj. se
vyřizuje')
insert into stav_objednavky (stav_objednavky) values ('Vyřízená obj.')

insert into typ_stravy (typ_stravy, nazev_typu, popis)
values ('AI', 'All inclusive', '24 hodin denní, zahrnuje snídani, oběd a
večeři formou bufetu')
insert into typ_stravy (typ_stravy, nazev_typu, popis)
values ('HB', 'Polopenze', 'Snídani a večeře formou bufetu')
insert into typ_stravy (typ_stravy, nazev_typu, popis)
values ('OB', 'Pouze snídani', 'Strava zahrnuje pouze snídani formou
švédských stolů')
insert into typ_stravy (typ_stravy, nazev_typu, popis)
values ('NO', 'Bez stravy', 'bez stravy')

insert into typ_ubytovani (nazev_typu_ubytovani) values ('Hotel')
insert into typ_ubytovani (nazev_typu_ubytovani) values ('Aptmán')
insert into typ_ubytovani (nazev_typu_ubytovani) values ('Pension')
insert into typ_ubytovani (nazev_typu_ubytovani) values ('Jiné')

insert into dopravni_spolecnost (nazev_dopravni_spolecnosti) values
('Wizz Air')
insert into dopravni_spolecnost (nazev_dopravni_spolecnosti) values
('Travel Service')
insert into dopravni_spolecnost (nazev_dopravni_spolecnosti) values
('Emirates')
insert into dopravni_spolecnost (nazev_dopravni_spolecnosti) values
('Ryanair')

insert into typ_pojisteni (kod_pojisteni, vekova_kategorie,
cena_pojisteni, info_o_pojisteni)
values ('2I', 'dospělý', 499, 'léčebné výlohy, úrazové pojištění,
odpovědnost za škodu, pojištění zavazadel, právní asistence')
insert into typ_pojisteni (kod_pojisteni, vekova_kategorie,
cena_pojisteni, info_o_pojisteni)
values ('2IC', 'dítě do 18 let', 299, 'léčebné výlohy, úrazové
pojištění, odpovědnost za škodu, pojištění zavazadel, právní
asistence')
insert into typ_pojisteni (kod_pojisteni, vekova_kategorie,
cena_pojisteni, info_o_pojisteni)
values ('2ES', 'dospělý', 899, 'léčebné výlohy, úrazové pojištění,
odpovědnost za škodu, pojištění zavazadel, právní asistence extrémní
sporty')
insert into typ_pojisteni (kod_pojisteni, vekova_kategorie,
cena_pojisteni, info_o_pojisteni)
values ('NO', 'NO', 0, 'bez pojištění')

insert into cestovni_agentura (ico_ca, nazev_cestovni_agentury,
oteviraci_doba, telefon, mail, web)
values (26702924, 'Invia.cz, a.s.', 'Po - Ne 7:00 -
22:00', 800999800, 'poradce@invia.cz', 'http://www.invia.cz/')

```

```

insert into cestovni_agentura (ico_ca, nazev_cestovni_agentury,
oteviraci_doba, telefon, mail, web)
values (35884797, 'Invia.sk, a.s.', 'Po - Ne 7:00 -
22:00', 800900100, 'poradca@invia.sk', 'http://www.invia.sk/')
insert into cestovni_agentura (ico_ca, nazev_cestovni_agentury,
oteviraci_doba, telefon, mail, web)
values (43226313, 'Invia.hu, a.s.', 'Po - Ne 7:00 -
22:00', 680100200, 'segitseg@invia.hu', 'http://www.invia.hu/')

insert into typ_dopravy (nazev_typu_dopravy) values ('Letecká')
insert into typ_dopravy (nazev_typu_dopravy) values ('Vlastní')
insert into typ_dopravy (nazev_typu_dopravy) values ('Autobusová')

insert into pritomnost_prodejce (stav_pritomnosti) values ('Pritomen')
insert into pritomnost_prodejce (stav_pritomnosti) values
('Nepracuje')
insert into pritomnost_prodejce (stav_pritomnosti) values
('Nepritomen')
insert into pritomnost_prodejce (stav_pritomnosti) values ('Dovolená')

insert into parkovani (letiste, od, do) values ('Praha', '2014-07-14
00:00:00', '2014-07-21 00:00:00')
insert into parkovani (letiste, od, do) values ('Praha', '2014-08-01
00:00:00', '2014-08-07 00:00:00')
insert into parkovani (letiste, od, do) values ('Brno', '2014-08-10
00:00:00', '2014-08-17 00:00:00')
insert into parkovani (letiste, od, do) values ('Brno', '2014-09-02
00:00:00', '2014-09-09 00:00:00')

insert into psc (psc, mesto) values (11004, 'Praha')
insert into psc (psc, mesto) values (61200, 'Brno')
insert into psc (psc, mesto) values (53002, 'Pardubice')
insert into psc (psc, mesto) values (76005, 'Zlín')

insert into lokalita (nazev, zkratka_zeme) values ('Mallorca', 'ESP')
insert into lokalita (nazev, zkratka_zeme) values ('Sicílie', 'ITA')
insert into lokalita (nazev, zkratka_zeme) values ('Slunečné
pobřeží', 'BGR')
insert into lokalita (nazev, zkratka_zeme) values ('Primorsko', 'BGR')

insert into ubytovani (nazev, gps_souradnice, sluzby_hotelu,
info_o_hotelu, kategorie, ID_typ_ubytovani, ID_lokalita)
values ('Zefir beach', '42.667306, 27.710333', 'bowling, minigolf,
vodní sporty',
'několik metrů od jemné písčité pláže s pozvolným vstupem do
moře', '****', 1, 3)
insert into ubytovani (nazev, gps_souradnice, sluzby_hotelu,
info_o_hotelu, kategorie, ID_typ_ubytovani, ID_lokalita)
values ('Hotel Primorsko', '42.265699, 27.751806', 'služby recepce pro
objednání výletů a zapůjčení aut',
'150 m od písčité jižní pláže a ideální pozici v blízkosti
centra', '****', 1, 4)
insert into ubytovani (nazev, gps_souradnice, sluzby_hotelu,
info_o_hotelu, kategorie, ID_typ_ubytovani, ID_lokalita)
values ('Blu Hotel Portorosa', '38.122294, 15.108832', 'recepce,
restaurace, bar u bazénu, bazén, sluneční terasa, Wi-Fi',
'písčná pláž s pozvolným vstupem do moře, plážový
servis', '*****', 1, 2)

```

```

insert into ubytovani (nazev, gps_souradnice, sluzby_hotelu,
info_o_hotelu, kategorie, ID_typ_ubytovani, ID_lokalita)
values ('Mariant', '39.569316, 3.372559', 'stolní tenis, fitness,
tenisový kurt, golfová hřiště, Wi-Fi',
'Menší písečná pláž s pozvolným vstupem do moře je vzdálena asi 300 m
od hotelu', '***', 1, 1)

insert into letiste (zkratka_letiste, nazev_letiste, zkratka_zeme)
values ('PRG', 'Letiště Václava Havla Praha', 'CZE')
insert into letiste (zkratka_letiste, nazev_letiste, zkratka_zeme)
values ('PMI', 'Letiště Palma De Mallorca', 'ESP')
insert into letiste (zkratka_letiste, nazev_letiste, zkratka_zeme)
values ('BOJ', 'Letiště Bargas', 'BGR')
insert into letiste (zkratka_letiste, nazev_letiste, zkratka_zeme)
values ('PMO', 'Letiště Palermo', 'ITA')
insert into letiste (zkratka_letiste, nazev_letiste, zkratka_zeme)
values ('BRQ', 'Letiště Brno Tuřany', 'CZE')

insert into prodejce (jmeno, prijmeni, rodne_cislo, telefon, mail,
ID_pritomen, ico_ca)
values
('Pavel', 'Špaček', 9002106396, 533111676, 'pavel.spacek@invia.cz', 1, 26702
924)
insert into prodejce (jmeno, prijmeni, rodne_cislo, telefon, mail,
ID_pritomen, ico_ca)
values
('Roman', 'Procházka', 8812108456, 533111223, 'roman.prochazka@invia.cz', 1
, 26702924)
insert into prodejce (jmeno, prijmeni, rodne_cislo, telefon, mail,
ID_pritomen, ico_ca)
values
('Monika', 'Tesařová', 9311052345, 533111503, 'monika.tesarova@invia.cz', 1
, 26702924)
insert into prodejce (jmeno, prijmeni, rodne_cislo, telefon, mail,
ID_pritomen, ico_ca)
values
('Peter', 'Švarný', 9204212277, 533100348, 'peter.svarny@invia.sk', 1, 35884
797)

insert into zakaznik (jmeno, prijmeni, psc, ulice, zkratka_zeme,
telefon, mail)
values ('Jaroslav', 'Novák', 53002, 'Bajkalská
8322', 'CZE', 732844321, 'jarek.novak11@seznam.cz')
insert into zakaznik (jmeno, prijmeni, psc, ulice, zkratka_zeme,
telefon, mail)
values ('Iveta', 'Stojaspalová', 61200, 'Václavská
6433', 'CZE', 777349203, 'ivet.stoj02@seznam.cz')
insert into zakaznik (jmeno, prijmeni, psc, ulice, zkratka_zeme,
telefon, mail)
values ('Richard', 'Turek', 11004, 'Beldova
666', 'CZE', 604573055, 'turek44@gmail.com')
insert into zakaznik (jmeno, prijmeni, psc, ulice, zkratka_zeme,
telefon, mail)
values ('Božena', 'Tomancová', 76005, 'Fabiánská
332', 'CZE', 737937397, 'tomancova.b@centrum.cz')

insert into rezervace (platnost_do_datum_cas, ID_stav_rezervace,
ID_prodejce, ID_zakaznik)
values ('2014-06-22 14:45:00', 2, 1, 3)

```

```

insert into rezervace (platnost_do_datum_cas, ID_stav_rezervace,
ID_prodejce, ID_zakaznik)
values ('2014-06-20 11:30:00',2,2,1)
insert into rezervace (platnost_do_datum_cas, ID_stav_rezervace,
ID_prodejce, ID_zakaznik)
values ('2014-06-19 15:00:00',2,3,2)
insert into rezervace (platnost_do_datum_cas, ID_stav_rezervace,
ID_prodejce, ID_zakaznik)
values ('2014-06-25 16:30:00',2,2,4)

insert into pokoj (podlazi, vyhled, ID_typ_pokoje, ID_ubytovani,
ico_ck) values (3, 'Na moře',1,4,45312974)
insert into pokoj (podlazi, vyhled, ID_typ_pokoje, ID_ubytovani,
ico_ck) values (4, 'Do zahrady',2,2,26141647)
insert into pokoj (podlazi, vyhled, ID_typ_pokoje, ID_ubytovani,
ico_ck) values (2, 'Do zahrady',3,3,27869237)
insert into pokoj (podlazi, vyhled, ID_typ_pokoje, ID_ubytovani,
ico_ck) values (4, 'Na moře',1,1,60192755)

insert into zajezd (od, do, cena_zahrnuje, delka_zajezdu,
cena_dospely, cena_dite_pristylka, cena_dospely_pristylka,
cena_infant, typ_stravy, ID_rezervace, ico_ck)
values ('2014-07-14 00:00:00', '2014-07-21 00:00:00', 'transfer z hotelu
a zpět, pojištění ck proti úpadku, služby
delegáta',8,16499,8999,14999,990, 'AI',1,45312974)
insert into zajezd (od, do, cena_zahrnuje, delka_zajezdu,
cena_dospely, cena_dite_pristylka, cena_dospely_pristylka,
cena_infant, typ_stravy, ID_rezervace, ico_ck)
values ('2014-08-01 00:00:00', '2014-08-07 00:00:00', 'transfer z hotelu
a zpět, pojištění ck proti úpadku, služby
delegáta',8,15499,7999,12999,990, 'HB',2,26141647)
insert into zajezd (od, do, cena_zahrnuje, delka_zajezdu,
cena_dospely, cena_dite_pristylka, cena_dospely_pristylka,
cena_infant, typ_stravy, ID_rezervace, ico_ck)
values ('2014-08-10 00:00:00', '2014-08-17 00:00:00', 'transfer z hotelu
a zpět, pojištění ck proti úpadku, služby
delegáta',8,18499,10999,15999,990, 'AI',3,27869237)
insert into zajezd (od, do, cena_zahrnuje, delka_zajezdu,
cena_dospely, cena_dite_pristylka, cena_dospely_pristylka,
cena_infant, typ_stravy, ID_rezervace, ico_ck)
values ('2014-09-02 00:00:00', '2014-09-09 00:00:00', 'transfer z hotelu
a zpět, pojištění ck proti úpadku, služby
delegáta',8,9499,6999,7999,0, 'NO',4,60192755)

insert into let (odlet, cislo_letu, terminal, trasa, cena,
ID_dopravni_spolecnosti, ID_rezervace, ID_typ_dopravy)
values ('2014-07-14 00:00:00', 'OK0526', 'Terminál 2', 'PRG-
PMI',2900,1,1,1 )
insert into let (odlet, cislo_letu, terminal, trasa, cena,
ID_dopravni_spolecnosti, ID_rezervace, ID_typ_dopravy)
values ('2014-07-21 00:00:00', 'OK0527', 'Terminál 2', 'PMI-
PRG',2900,1,1,1 )
insert into let (odlet, cislo_letu, terminal, trasa, cena,
ID_dopravni_spolecnosti, ID_rezervace, ID_typ_dopravy)
values ('2014-08-01 00:00:00', 'LS0826', 'Terminál 1', 'PRG-
BOJ',2300,2,2,1 )
insert into let (odlet, cislo_letu, terminal, trasa, cena,
ID_dopravni_spolecnosti, ID_rezervace, ID_typ_dopravy)

```

```

values ('2014-08-17 00:00:00','LS0827','Terminál 1','BOJ-
PRG',2300,2,2,1 )
insert into let (odlet, cislo_letu, terminal, trasa, cena,
ID_dopravni_spolecnosti, ID_rezervace, ID_typ_dopravy)
values ('2014-08-10 00:00:00','LH1398','Terminál 2','BRQ-
PMO',2100,3,3,1 )
insert into let (odlet, cislo_letu, terminal, trasa, cena,
ID_dopravni_spolecnosti, ID_rezervace, ID_typ_dopravy)
values ('2014-08-17 00:00:00','LH1399','Terminál 2','PMO-
BRQ',2100,3,3,1 )
insert into let (odlet, cislo_letu, terminal, trasa, cena,
ID_dopravni_spolecnosti, ID_rezervace, ID_typ_dopravy)
values ('2014-09-02 00:00:00','NH6198','Terminál 1','BRQ-
BOJ',2300,4,4,1 )
insert into let (odlet, cislo_letu, terminal, trasa, cena,
ID_dopravni_spolecnosti, ID_rezervace, ID_typ_dopravy)
values ('2014-09-09 00:00:00','NH6199','Terminál 1','BOJ-
BRQ',2300,4,4,1 )

insert into objednavka (ID_stav_objednavky, ID_prodejce, ID_zakaznik)
values (2,1,3)
insert into objednavka (ID_stav_objednavky, ID_prodejce, ID_zakaznik)
values (2,2,1)
insert into objednavka (ID_stav_objednavky, ID_prodejce, ID_zakaznik)
values (2,3,2)
insert into objednavka (ID_stav_objednavky, ID_prodejce, ID_zakaznik)
values (2,2,4)

insert into cestujici (rodne_cislo, jmeno, prijmeni, zkratka_zeme,
kod_pojisteni, ID_zajezd, ID_pokoj) values
(7712036622,'Patrik','Černý','CZE','2I',1,1)
insert into cestujici (rodne_cislo, jmeno, prijmeni, zkratka_zeme,
kod_pojisteni, ID_zajezd, ID_pokoj) values
(7805031749,'Simona','Černá','CZE','2I',1,1)
insert into cestujici (rodne_cislo, jmeno, prijmeni, zkratka_zeme,
kod_pojisteni, ID_zajezd, ID_pokoj) values
(8806074402,'Miroslav','Polák','CZE','2I',2,2)
insert into cestujici (rodne_cislo, jmeno, prijmeni, zkratka_zeme,
kod_pojisteni, ID_zajezd, ID_pokoj) values
(8503304022,'Lenka','Poláková','CZE','2I',2,2)
insert into cestujici (rodne_cislo, jmeno, prijmeni, zkratka_zeme,
kod_pojisteni, ID_zajezd, ID_pokoj) values
(6710011782,'Pavel','Bauer','CZE','NO',3,3)
insert into cestujici (rodne_cislo, jmeno, prijmeni, zkratka_zeme,
kod_pojisteni, ID_zajezd, ID_pokoj) values
(6504217732,'Pavčina','Bauerová','CZE','NO',3,3)
insert into cestujici (rodne_cislo, jmeno, prijmeni, zkratka_zeme,
kod_pojisteni, ID_zajezd, ID_pokoj) values
(9005204455,'Zdeněk','Síkora','CZE','NO',4,4)
insert into cestujici (rodne_cislo, jmeno, prijmeni, zkratka_zeme,
kod_pojisteni, ID_zajezd, ID_pokoj) values
(9107061213,'Jitka','Síkrová','CZE','NO',4,4)

insert into objednavka_zajezd (ID_objednavka, ID_zajezd) values (1,1)
insert into objednavka_zajezd (ID_objednavka, ID_zajezd) values (2,2)
insert into objednavka_zajezd (ID_objednavka, ID_zajezd) values (3,3)
insert into objednavka_zajezd (ID_objednavka, ID_zajezd) values (4,4)

insert into cestovni_kancelar_let (ico_ck, ID_let) values (45312974,1)

```

```

insert into cestovni_kancelar_let (ico_ck, ID_let) values (45312974,2)
insert into cestovni_kancelar_let (ico_ck, ID_let) values (26141647,3)
insert into cestovni_kancelar_let (ico_ck, ID_let) values (26141647,4)
insert into cestovni_kancelar_let (ico_ck, ID_let) values (27869237,5)
insert into cestovni_kancelar_let (ico_ck, ID_let) values (27869237,6)
insert into cestovni_kancelar_let (ico_ck, ID_let) values (60192755,7)
insert into cestovni_kancelar_let (ico_ck, ID_let) values (60192755,8)

insert into let_cestujici (ID_let, rodne_cislo) values (1,7712036622)
insert into let_cestujici (ID_let, rodne_cislo) values (1,7805031749)
insert into let_cestujici (ID_let, rodne_cislo) values (2,7712036622)
insert into let_cestujici (ID_let, rodne_cislo) values (2,7805031749)
insert into let_cestujici (ID_let, rodne_cislo) values (3,8806074402)
insert into let_cestujici (ID_let, rodne_cislo) values (3,8503304022)
insert into let_cestujici (ID_let, rodne_cislo) values (4,8806074402)
insert into let_cestujici (ID_let, rodne_cislo) values (4,8503304022)
insert into let_cestujici (ID_let, rodne_cislo) values (5,6710011782)
insert into let_cestujici (ID_let, rodne_cislo) values (5,6504217732)
insert into let_cestujici (ID_let, rodne_cislo) values (6,6710011782)
insert into let_cestujici (ID_let, rodne_cislo) values (6,6504217732)
insert into let_cestujici (ID_let, rodne_cislo) values (7,9005204455)
insert into let_cestujici (ID_let, rodne_cislo) values (7,9107061213)
insert into let_cestujici (ID_let, rodne_cislo) values (8,9005204455)
insert into let_cestujici (ID_let, rodne_cislo) values (8,9107061213)

insert into objednavka_parkovani (ID_objednavka, ID_parkovani) values
(1,1)
insert into objednavka_parkovani (ID_objednavka, ID_parkovani) values
(2,2)
insert into objednavka_parkovani (ID_objednavka, ID_parkovani) values
(3,3)
insert into objednavka_parkovani (ID_objednavka, ID_parkovani) values
(4,4)

insert into zajezd_pokoj (ID_zajezd, ID_pokoj) values (1,1)
insert into zajezd_pokoj (ID_zajezd, ID_pokoj) values (2,2)
insert into zajezd_pokoj (ID_zajezd, ID_pokoj) values (3,3)
insert into zajezd_pokoj (ID_zajezd, ID_pokoj) values (4,4)

insert into zajezd_let (ID_zajezd, ID_let, smer) values (1,1,'tam')
insert into zajezd_let (ID_zajezd, ID_let, smer) values (1,2,'zpet')
insert into zajezd_let (ID_zajezd, ID_let, smer) values (2,3,'tam')
insert into zajezd_let (ID_zajezd, ID_let, smer) values (2,4,'zpet')
insert into zajezd_let (ID_zajezd, ID_let, smer) values (3,5,'tam')
insert into zajezd_let (ID_zajezd, ID_let, smer) values (3,6,'zpet')
insert into zajezd_let (ID_zajezd, ID_let, smer) values (4,7,'tam')
insert into zajezd_let (ID_zajezd, ID_let, smer) values (4,8,'zpet')

insert into letiste_let (ID_letiste, ID_let, odlet_prilet) values
(1,1,'odlet')
insert into letiste_let (ID_letiste, ID_let, odlet_prilet) values
(2,1,'prilet')
insert into letiste_let (ID_letiste, ID_let, odlet_prilet) values
(2,2,'odlet')
insert into letiste_let (ID_letiste, ID_let, odlet_prilet) values
(1,2,'prilet')
insert into letiste_let (ID_letiste, ID_let, odlet_prilet) values
(1,3,'odlet')

```

```

insert into letiste_let (ID_letiste, ID_let, odlet_prilet) values
(3,3,'prilet')
insert into letiste_let (ID_letiste, ID_let, odlet_prilet) values
(3,4,'odlet')
insert into letiste_let (ID_letiste, ID_let, odlet_prilet) values
(1,4,'prilet')
insert into letiste_let (ID_letiste, ID_let, odlet_prilet) values
(5,5,'odlet')
insert into letiste_let (ID_letiste, ID_let, odlet_prilet) values
(4,5,'prilet')
insert into letiste_let (ID_letiste, ID_let, odlet_prilet) values
(4,6,'odlet')
insert into letiste_let (ID_letiste, ID_let, odlet_prilet) values
(5,6,'prilet')
insert into letiste_let (ID_letiste, ID_let, odlet_prilet) values
(5,7,'odlet')
insert into letiste_let (ID_letiste, ID_let, odlet_prilet) values
(3,7,'prilet')
insert into letiste_let (ID_letiste, ID_let, odlet_prilet) values
(3,8,'odlet')
insert into letiste_let (ID_letiste, ID_let, odlet_prilet) values
(5,8,'prilet')

```

```

create procedure novy_zakaznik
@jmeno varchar(20),
@prijmeni varchar(30),
@psc numeric(5),
@ulice varchar(50),
@zkratka_zeme varchar(3),
@telefon numeric(9),
@mail varchar(60)

as
begin
if not exists (select * from zákazník.telefon where @telefon =
zákazník.telefon)
    insert into zákazník (jmeno, prameni, psc, ulice, zkratka_zeme,
telefon, mail)
    values (@jmeno,@prijmeni,@psc, @ulice, @zkratka_zeme, @telefon,
@mail)
else
    print ('Zákazník s tímto telefonním číslem už je v databázi')
end

```

```

execute novy_zakaznik
'Tomáš', 'Jedno', 76005, 'Horáková', 'CZE', 732449331, 'tomas12@seznam.cz'

```

```

create procedure novy_let
@odlet smalldatetime,
@trasa varchar(25),
@cena int,
@ID_dopravni_spolecnosti int,
@ID_typ_dopravy int,
@ID_letiste_odlet int,

```

```

@ID_letiste_prilet int,
@ID_let int,
@odlet varchar (10),
@prilet varchar (10)

as
begin
insert into let (odlet, trasa, cena, ID_dopravni_spolecnosti,
ID_typ_dopravy)
values (@odlet, @trasa, @cena, @ID_dopravni_spolecnosti,
@ID_typ_dopravy)

insert into letiste_let (ID_letiste, ID_let, odlet_prilet)
values (@ID_letiste_odlet, @ID_let, @odlet)

insert into letiste_let (ID_letiste, ID_let, odlet_prilet)
values (@ID_letiste_prilet, @ID_let, @prilet)
end

execute novy_let
'2014-07-21 22:00:00', 'BRN - PMI', 4800, 2, 1, 1, 2, 9, 'odlet', 'prilet'

```

```

create view souhrn_objednavky as
select
zakaznik.jmeno, zakaznik.prijmeni, ubytovani.nazev,
zajezd.delka_zajezdu, prodejce.jmeno, prodejce.prijmeni,
typ_pokoje.nazev_typu
From zakaznik, objednavka, prodejce, objednavka_zajezd, zajezd,
zajezd_pokoj, pokoj, ubytovani, typ_pokoje
where zakaznik.ID_zakaznik = objednavka.ID_zakaznik and
prodejce.ID_prodejce = objednavka.ID_prodejce and
objednavka.ID_objednavka = objednavka_zajezd.ID_objednavka and
objednavka_zajezd.ID_zajezd = zajezd.ID_zajezd and zajezd.ID_zajezd =
zajezd_pokoj.ID_zajezd and zajezd_pokoj.ID_pokoj = pokoj.ID_pokoj and
pokoj.ID_ubytovani = ubytovani.ID_ubytovani and pokoj.ID_typ_pokoje =
typ_pokoje.ID_typ_pokoje

```