

**EKONOMICKÁ UNIVERZITA V BRATISLAVE**  
**FAKULTA HOSPODARSKEJ INFORMATIKY**

Evidenčné číslo: 103004/B/2025/36146835373117188

**POKROČILÉ VYUŽITIE DATABÁZ NA MANAŽOVANIE**  
**NEURČITÝCH ÚDAJOV**

Bakalárska práca

**EKONOMICKÁ UNIVERZITA V BRATISLAVE**  
**FAKULTA HOSPODARSKEJ INFORMATIKY**

**POKROČILÉ VYUŽITIE DATABÁZ NA MANAŽOVANIE**  
**NEURČITÝCH ÚDAJOV**

Bakalárska práca

**Študijný program:** Hospodárska Informatika  
**Študijný odbor:** Hospodárska Informatika  
**Školiace pracovisko:** Katedra aplikovanej informatiky  
**Vedúci záverečnej práce:** doc. Ing. Miroslav Hudec, Dr.

## Čestné vyhlásenie

Čestné vyhlasujem, že túto záverečnú prácu som vypracovala samostatne na základe získaných teoretických vedomostí a uviedla som všetku použitú literatúru.

V \_\_\_\_\_ dňa \_\_\_\_\_

.....  
Vlastnoručný podpis

## Abstrakt

SINOZHATSKA, Anzhelika-Viktoriia: *Pokročilé využitie databáz na manažovanie neurčitých údajov*. – Ekonomická univerzita v Bratislave. Fakulta Hospodárskej Informatiky; Katedra aplikovanej informatiky. – Vedúci záverečnej práce: doc. Ing. Miroslav Hudec, Dr. – Bratislava: FHI, 2025, 66 s.

Cieľom záverečnej práce je vytvorenie relačnej databázy pre účely cestovnej agentúry, ktorá umožňuje prácu s nepresnými údajmi, najmä so subjektívnymi požiadavkami zo strany klienta. Pozornosť je zameraná predovšetkým na spracovanie nepresných hodnôt v kategóriách cien a teplôt pomocou fuzzy logiky. Dáta použité na splnenie tejto úlohy sú ukázkové (najmä v oblasti osobných údajov), avšak maximálne priblížené reálnym podmienkam. Práca pozostáva zo štyroch kapitol, obsahuje 22 obrázkov a 7 tabuliek. V teoretickej časti je opísaný súčasný stav problematiky práce s fuzzy logikou, jej pôvod a možnosti riešenia na základe zahraničných a slovenských štúdií. Druhá kapitola jasne a stručne definuje ciele a podciele práce. Tretia kapitola sa venuje predstaveniu použitej metodiky vrátane SQL nástrojov na realizáciu databázy. Záverečná časť práce prezentuje dosiahnuté výsledky a predstavuje realizovanú prostredníctvom MySQL servera databázu, ktorá poskytuje kompletný prehľad údajov a tvorí škálovateľný základ pre potenciálnu databázu reálnej cestovnej agentúry s možnosťou uchovávanía fuzzy údajov.

**Kľúčové slova:** relačná databáza, subjektívne požiadavky, fuzzy logika, fuzzy údaje, spracovanie nepresných údajov, SQL, MySQL server

## **Abstract**

SINOZHATSKA, Anzhelika-Viktoriia: Advanced use of databases for managing uncertain data. – University of Economics in Bratislava. Faculty of Economic Informatics; Department of Applied Informatics. – Supervisor of the final thesis: doc. Ing. Miroslav Hudec, Dr. – Bratislava: FHI, 2025, 66 p.

The aim of the final thesis is to create a relational database for the purposes of a travel agency, which allows working with imprecise data, especially with subjective requirements from the client. The focus is primarily on processing imprecise values in the price and temperature categories using fuzzy logic. The data used to fulfill this task are exemplary (especially in the area of personal data), but maximally approximate to real conditions. The thesis consists of four chapters, contains 22 figures and 7 tables. The theoretical part describes the current state of the issue of working with fuzzy logic, its origin and possible solutions based on foreign and Slovak studies. The second chapter clearly and concisely defines the goals and sub-goals of the work. The third chapter is devoted to the presentation of the methodology used, including SQL tools for the implementation of the database. The final part of the work presents the achieved results and presents a database implemented via MySQL server, which provides a complete overview of data and forms a scalable basis for a potential database of a real travel agency with the possibility of storing fuzzy data.

**Keywords:** relational database, subjective requirements, fuzzy logic, fuzzy data, processing of imprecise data, SQL, MySQL server

# Obsah

Úvod .....	9
<b>1. Súčasný stav riešenej problematiky doma a v zahraničí .....</b>	<b>10</b>
1.1 Základne pojmy.....	10
1.1.1 Charakteristika nepresných údajov .....	10
1.1.2 Relačná databáza a jej využitie .....	11
1.1.3 Výskum v oblasti fuzzy databáz .....	15
1.1.4 Aplikácia v praxi.....	16
1.2 Spracovanie neurčitých údajov pomocou fuzzy logiky .....	17
1.2.1 Pojem fuzzy logika .....	17
1.2.2 Operácie s fuzzy množinami.....	18
1.3 Vloženie nepresných dát do databázových systémov.....	21
1.3.1 Štandardne SQL nástroje .....	21
1.3.2 Rozšírenia pre fuzzy prácu s dátami .....	21
1.3.3 Výhody a nevýhody rožných prístupov .....	22
1.4 Zhrnutie kapitoly.....	24
<b>2 Cieľ práce.....</b>	<b>25</b>
2.1 Stanovenie hlavného cieľa práce.....	25
2.2 Špecifikácia čiastkových cieľov .....	25
<b>3 Metodika práce a metódy skúmania.....</b>	<b>27</b>
3.1 Funkcie príslušnosti.....	27
3.2 Entitno-relačný diagram a normalizácia databázy .....	30
3.3 Algoritmus zavedenia neurčitosti do relačného modelu.....	31
3.4 Postup tvorby databázy pomocou SQL v prostredí MySQL.....	33
<b>4 Výsledky práce a diskusia.....</b>	<b>34</b>
4.1 Získavanie a úprava údajov.....	34
4.1.1 Typy použitých údajov.....	35
4.1.2 Výpočet fuzzy príslušnosti.....	35
4.2 Návrh databázovej štruktúry. ER diagram .....	41
4.3 Implementácia a testovanie databázy.....	50
4.3.1 Implementácia na MySQL serveri. Logický model databázy.....	50
4.3.2 Výsledky dotazovania.....	52
4.3.3 Manipulácie s fuzzy údajmi .....	54
4.4 Simulácia výberu zájazdu pre klienta .....	56
4.5 Diskusia a možnosti zlepšenia .....	60
<b>Záver .....</b>	<b>62</b>
<b>Zoznam použitej literatúry .....</b>	<b>63</b>

## Zoznam tabuliek

Tabuľka 1. Porovnanie databáz .....	13
Tabuľka 2. FSQL funkcie pre praktické použitie .....	22
Tabuľka 3. Klasické vs Fuzzy riešenie .....	23
Tabuľka 4. Zoznam teplôt 2024 .....	36
Tabuľka 5. Trojuholníková funkcia príslušnosti .....	37
Tabuľka 6. Zoznam cien .....	39
Tabuľka 7. Lichobežníková funkcia príslušnosti .....	40

## Zoznam obrázkov

Obrázok 1. Entitno-relačný diagram "Pes" .....	12
Obrázok 2. Rebríček databáz 2025 .....	14
Obrázok 3 (a,b). Fuzzy Teplota .....	20
Obrázok 4. Trojuholníková fuzzy funkcia: a) všeobecná b) symetrická. ....	27
Obrázok 5. Lichobežníková fuzzy funkcia .....	28
Obrázok 6. Lingvistická premenná .....	29
Obrázok 7. Entitno-relačný diagram .....	30
Obrázok 8. ER diagram .....	41
Obrázok 9. Schéma 1 .....	43
Obrázok 10. Schéma 2 .....	45
Obrázok 11. Schéma 3 .....	46
Obrázok 12. Schéma 4 .....	48
Obrázok 13. Schéma 5 .....	49
Obrázok 14. Logický dizajn databázy .....	51
Obrázok 15. Dotaz 1 .....	53
Obrázok 16. Dotaz 2 .....	54
Obrázok 17. Dotaz 3 .....	55
Obrázok 18. Zoznam hotelov .....	56
Obrázok 19. Zoznam hotelov podľa teploty .....	57
Obrázok 20. Zoznam hotelov podľa teploty a ceny .....	58
Obrázok 21. Zoznam izieb .....	59
Obrázok 22. . Evidencia nového klienta .....	60

## Zoznam skratiek (Slovník)

CAD/CAM	Počítačom podporované navrhovanie a výroba
CMS	System na správu obsahu
CODASYL	Konferencia o jazykoch pre dátové systémy
CRUD	Skratka pre základné operácie s dátami v databázach
CSV	Hodnoty oddelené čiarkou
CRM	Riadenie vzťahov so zákazníkmi
DB	Databáza
DBMS	System na správu databáz
DEA	Analýza obalu dát
DNS	System doménových mien
ERM	Modelovanie entít a vzťahov
ERP	Plánovanie podnikových zdrojov
GEFRED	Všeobecný rámec pre fuzzy relačné databázy
IBM IMS	Informačný systém IBM
ID	Identifikátor
IDMS	Integrovaný databázový systém
IDEF1X	Štandard pre modelovanie informácií
LDAP	Lahký protokol prístupu k adresáru
MAE	Priemerná absolútna chyba
MHz	Mega hertz, jednotka frekvencie.
MS	Microsoft
NF	Normálna forma
SQL	Štruktúrovaný dotazovací jazyk

## Úvod

S rastúcim počtom produktov a služieb sa kľúčovými ukazovateľmi spokojnosti, ak pre spotrebiteľa, tak aj pre dodávateľa, stávajú rýchlosť a kvalita poskytovaných služieb. Pre klienta je v podmienkach širokej dostupnosti technológií prioritou minimalizovať čas potrebný na získanie požadovaného produktu alebo služby. Na druhej strane, pre poskytovateľa je dôležité zabezpečiť rýchlu a kvalitnú reakciu na požiadavky klienta, aby si udržal konkurencieschopnosť na trhu. Aby možnosť efektívne reagovať na požiadavky klientov existovala, je nevyhnutné mať k dispozícii všetky potrebné dáta na vyhľadanie najvhodnejšieho produktu alebo riešenia. Preto je potrebné zabezpečiť kvalitne ukladanie a uchovávanie údajov na organizačnej úrovni, aby bol možný rýchly prístup k nim. Najklasickejším a najrozšírenejším spôsobom ukladania dát je relačná databáza. Ale čo však nastane, keď dáta nie je možné uložiť v databáze bez porušenia hlavných pravidiel relačných modelov na zabezpečenie konzistencie, správnosti a spoľahlivosti? V tejto práci sme sa zaoberali problémom neurčitých údajov, ktoré porušujú klasický prístup k uchovávaniu v a možnosťami ich úpravy pre vhodné uloženie v databáze. Problém nedostatočne modernizovaných systémov je obzvlášť výrazný v oblasti cestovných agentúr, kde sa konzultanti často stretávajú s opisom požiadaviek od zákazníkov ako napríklad: „teplejšie“, „mierne drahé“. Na základe štandardov relačných modelov nie je možné slovné opisy alebo rozmedzia uchovávať v databáze. Takéto formulácie sú subjektívne a závisia od individuálneho vnímania každého človeka. Táto bakalárska práca sa venuje problému nejednoznačnosti vo formuláciách klientov, ktoré sa prejavujú vo forme nepresného (fuzzy) zadania údajov. Takéto požiadavky komplikujú výber vhodného produktu alebo služby a spôsobujú časové straty na strane klienta aj poskytovateľa. V práci opisali sme algoritmus na adaptáciu neurčitých typov údajov a ich správne uloženie. Keďže najrozšírenejším spôsobom uchovávanie veľkého množstva údajov je databázový prístup, rozhodli sme sa vytvoriť ukážkový model databázy pre cestovnú agentúru. Cieľom práce bolo navrhnúť riešenie spracovateľského problému pomocou relačnej databázy s prvkami fuzzy logiky. Pomocou fuzzy algoritmov vytvorili sme databázy, ktorá dokáže spracovať a uchovávať nepresné vstupy bez narušenia integrity a jednoznačnosti štruktúry systému. Takýto prístup umožní spracúvať nejasné požiadavky klientov presnejšie a rýchlejšie, čo zvýši celkovú efektivitu poskytovaných služieb.

# 1. Súčasný stav riešenej problematiky doma a v zahraničí

V tejto kapitole sme opísali aktuálny stav problematiky v oblasti relačných databáz a aplikácie fuzzy logiky. Pri štúdiu tejto témy bola využitá výhradne odbornú literatúru a kvalitné zdroje.

## 1.1 Základne pojmy

Na podrobné preskúmanie databáz s integráciou nepresnej logiky sme najskôr definovali pojem nepresný údaj, nasledoval konkrétny účel využitia relačných modelov dátových systémov, oblasti a dôvody uplatnenia fuzzy logiky v relačných databázach

### 1.1.1 Charakteristika nepresných údajov

Podľa Ma a Yan v databázach sa môžu vyskytnúť päť typov nepresných údajov: nekonzistentné, neprenosné, vágne, nejasné a nejednoznačné.

1. Nekonzistentnosť: vyskytuje sa, keď popis jedeného objektu je v konflikte s jeho istotou (popis objektu podľa dvoch respondentov).
2. Nepresnosť: nepresnosť sa týka množstva údajov, ktoré sa nachádzajú v rozpätí alebo v rozsahu (od 12 po 15).
3. Vágnosť: sa týka údajov, ktoré boli popísané podľa lingvistických termínov („veľký“, „malý“).
4. Nejasnosť: vzťahuje sa na istotu alebo spoľahlivosť daného počtu údajov, ktoré možno použiť na predpovedanie alebo predikciu budúcich dáta.
5. Nejednoznačnosť: k tomu dochádza, keď údaje nie sú jasne definované (2010).

V reálnom svete existuje veľa typov takýchto údajov, najmä ak údaje pochádzajú od ľudí (napríklad lekárov, inžinierov) alebo senzorov a modelov. Tento jav nastáva preto, že v pracovnom prostredí týchto oblastí je prakticky nemožné popísať údaje presným, číselným spôsobom. Slovné opisy príznakov chorôb a interpretácia obrazových dát (napríklad RTG snímok alebo MRI) môžu byť nepresné alebo vágne. Rovnako aj v inžinierskej praxi sa pri odhadoch rizík alebo pravdepodobností porúch často vychádza zo skúseností, a teda údaje nemusia byť úplne presné. Podobne aj ekonomické a štatistické prognózy nereprezentujú úplnú presnosť nameraných výsledkov, pretože vždy obsahujú určitý stupeň neistoty a odchýlok (Rajeev, 2015). Wang a Liu naznačovali :”K chybám údajov dochádza, keď nie

je známa presná hodnota charakteristiky, napríklad keď taktovanie počítača je v rozsahu 25 MHz až 33 MHz”. K nekonzistentnosti údajov môže viesť nesprávnosť výpočtových systémov alebo ľudská chyba. Poruchy hardvéru môžu poškodiť alebo mierne zmeniť hodnoty údajov. Chyby pri písaní výpočtových algoritmov vedú k priamym nepresnostiam v získaných výsledkoch. Nepresné zadanie formátov údajov, napríklad použitie číselného formátu namiesto formátu dátumu, môže viesť k deformácii údajov a následnej nekonzistentnosti. Súčasne platí, že pri absencii údajov v relačných modeloch databáz je vhodnejšie nenechávať prázdne bunky bez spracovania, ale priradiť ich k špeciálnemu typu údajov, ktorý tiež je priradený do nepresných variácií. Autori v svojej práci vysvetlili, že: “Chýbajúce hodnoty možno považovať za neplatné varianty údajov”, čo sa týka nulových hodnôt atribútov v databázach (2009). Na základe štúdií zameraných na translačný výskum a oblasť onkológie na Slovensku sa ukázalo, že pri zbere údajov priamo od pacientov sa často vyskytujú výrazy vyjadrujúce mieru vágnosti alebo nepresnosti, ako napríklad: „mierna bolesť“ či „časté bolesti hlavy“. Aj keď údaje pochádzajú z dotazníkov, laboratórnych výsledkov alebo senzorov, často obsahujú rôzne úrovne neistoty. (Mego, Svetlovská, Mardiak, 2014).

### 1.1.2 Relačná databáza a jej využitie

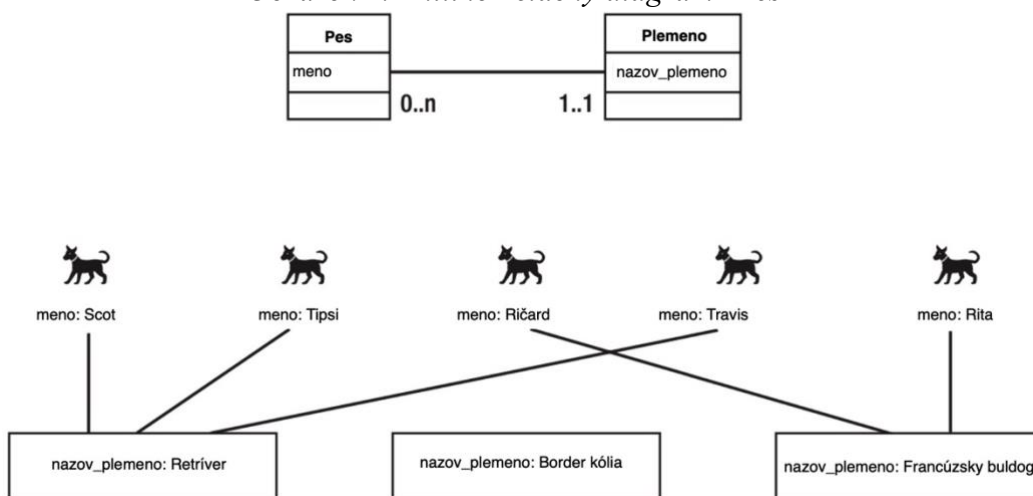
Vzhľadom na rôznorodosť prístupov a kontextov použitia v súčasnosti neexistuje všeobecne akceptovaná definícia pojmu „databáza“. Väčšina výrazov vo všeobecnosti popisuje databázu ako „zbierku informácií“. Preto, pre komplexné pochopenie databáz je dôležité oboznámiť sa s rôznymi materiálmi od kľúčových autorov. Podľa Connollyho a Begga je databáza „kolekciou logicky prepojených údajov, určenou na uspokojenie informačných potrieb organizácie“ (2004). Elmarsi a Navathe popisali databázu nasledujúce: „Zbierka súvisiacich údajov. Údajmi sú známe fakty, ktoré môžu byť zaznamenané a majú implicitný význam“ (2016). Garcia-Molina, Ullman a Windom definovali databázu takto: „V podstate je databáza ničím iným ako zbierkou informácií, ktoré existujú po dlhé obdobie, často mnoho rokov. V bežnej reči termín databáza označuje zbierku údajov, ktoré sú spravované systémom na manažovanie databáz (DBMS)“ (2008).

Existuje obrovské množstvo databáz, v ktorých sú uložené tisíce jednotiek údajov. S rozširovaním a rastom objemu dát sa zároveň vyvíjali a formulovali rôzne dátové modely. Tieto modely sa medzi sebou odlišujú najmä z dôvodu:

- rôznych typov údajov a dátových štruktúr,
- odlišných požiadaviek na výkon a rýchlosť spracovania,
- špecializácie na konkrétne typy úloh a aplikácií.

Relačný dátový model po prvýkrát bol predstavený v roku 1970. Tento model predstavuje základ pre bezpečné a konzistentné ukladanie údajov, pričom eliminuje redundanciu. Zároveň zavádza logické obmedzenia pre existujúce systémy s formátovanými údajmi. Z matematického hľadiska je definovaná relácia  $R$  ako podmnožina Karteziánskeho súčinu  $R = S_1 \times S_2 \times \dots \times S_n$ , kde:  $S_1$  až  $S_n$  - zadané množiny; domény, ktoré zodpovedajú stĺpcom v tabuľke.  $R$  - relácia (Codd, 1970). Perfektný relačný model popísali Connolly a Begg: „V relačnom modeli sú všetky údaje logicky štruktúrované v reláciách (tabuľkách). Každá relácia má svoj názov a pozostáva z atribútov (stĺpcov), ktoré majú pomenované dátové typy. Každá n-tica (riadok) obsahuje jednu hodnotu pre každý atribút“ (2015). Veľkou výhodou relačného modelu je práve jeho jednoduchá a prehľadná logická štruktúra. Na *Obrázku 1* je uvedený diagram relačného modelu „Pes“. Vzťah medzi entitami Pes a Plemeno je typu 1:N (jeden na mnoho): jeden záznam v tabuľke Plemeno sa môže viazať na viacero záznamov v tabuľke Pes, Každý Pes má presne jedno Plemeno. Tento diagram znázorňuje entitno-relačný model (ERM) pre vzťah medzi entitami Pes a Plemeno, kde: Pes má atribút meno, Plemeno má atribút nazov\_plemeno, Každý pes patrí práve k jednému plemenu (kardinalita 1..1), jedno plemeno môže mať viacero psov (kardinalita 0..n).

Obrázok 1. Entitno-relačný diagram "Pes"



Zdroj: Churcher (2007)

*Tabuľka 1* opisuje jednotlivé databázové modely podľa spôsobu organizácie a možností ukladania dát, a to na základe teoretických prístupov známych autorov a odborníkov: relačný model podľa E. F. Codd (1970), hierarchický model podľa Silberschatza a Kedema (1980), sieťový model vyvinutý kolektívom CODASYL (1976), objektovo-orientovaný model podľa Wona Kima (1990), kľúč-hodnotový model podľa Decandiu a kol. (2007), grafový model podľa Fegarasa a Maiera (1995), dokumentovo-orientovaný model podľa Chodorow (2019). V stĺpci „Flexibilita“ sa vysvetľuje do akej miery je možné jednoducho meniť údaje uložené v databáze, bez porušenia pravidiel systému. Stĺpec „Podpora vzťahov“ objasňuje, ako rýchlo a presne dokáže daný model identifikovať priame vzťahy medzi uloženými údajmi. Zároveň, v tabuľke definovane možné praktické využitia rôznych modelov v reálnom svete, ako aj softvérové riešenia, pomocou ktorých je možné dané databázové modely implementovať.

*Tabuľka 1. Porovnanie databáz*

<b>Databázový model</b>	<b>Štruktúra dát</b>	<b>Flexibilita</b>	<b>Podpora vzťahov</b>	<b>Použitie</b>	<b>Priklady systémov</b>
Relačný	Tabuľky (relácie)	Stredná až vysoká	Silná	Obchodne aplikácie, banky, ERP	MySQL, PostgreSQL, Oracle
Hierarchický	Strom	Nízka	Jeden-na-viac	Organizácia súborových systémov; Pripojenie DNS a LDAP.	IBM IMS
Sieťový	Graf (s viacerými vzťahmi)	Stredná	Viac-na-viac	Historické systémy	IDMS
Objektovo-orientovaný	Objekty (triedy)	Stredná až vysoká	Silná (cez asociácie)	Multimédia, CAD/CAM	Db4o, ObjectDB
Dokumentovo-orientovaný	Dokumenty (JSON, XML, BSON)	Vysoká	Obmedzená	Webové aplikácie, CMS	MongoDB, CouchDB
Kľúč-hodnota	Páry kľúč-hodnota	Vysoká	Žiadna	Cache, real-time aplikácie	Redis, Memcached
Grafový	Uzly a hrany grafu	Vysoká	Veľmi silná	Sociálne siete, finančne prostredie	Neo4j, JanusGraph

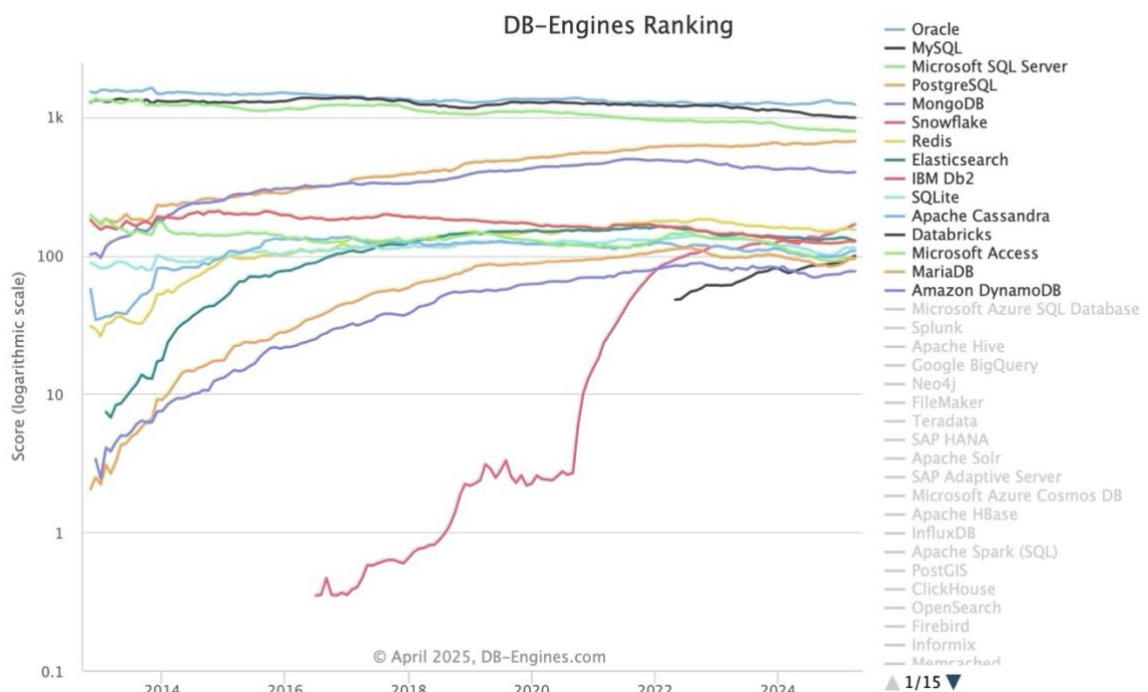
*Zdroj: Vlastne spracovanie podľa Codd (1970), Silberschatza a Kedema (1980), CODASYL (1976), Kima (1990), Decandiu a kol. (2007), Fegarasa a Maiera (1995), Chodorow (2019)*

Ako uvádza tabuľka, relačný databázový model má odlišnú štruktúru uchovávaných v ňom dát od iných databázových modelov. Najviac podobné štruktúre dát relačného modelu sú hierarchické a sieťové modely, ale majú odlišné pravidlá pre vytváranie vzťahov medzi

uloženými objektmi. Relačný model poskytuje vysokú presnosť v štruktúre údajov, ale obmedzenú flexibilitu pri práci s neštruktúrovanými alebo hierarchickými údajmi. Pretože určuje konkrétne pravidlá pre vkladanie jednotlivých údajov a ich zmena je zakázaná pre definované dátové formáty v rozpracovanom systéme, lebo to vedie k porušeniu referenčnej integrity. Model má silnú podporu vzťahov vďaka primárnym a cudzím kľúčom a možnosti logicky prepájať objekty medzi entitnými množinami bez množstevných obmedzení vo forme rôznych kardinalít (napr. 1:N, M:N), čo umožňuje flexibilné modelovanie väzieb medzi údajmi.

Na základe prieskumu, ktorý uskutočnila britská spoločnosť Redgate Software v apríli 2025, sa zistilo, že relačné databázy momentálne obsadzujú prvé tri miesta na trhu spomedzi 424 systémov v rebríčku. Tento prieskum bol realizovaný pomocou kombinovaného hodnotenia viacerých ukazovateľov, ako sú počet vyhľadávaní, diskusie na odborných fórach, zmienky v pracovných ponukách či prítomnosť na sociálnych sieťach: „Hodnota obľúbenosti systému vypočítaná štandardizáciou a spriemerovaním jednotlivých parametrov. Tieto matematické transformácie sú robené tak, aby bola zachovaná vzdialenosť jednotlivých systémov“ (DB-Engines, 2025). To naznačuje, že relačný dátový model je aj naďalej veľmi žiadaný a široko používaný v praxi. Keďže sme sa v našej práci rozhodli práve pre tento typ modelu, ďalej je uvedený podrobnejší popis relačného modelu.

Obrázok 2. Rebríček databáz 2025



Zdroj: Redgate Software. (2025, April)

### 1.1.3 Výskum v oblasti fuzzy databáz

Hlavnou odlišnosťou relačných databáz od fuzzy databáz je prítomnosť hodnotenia (grade value). Vo svojej knihe autori Galindo, Urrutia a Piattini vysvetlili, že pre bežnú formu fuzzy relačnej databázy je charakteristická existencia hodnoty na intervale  $[0,1]$  pre každý záznam. Práve na tomto princípe je postavená fuzzy databáza, pričom hodnotenie je abstraktné a závisí od preferencií tvorcu relácie (2005). Autori zároveň tvrdia: „Napriek tomu sémantika, ktorá je tejto hodnote pridelená, určuje jej užitočnosť a táto hodnota bude použitá v dotazovacích procesoch“ (2005). Charakteristickou výhodou fuzzy databáz je ich schopnosť realistickejšie popísať bežné problémy vďaka štandardizovanému „uchovávaní“ fuzzy údajov. „Teória fuzzy množín zároveň umožňuje používať jazykové údaje priamo v modeli DEA,“ potvrdzuje autor Lertworasirikul (2022). Vo svojej štúdiu opisuje vplyv fuzzy logiky na systém Dynamics Analysis Environment, čo je softvérové prostredie určené na dynamickú analýzu údajov. Táto technológia sa často využíva v oblasti kybernetickej bezpečnosti.

Významný výskumník Zongmin Ma vo svojej práci venovanej štúdiu fuzzy údajov v inžinierstve uvádza: „Fuzzy entity inštalácie vo fuzzy modeli IDEF1X môžu byť reprezentované ako záznamy so stupňom príslušnosti vo fuzzy relačných databázach. A fuzzy atribútové hodnoty vo fuzzy modeli IDEF1X sa môžu stať fuzzy atribútovými hodnotami záznamov vo fuzzy relačných databázach“ (2006). Tento model slúži na dokumentovanie a návrh relačných databáz. Pomocou grafov model prezentuje štruktúru a sémantiku informácií v podniku.

Čo sa týka využitia fuzzy logiky v programovaní odpovedej umelej inteligencie, výskumník v oblasti fuzzy logiky z univerzity Alberta vo svojom učení napísal: „Riešenia a nástroje umelej inteligencie sa naďalej rozširujú v ekonomike v rôznych sektoroch, nakoľko sa rozširujú aj možnosti programovania fuzzy logiky“ (Scott, 2023). To opisuje schopnosť umelej inteligencie ľahko spracovávať používateľské dotazy aj pri nepresnom vstupe, napríklad pri jazykovom popise hodnôt („pekný“, „vysoký“, „veľký“), približných údajoch (napr. „cca 17“) alebo nepresne zadaných údajoch (napr. nesprávne meno autora/výskumníka). V štúdiu Viscontiho, Andersona a Kerleyho sa opisuje transformácia fuzzy vstupov pre Quixel Megascans (2023). Ide o rozsiahlu knižnicu fotorealistických 3D skenerov. Tento skener sa často používa pri vývoji videohier a architektonickej vizualizácii. Používateľ môže zadávať hodnoty atribútov ako napríklad „Kvalita“ alebo „Kontrast“. Systém automaticky prevedie nepresnú hodnotu na fuzzy číslo so stupňom príslušnosti.

„Súčasný systém podporuje iba základnú fuzzy množinu, kde zvolené hodnoty majú stupeň príslušnosti 1,0 a všetky ostatné 0,0. Napríklad výber „vysoká“ pre „Kvalitu“ sa prevedie na fuzzy množinu {nízka: 0,0, stredná: 0,0, vysoká: 1,0}“ (2023).

#### 1.1.4 Aplikácia v praxi

V praxe, zameranej na zlepšenie riadenia CRM systémov sa dokázalo, že v súčasnosti je konkurencia na vysokej úrovni, a preto je dôležité si udržať každého zákazníka. Taktiež sa zdôrazňuje, že v oblasti riadenia databáz zákazníkov nastal veľký prelom a analýza údajov sa v súčasnosti nachádza vo fáze každodenného technologického rozvoja. V tejto štúdií je prítomná metodológia rozvoja CRM prostredníctvom zavedenia fuzzy logiky, aby sa udržali existujúci zákazníci a pritiahli noví. Navrhovaný model je schopný automaticky reagovať na neistoty v mnohých záznamoch a vytvárať asociačne fuzzy pravidla, ktoré nevyžadujú presnosť ani spoľahlivosť. Tým sa znižuje počet duplicitných pravidiel, čo uľahčuje interpretáciu výsledkov a prijímanie rozhodnutí. Táto štúdia teda ukazuje, že fuzzy logika a genetické algoritmy môžu hrať významnú úlohu pri zlepšovaní vzťahov so zákazníkmi z hľadiska lojality a záväzku. (Abu-Al Dahab, Haggag2, Fotouh, 2023).

V rámci rozvoja bankových CRM systémov Bojanowska a Kulisz vytvorili model v Simulinku s použitím fuzzy algoritmu Mamdani: „Ide o rozhodovací model, ktorý poskytuje výsledok vo forme binárnej hodnoty 0 alebo 1, kde 1 znamená, že sa oplatí investovať do zákazníka, zatiaľ čo 0 znamená, že sa neoplatí“ (2023). Autorky zdôrazňujú, že pre bankové CRM systémy je dôležité nielen správne uchovávanie údajov, ale aj rozvoj modernizačných algoritmov, v čom pomáha fuzzy logika. Vďaka algoritmu Mamdani, ktorý využíva pravidlá IF-THEN a algoritmu defuzzifikácie, je možné získať priamočiare hodnoty skóre. Autorky ako príklad uvádzajú pravidlo, podľa ktorého ak je riziko nízke a lojalita vysoká, potom je hodnota zákazníka vysoká. (2023)

Praca týkajúca sa priemyselného sektora, konkrétne výroby kartónu čínskej produkcie, poukazuje na pozitívny vplyv fuzzy logiky na relačné databázy. Autori Jimenez, Urrutia, Galindo, Zaraté opisujú atribúty neostrej modelácie údajov v rámci modelu GEFRED (všeobecný model pre neostrú relačnú databázu), ich implementáciu v bežnej relačnej databáze. Táto štúdia je priamo spojená so spoločnosťou, ktorá sa zameriava na výrobu kartónu v regióne Maule. Osobitná pozornosť bola venovaná úrovni kontroly kvality hotových výrobkov, kde sa okrem cieľových vizuálnych charakteristík zohľadňovali aj

subjektívne (ľudské) testy získané v dôsledku vizuálnej kontroly a taktilných vnemov operátora. Takéto údaje nie je možné presne opísať číselnými metódami, preto využitie fuzzy logiky pomohlo zlepšiť kontrolu kvality (2025).

Farokhi, Vahid, Nilashi a Ibrahim reprezentovali TripAdvisor dataset realizovaný na základe metódy fuzzy zhlukovania s použitím C-means algoritmu. Ako evaluačná metrika bola v tejto práci použitá Mean Absolute Error (MAE), ktorá poukázala na zvýšenie presnosti navrhovanej metódy. Táto štúdia ukázala, že využitie fuzzy logiky zlepšilo metódu vyhľadávania turistických destinácií a vďaka MAE sa zistilo, že táto metóda nebola horšia ako klasický spôsob vyhľadávania. „Keďže navrhovaný odporúčací systém zvyšuje presnosť techník kolaboratívneho filtrovania, bude tento systém sľubnou metódou odporúčaní pre úlohy odporúčania položiek v oblasti cestovného ruchu.“, - naznačujú autori (2016).

Pomocou fuzzy logiky Takumi a Tachibana vytvorili aplikáciu pre Android smartfóny na podporu Hiroshima Tourist Information. Táto aplikácia dokáže „spracovávať“ ľudské emócie a vytvárať zoznam turistických odporúčaní na ich základe. Výhodou tejto technológie je bezmocnosť počítačových systémov a výpočtových strojov pred definíciou ľudského pocitu (2018).

## 1.2 Spracovanie neurčitých údajov pomocou fuzzy logiky

V danej podkapitole opísali sme, na čom je založená fuzzy logika a ako veľmi sa líši od klasickej logiky. Taktiež sme uviedli príklady možných manipulácií s fuzzy množinami. Pre databázové prostredie sme definovali najvhodnejšie algoritmy a funkcie štruktúrovaného dopytovacieho jazyka SQL.

### 1.2.1 Pojem fuzzy logika

Ako uviedol zakladateľ teórie fuzzy množín, americký matematik a profesor Lotfi Zadeh: „Fuzzy logika je presná logika umožňujúca pracovať s nepresnými a približnými úvahami“. Fuzzy logika je forma logiky, ktorá môže byť pravdivá pre akúkoľvek hodnotu medzi 0 a 1. Používa sa na opis čiastočnej pravdy, teda výrokov, ktoré nie sú pravdivé alebo nepravdivé v plnej miere. Na rozdiel od racionálneho myslenia, kde pravda môže byť 0 alebo 1, fuzzy logika umožňuje rozšíriť iracionalitu (1965).

Dôvod zmätku je založený na myšlienke, že ľudia sa rozhodujú na základe tvrdých, nekvantifikovateľných informácií. Preto v reálnom svete chyba vyhodnocovanie na základe

ľudských posudzovaní. Fuzzy množiny a modely sú užitočné nástroje matematického pôvodu na popis neurčitých a neistých údajov. Tieto metódy umožňujú používateľovi spravovať, skúmať a používať fuzzy údaje, ktoré boli predtým nedostupné pre bežne systémy. Fuzzy logika našla uplatnenie v rôznych oblastiach, od riadiacich systémov až po umelú inteligenciu (Ghahnavieh, Manesh, Sheikmoradi, 2024).

Často sa vo fuzzy logike využíva lingvistická premenná. Je to opis predmetu alebo stavu v slovnej formulácii. Taktiež, bežne sa používa miera alebo intenzita, ktorá vyjadruje stupeň daného javu. Napríklad „veľmi“, „mierne“, „malo“. Tieto výrazy umožňujú presnejšie modelovať neurčité alebo vágne informácie. Lotfi Zadeh to popísal premennú nasledujúce: „Jazykovou premennou rozumieme premennú, ktorej hodnotami sú slová alebo vety v prirodzenom alebo umelom jazyku. Napríklad vek je lingvistickou premennou, ak sú jej hodnoty skôr jazykové ako číselné, t. j. mladý, nie mladý, veľmi mladý, celkom mladý, starý, nie veľmi starý a nie veľmi mladý atď., a nie 20, 21, 22, 23“ (1975).

Fuzzy logika sa využíva v priemyselnom riadení, robotike, prognózovaní a modelovaní. Je to odvetvie viac-hodnotovej logiky, ktoré je určené na spracovanie nepresných dát charakterizovaných lingvistickou neurčitou. Hodnoty klasickej logiky 0 alebo 1 nestačia na riešenie mnohých úloh v reálnom živote. Riadenie procesov, ktoré sa objavilo počas vývoja elektronických zariadení, pripravilo cestu pre moderný rozvoj teórie fuzzy množín. V kontexte fuzzy logiky neexistuje jednoduchý rozdiel medzi pravdou a nepravdou. V skutočnosti je medzi nimi možná akákoľvek hodnota (Dezhic, 2018).

### 1.2.2 Operácie s fuzzy množinami

Pojem fuzzy množín je odvodený zo zovšeobecnenia tradičnej predstavy o podmnožinách, čo umožňuje zahrnutie vágnych a neistých pojmov. Dodatočné členstvo v množine sa berie ako otázka, kedy klasické množiny patria do primárnych množín alebo nie. Pri niektorých položkách nemusí byť jasné, či sú súčasťou zbierky alebo nie. Na prekonanie tejto neistoty je každej zložke pridelená členská hodnota, ktorá označuje, aká silná je v asociácii. Tento stupeň je vyjadrený v hodnotách na intervale [0,1].

*Definícia 1:* Fuzzy množina  $A$  nad vesmírom konečného alebo nekonečného intervalu  $X$  (v rámci ktorého fuzzy množina môže nadobúdať hodnotu) je množinou párov:

$$A = \left\{ \frac{\mu_A(x)}{x} \mid x \in X, \mu_A \in [0,1] \in \mathfrak{R} \right\}$$

Kde:

$\mu_A(x)$  stupeň členstva prvku  $x$  do fuzzy množiny  $A$ . Tento stupeň sa pohybuje medzi extrémami 0 a 1.

$\mu_A(x) = 0$  znamená, že  $x$  v žiadnom prípade nepatrí do fuzzy množiny  $A$

$\mu_A(x) = 1$  označuje že  $x$  úplne patrí do fuzzy množiny  $A$ .

$\mu_A(x) = 0,5$  je bod najväčšej neistoty.

$X$  je základná doména a vo všeobecnejšej oblasti fuzzy množinu

$A$  je funkcia  $\mu_A$  ktorý zodpovedá každému prvku vesmíru  $X$  s jeho stupňom členstva k  $A$ :

$$\mu_A(x): X \rightarrow [0,1]$$

Vesmír diskurzu  $X$  alebo súbor množina hodnoty, môžu byť týchto dvoch typov:

1. Konečný alebo diskretný vesmír diskurzu  $X = \{x_1, x_2, \dots, x_n\}$ , kde fuzzy množina

$A$  môže byť reprezentovaný ako:

$$A = \frac{\mu_1}{x_1} + \frac{\mu_2}{x_2} + \dots + \mu_n/x_n$$

kde  $\mu_i$  s  $i = 1, 2, \dots, n$  predstavuje stupeň členstva prvku  $x_i$ . kde  $\mu_i$  s  $i = 1, 2, \dots, n$  predstavuje stupeň príslušnosti prvku  $x_i$ . Tu “+“ nemá rovnaký význam ako v aritmetickom súčte, ale má skôr význam agregácie a neznamená rozdelenie, ale skôr spojenie oboch hodnôt.

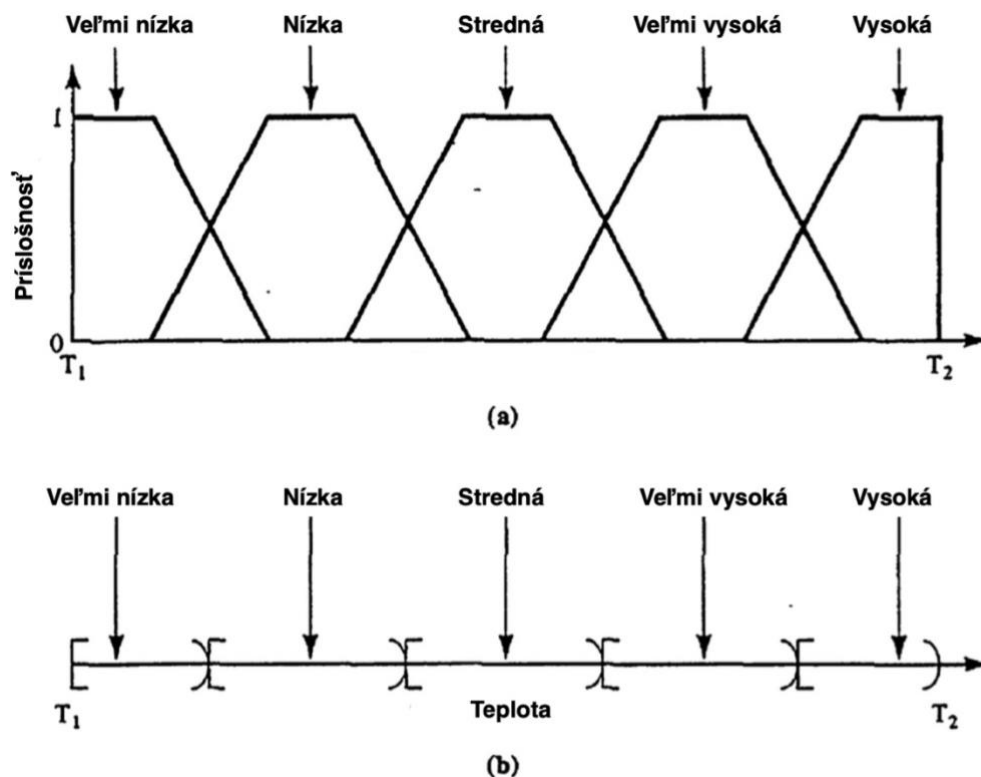
2. Nekonečný vesmír diskurzu:

$$A = \int \mu_A(x)$$

Funkcia členstva  $\mu_A(x)$  fuzzy množiny  $A$  vyjadruje mieru, v ktorej  $x$  overuje kategóriu špecifikovanú  $A$  (Galindo, 2008).

Na opis hodnôt fuzzy premennej sa bežne využívajú viaceré fuzzy množiny, ktoré reprezentujú lingvistické termíny, ako napríklad „nízka“, „stredná“ alebo „vysoká“. Takáto premenná sa označuje ako fuzzy premenná. Na Obrázku 3a je napríklad teplota v rozsahu  $[T_1, T_2]$  charakterizovaná ako fuzzy premenná a na Obrázku 3b je kontrastná s porovnateľnou tradičnou (nefuzzy) premennou. Stav fuzzy premennej sú reprezentované piatimi fuzzy množinami, ktoré zodpovedajú lingvistickým pojmom: „veľmi nízka“, „nízka“, „stredná“, „vysoká“ a „veľmi vysoká“. Tieto množiny sú definované pomocou funkcií príslušnosti s definičným oborom  $[T_1, T_2]$ , a hodnotovým rozsahom  $[0, 1]$ , Grafy týchto funkcií majú najčastejšie lichobežníkový tvar, ktorý – spolu s trojuholníkovým – patrí medzi najpoužívanejšie v súčasnej praxi. Naproti tomu, stavy klasickej premennej sú vymedzené ako ostré množiny pomocou pravostranných otvorených intervalov reálnych čísel. Význam fuzzy premenných spočíva v tom, že umožňujú plynulé prechody medzi stavmi, a tým poskytujú prirodzený spôsob, ako vyjadriť neurčitosť pri pozorovaní a meraní. Fuzzy logika lepšie vystihuje realitu, pretože počíta s neistotou, ktorá je bežná pri meraniach alebo subjektívnych hodnoteniach (Klir & Yuan, 1995)

Obrázok 3 (a,b). Fuzzy Teplota



Zdroj: Klir & Yuan (1995)

### 1.3 Vloženie nepresných dát do databázových systémov

V danej podkapitole sme predstavili možnosti vloženia fuzzy údajov do relačných databáz. Preskúmali sme existujúce možnosti, napríklad klasický SQL alebo jeho rozšírenú verziu pre spracovanie nepresných dát. Použitím presných metód sme poskytli informácie o tom, ako rôzne databázové systémy (PostgreSQL, Oracle, My SQL) môžu tento problém vyriešiť, napríklad funkcie prepínania, operátory alebo rozšírenia. Hoci práca s fuzzy údajmi je zložitý proces, existujú dostupné nástroje na túto náročnú úlohu.

#### 1.3.1 Štandardne SQL nástroje

Structured Query Language (SQL) je jazyk, ktorý databáza dokáže interpretovať. SQL umožňuje vytvárať štruktúry, vykonávať úlohy správy údajov (pridávať, mazať, upravovať) a vykonávať zložité operácie, ktoré transformujú nespracované údaje na informácie. Jazyk SQL je tiež jazykom voľby pre takmer každý systém manažovania databáz. Je to textovo orientovaný jazyk vyžadujúci iba textový procesor, pretože bol vyvinutý dávno pred grafickými používateľskými rozhraniami (Shields, 2019).

#### 1.3.2 Rozšírenia pre fuzzy prácu s dátami .

Jazyk FSQL je rozšírením jazyka SQL, ktorý umožňuje spracovávať fuzzy informácie v fuzzy alebo ostrých databázach. Prvá verzia FSQL bola implementovaná pre databázy Oracle. FSQL obsahuje definíciu niekoľkých užitočných funkcií. Medzi najzaujímavejšie funkcie patria nasledujúce:

- FDEGREE(list\_ atribútov) - vracia fuzzy stupeň priradený k danému atribútu alebo zoznamu atribútov.
- FDEGROW(tabuľka) - vracia fuzzy stupeň priradený celému riadku. Argumentom tejto funkcie je názov tabuľky.
- CARD(fuzzy\_udaj) - vracia kardinalitu fuzzy hodnoty. Pomáha vymedziť riadky s menšou neurčitosťou v danom atribúte.
- NORM(fuzzy\_udaj) - normalizuje fuzzy hodnotu, delí pôvodnú funkciu príslušnosti výškou fuzzy hodnoty (maximálnou hodnotou príslušnosti).
- CONC\_DILAT(fuzzy\_udaj, p) - vracia koncentrovanú alebo rozšírenú verziu fuzzy hodnoty podľa exponentu p: ak  $p > 1$  - koncentrácia: výsledná fuzzy hodnota má nižšie stupne príslušnosti ( $p = 2$ ).

- MORE\_CONTRAST(fuzzy\_udaj, p) - Funkcia na zosilnenie kontrastu fuzzy hodnoty. Členstvá menšie ako 0.5 sa znižujú, hodnoty nad 0.5 sa zvyšujú.
- FUZZIFICATION(fuzzy\_udaj, p) - Funkcia s opačným účinkom ako zosilnenie kontrastu. Teda znižuje kontrast fuzzy množiny.

(Galindo, 2005).

V *Tabuľke 2* sme uviedli konkrétny syntax pri použití rozšírených funkcií FSQL, ako aj výsledok ich aplikácie.

*Tabuľka 2. FSQL funkcie pre praktické použitie*

Funkcia	Definícia
SELECT nazov, MORE_CONTRAST(kvalita, 2) FROM Hotely;	Aplikuje zosilnenie kontrastu na fuzzy hodnoty kvality každého hotela – slabé sa stanú ešte slabšími, silné silnejšími
SELECT * FROM Hotely WHERE FDEGREE(cena ≈ lacna) > 0.7;	Vyberie hotely, ktorých cena má stupeň príslušnosti k fuzzy množine "lacná" väčší ako 0.7
SELECT * FROM Hotely WHERE CARD(kvalita) < CARD(3 +- 1);	Vyberie hotely, ktorých fuzzy kvalita je menej neurčitá než fuzzy konštanta s rozptylom 1.
SELECT * FROM Hotely WHERE INTERSECTION((cena ≈ lacna, kvalita ≈ vysoka), 'min') > 0.5;	Prienik dvoch fuzzy podmienok (lacná + vysoká kvalita) cez minimálnu t-normu

*Zdroj: Vlastne spracovanie podľa (Galindo, 2005).*

### 1.3.3 Výhody a nevýhody rožných prístupov

Na základe práce o GEFRED modeli (Medin, Ponsova, Vil, 1994), práce o analýze funkčných závislosti fuzzy databáz (Vučetić, 2011), sme vymedzili hlavne odlišnosti dvoch prístupov. Práca o efektívite implementácie a aplikácie fuzzy databáz umožnila doplniť porovnávaciu *Tabuľku 3*, ktorá umožňuje komplexný prehľad oboch prístupov v rámci dôležitých parametrov. (Chukwu, Okoronkwo, Ezema, Achie & Nwosu, 2024)

Podľa *Tabuľky 3* - hlavnými výhodami fuzzy riešení: lepšia práca s neurčitosťou a vágnosťou, keďže nie je potrebné vykonávať špecifické výpočty a transformáciu údajov. Taktiež sú tieto riešenia maximálne priblížené ľudskému mysleniu a nevyžadujú konkrétnosť. Takisto je možné daný model riešenia prispôbiť mnohým riadiacim systémom, kde je prítomné rozostrenie údajov. Nevýhody fuzzy riešení: zložitosť pri analýze údajov – keďže údaje boli od začiatku nepresné, pri validácii môže dôjsť k chybe a výskum nebude odrážať realitu. Zložitosť sa tiež vyskytuje pri určovaní pravidiel, keďže pre vytvorenie systému musí byť určený význam každého pojmu (napríklad „vysoká teplota“). Fuzzy riešenia nemajú výhodu na uplatnenie v každom systéme, keďže nie všetky systémy podporujú dané technológie.

Výhody klasických riešení: hlavnou výhodou je presnosť a definovanosť, čo umožňuje spoľahlivé zostavenie analytických výstupov pre následnú optimalizáciu systému. Na aplikáciu daného riešenia nie sú potrebné dodatočné funkcie a rozšírenia, čo umožňuje zavedenie tejto technológie do rôznych štruktúr. Nevýhody klasických riešení: horšia použiteľnosť v reálnom čase – pri veľkom množstve údajov klasické systémy potrebujú viac času na ich spracovanie. Klasické riešenia nepodporujú spracovanie ľudských faktorov a aspektov, pracujú výlučne s presnými typmi údajov.

*Tabuľka 3. Klasické vs Fuzzy riešenie*

	Klasické riešenie	Fuzzy riešenie
Rozšíriteľnosť dát	Presne definovaný dátový typ	Neurčite hodnoty a lingvistické výrazy
Dotazovanie	logické porovnávacie operátory (>, <, =, <=, >=, !=)	Fuzzy podmienky ≈, „mierne“, „približne“
Podpora DB systémov	Oracle, MySQL, PostgreSQL	Určite rozšírenia, ktoré nie každý systém podporuje (podľa aktualizácie verzie)
Normalizácia	Upresnene funkčne závislosti	Špeciálne pravidla fuzzy závislostí
Výkonnosť	Rýchle vyhodnotenie dotazov, použitie transakcií a triggerov, indexov	Rýchlosť sa znižuje kvôli výpočtom mier príslušnosti
Použitie	Bankovníctvo, CRM, ERP systémy, vzdelávanie	Medicína, inžinierstvo, cestovný ruch, logistika, umelá inteligencia

*Zdroj: Vlastne spracovanie podľa Chukwu, Okoronkwo, Ezema, Achie & Nwosu (2024)*

## 1.4 Zhrnutie kapitoly

Najpopulárnejším dátovým modelom je relačný model, ktorý poskytuje efektívne, konzistentné a trvalé ukladanie dát vo forme tabuliek. Databáza je logická kolekcia informácií, ktoré spĺňajú potreby organizácie. Relačné databázy znižujú pravdepodobnosť straty pri uchovávaní údajov, vytvárajú priame prepojenia medzi reláciami prostredníctvom primárneho a cudzieho kľúča a zabezpečujú konzistenciu. Dane databázy sa používajú napríklad, v bankových, komerčných alebo priemyselných systémoch. Napriek týmto výhodám čelia tradičné relačné databázy problémom pri práci s fuzzy údajmi.

Fuzzy údaje môžu byť neúplné, vágne, nejednoznačné alebo nekonzistentné. Tieto typy údajov znižujú presnosť a spoľahlivosť analýzy. Ich prítomnosť v databázach môže byť spôsobená ľudskými chybami, technickými obmedzeniami alebo štrukturálnymi problémami spojenými s jazykovou analýzou. V mnohých prípadoch existujúce systémy správy údajov nemajú vstavané nástroje na riešenie týchto typov problémov.

Fuzzy logika je komplexným riešením daného problému. Neopisuje údaj ako „pravdu“ alebo „nepravdu“, ale dáva hodnotenie medzi 0 a 1, ktoré lepšie odráža fakty a názory. Fuzzy databázy zavádzajú koncept hodnôt, ktorý umožňuje určiť, do akej miery objekt patrí do triedy. Podobné modely boli integrované do poradenských, medicínskych a turistických systémov.

Tradičným informačným systémom často chýba transparentnosť, jednoduchá integrácia a flexibilita. To znamená, že výskum musí spájať výhody štandardizovaných údajov (organizačnú prehľadnosť, konzistentnosť, integritu údajov) so schopnosťou pracovať s neúplnými a nepresnými údajmi neštandardnými spôsobmi. Integrácia relačných databáz a fuzzy logiky zlepšuje pochopenie reálnych situácií a uľahčuje rozhodovanie pri vyskytnutí neistoty.

## 2 Cieľ práce

Databázy sú dôležitým nástrojom na uchovávanie dát v mnohých organizáciách. Pomáhajú disciplinovať a štruktúrovať prácu s citlivými údajmi, ako sú údaje o klientoch, zamestnancoch, predávaných produktoch či nákupoch. Práca sa zaoberá problematikou vkladania neurčitých údajov do relačných databázových systémov podľa normalizačných pravidiel konzistentnosti údajov. To zabezpečuje, že dáta uložené v databáze vytvorenej pre účely cestovnej agentúry sú spoľahlivo chránené a ľahko dostupné. Najskôr bol špecifikovaný hlavný cieľ práce, ktorý určuje jej zmysel a smerovanie. Následne boli definované čiastkové ciele, ktoré rozdeľujú hlavný cieľ na jednotlivé kroky. Vďaka ich špecifikácii bola zabezpečená logická kontinuita výskumného procesu.

### 2.1 Stanovenie hlavného cieľa práce

Hlavným cieľom danej záverečnej práce je vytvorenie odbornej a prakticky využiteľnej databázy, ktorá dokáže riešiť problém vkladania a uchovávanía nepresných údajov. Vytvorená databáza musí obsahovať komplexný prehľad informácií o klientoch a destináciách cestovania, a zároveň umožní uchovávanie intervalových údajov, ako sú teplota a cena ubytovania, ktoré sú reprezentované fuzzy množinami. Zavedením fuzzy logiky táto práca poukazuje na možnosť prekonať obmedzenia relačného modelu. Navrhované riešenie sa snaží uľahčiť úlohu vyhľadávania konkrétneho turistického zájazdu a hotela pre manažéra turistických služieb a zjednodušiť rozhodovací proces pre klienta. Práca zároveň demonštruje možnosť integrácie fuzzy prístupu pri zachovaní relačného modelu a referenčnej integrity údajov

### 2.2 Špecifikácia čiastkových cieľov

Na splnenie hlavného cieľa v plnej miere je potrebné definovať čiastkové ciele, ktoré predstavujú jednotlivé etapy tvorby databázy s prvkami fuzzy logiky.

1. Získanie nepresných údajov a ich modifikácia pomocou funkcií príslušnosti, ktoré umožňujú reprezentáciu nepresných alebo lingvisticky formulovaných hodnôt vo forme fuzzy množín. Plnenie danej čiastkovej úlohy umožňuje integráciu fuzzy údajov do relačnej databázy vo forme prípustných dátových formátov.

2. Návrh konceptuálneho modelu databázy s využitím entitno-relačného prístupu, identifikáciou hlavných entít, ich atribútov a väzieb, so zreteľom na požiadavky cestovnej agentúry. Cieľom je vytvoriť správnu štruktúru na uchovávanie dát pre spoľahlivé nasadenie na serverovej platforme.
3. Implementácia databázy v prostredí MySQL Server, vrátane vytvárania tabuliek, definovania primárnych a cudzích kľúčov a integrácie fuzzy atribútov podľa definovaných množín pomocou SQL jazyka.
4. Overenie funkčnosti riešenia pomocou simulácie reálnej používateľskej situácie, v ktorej klient zadáva nepresne špecifikované požiadavky; následné vyhodnotenie výstupov z databázy z hľadiska ich zrozumiteľnosti a relevantnosti.

Splnenie týchto čiastkových cieľov nielenže poskytne plne funkčnú databázu schopnú spracovávať neurčité údaje, ale zároveň vytvorí základ pre ďalšiu analýzu možností jej rozšírenia a návrhu pokročilých funkcionalít.

### 3 Metodika práce a metódy skúmania

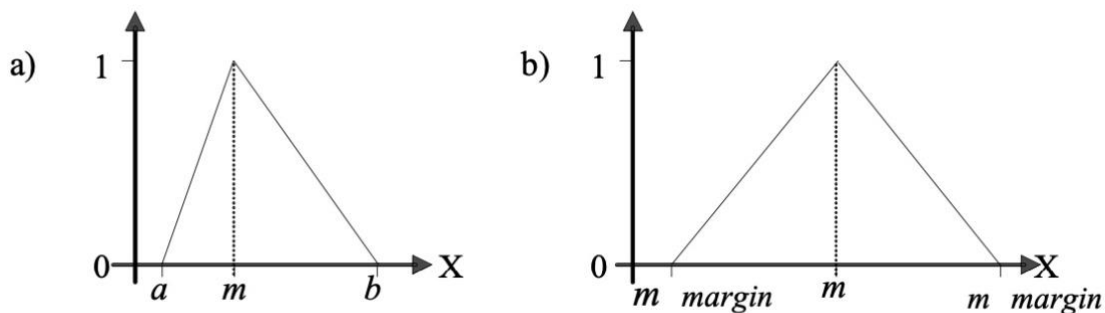
Vzhľadom na stanovený cieľ práce a doplnujúce čiastkové ciele, ktorými je návrh databázového systému schopného spracúvať nepresné požiadavky klienta, vyjadrené fuzzy množinami, boli zvolené metódy návrhu relačných databáz. Hlavným metodologickým prístupom bola integrácia fuzzy premenných do databázy prostredníctvom funkcií príslušnosti. Zvolené postupy umožňujú efektívne reprezentovať lingvisticky vyjadrené preferencie pomocou fuzzy množín a následne ich spracovávať v prostredí relačného databázového modelu.

#### 3.1 Funkcie príslušnosti

Na určenie miery, do akej patrí prvok do fuzzy množiny, sa vo fuzzy logike používajú funkcie príslušnosti. Na rozdiel od klasickej logiky, kde prvok buď úplne patrí alebo vôbec nepatrí do množiny, fuzzy logika umožňuje stupňovú príslušnosť. Je to hodnota v rozsahu od 0 po 1. Na určenie príslušnosti fuzzy množín, ako sú cena a teplota boli použité nasledujúce funkcie príslušnosti.

1. Trojuholníková funkcia je definovaná troma hodnotami – dolnou hranicou  $a$ , hornou hranicou  $b$  a vrcholovou (modálnou) hodnotou  $m$ , pričom platí vzťah  $a < b < m$ . Ak je rozdiel medzi  $b$  a  $m$  rovnaký ako rozdiel medzi  $m$  a  $a$ , tak funkcia má symetrickú šírku okolo vrcholu (tzv. „margin“). Trojuholníková funkcia umožňuje vyjadriť mieru do akej určitá hodnota zodpovedá danému javu.

Obrázok 4. Trojuholníková fuzzy funkcia: a) všeobecná b) symetrická.



Zdroj: Galindo, Urrutia, Piattini (2005)

Funkcia má nasledujúci tvar:

$$A(x) = \begin{cases} 0 & \text{pre } x \leq a \\ \frac{(x-a)}{m-a} & \text{pre } x \in (a, m] \\ \frac{(b-x)}{b-m} & \text{pre } x \in (m, b] \\ 1 & \text{pre } x \geq b \end{cases}$$

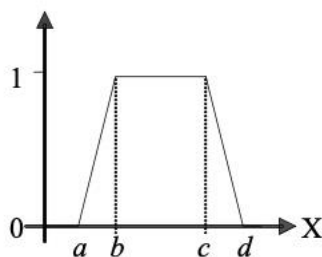
Kde:

$x$  - vstupná hodnota (napr. teplota, cena, skóre)

$A(x)$  - stupeň príslušnosti hodnoty  $x$  do fuzzy množiny

2. Lichobežníková (trapezoid) funkcia: je definovaná dolnou hranicou  $a$  a hornou hranicou  $d$ , ako aj dolnou a hornou hranicou svojho jadra (nukleu), ktoré sú označené ako  $b$  a  $c$ . Lichobežníková funkcia príslušnosti je jednou z najčastejšie používaných funkcií na modelovanie fuzzy množín. Je veľmi flexibilná a efektívna na reprezentáciu vágnych alebo nepresných pojmov, ktoré majú určitý rozsah úplnej pravdivosti (jadro). Dana funkcia schopná vyjadriť vágnosť v jazykových termínoch ako „nízka teplota“, „primeraná cena“, „vysoká spokojnosť“.

Obrázok 5. Lichobežníková fuzzy funkcia



Zdroj: (Galindo, Urrutia, Piattini, 2005)

Funkcia má tvar:

$$T(x) = \begin{cases} 0 & \text{pre } (x \leq a) \text{ o } (x \geq d) \\ \frac{(x-a)}{b-a} & \text{pre } x \in (a, b] \\ 1 & \text{pre } x \in (b, c) \\ \frac{d-x}{(d-c)} & \text{pre } x \in (c, d) \end{cases}$$

Kde:

$T(x)$  - funkcia príslušnosti pre lichobežníkovú fuzzy množinu

$x$  - vstupná hodnota

(Galindo, Urrutia, Piattini, 2005).

3. Lingvistická premenná  $X$  pozostáva zo štvorky  $(T, U, G, M)$ .

Kde:

$T$  - množina lingvistických termínov  $X$

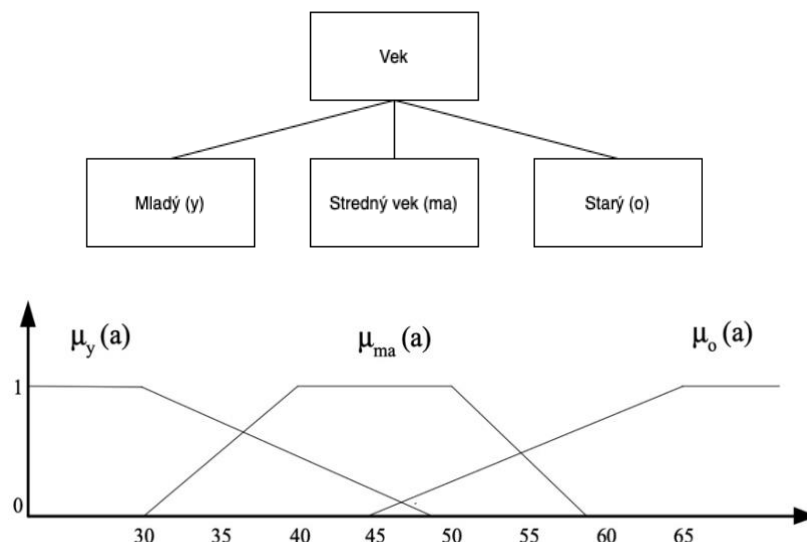
$U$  - univerzum diskurzu

$G$  - množina syntaktických pravidiel, ktoré generujú prvky  $T$ ,

$M$  - množina sémantických pravidiel, ktoré sú preložené z  $T$  a zodpovedajú fuzzy podmnožine  $U$

Na základe tohto vymedzenia je možné definovať konceptuálny model danej premennej a jeho matematické vyjadrenie na *Obrázku 6*. Napríklad, ak  $X = \text{Vek}$ , množina  $T$  je reprezentovaná lingvistickými termínami ako  $\{\text{Mladý, Stredný, Starý}\}$ , generovanými pomocou množiny  $G$ . Každý termín z  $T$  je následne špecifickým spôsobom spracovaný pomocou  $M$  ako fuzzy množina. Zároveň sa definuje typ zodpovednosti medzi entitou a fuzzy entitou, ako aj množina hodnôt, z ktorej stupeň príslušnosti (členstva) vychádza z fuzzy množiny: 1:1, 1:N, N:M, čím sa fuzzy logika integruje do ER modelu. (Chen, 1988 )

Obrázok 6. Lingvistická premenná



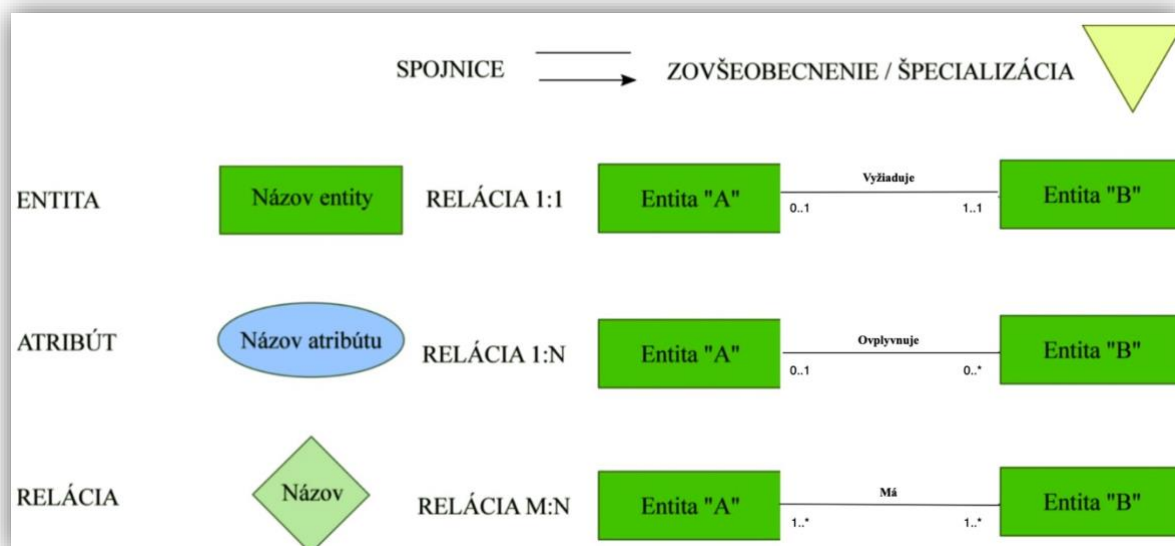
Zdroj: (Chen, 1988)

### 3.2 Entitno-relačný diagram a normalizácia databázy

Na návrh databázového systému bol použitý grafický nástroj – entitno-relačný (E-R) diagram. Tento diagram slúži na vyjadrenie dátového modelu, pričom definuje jednotlivé entity a vzájomné vzťahy medzi nimi.

Podľa Obrázku 7, kde je znázornený postup tvorby entitno-relačného diagramu, entita je objekt, o ktorom sa v databáze uchováva údaje. Atribúty entity vyjadrujú jej bližšiu charakteristiku alebo vlastnosť, pričom každý atribút má svoju doménu – množinu prípustných hodnôt. Relácia definuje vzťah, spojenie medzi jednotlivými entitami. Každá entita musí byť jednoznačne identifikovateľná prostredníctvom primárneho kľúča, ktorý jednoznačne identifikuje každú jej inštanciu (záznam). Cudzí kľúč predstavuje primárny kľúč z inej entity, prostredníctvom ktorého odkazujeme na inštalácie tejto druhej entity. Kardinalita relácie vyjadruje maximálny počet inštancií jednej entity, ktoré môžu byť asociované aspoň s jednou inštanciou inej entity v rámci relácie. Okrem toho relácia je klasifikovaná aj podľa jej stupňa. Existujú tri druhy relácií: binárne, ternárne a všeobecné N-árne relácie. Členstvo entít vo vzťahu môže byť povinné alebo nepovinné. Opačným procesom je špecializácia, pri ktorej rozčleňujeme entitu na nové, špecifickejšie entity, ktoré dedia časť vlastností pôvodnej entity. (Štrbo, 2020)

Obrázok 7. Entitno-relačný diagram



Zdroj: (Štrbo, 2020)

Väzbu medzi triedami znázorňuje čiara spájajúca ich obdĺžniky. Táto čiara môže byť pomenovaná v kontexte vzťahu medzi nimi. Na koncoch čiary sú uvedené čísla označujúce násobnosť (multiplicitu) vzťahu. Prvé číslo je minimálnym počtom objektov (zvyčajne 0 alebo 1). Druhé číslo je maximálnym počtom objektov (často 1 alebo \*, kde „\*“- nekonečnosť), ktorý určuje, koľko objektov môže byť spojených. Tieto čísla spoločne definujú kardinalitu vzťahu. Označenie „1..\*“ znamená, že musí existovať aspoň jeden súvisiaci objekt, pričom ich počet nie je obmedzený. (Chodorow, 2019)

Existujú také hlavne typy vzťahov:

- 1) N:1 (mnoho-na-jeden) binárny vzťah. Primárny kľúč nadradenej tabuľky sa vloží ako cudzí kľúč do podradenej tabuľky.
- 2) M:N (mnoho-na-mnoho) binárny vzťah. Vytvorí sa pomocná tabuľka (tzv. „most“) na reprezentáciu vzťahu. Primárne kľúče z oboch nadradených tabuliek sa vložia do tejto pomocnej tabuľky ako cudzie kľúče.
- 3) 1:1 (jeden-na-jeden) binárny vzťah s povinnou účasťou na jednej strane. Primárny kľúč z tabuľky na strane „1..1“ sa vloží do tabuľky na strane „0..1“ ako cudzí kľúč. (Silpiö, 2016)

Záverečným krokom pred implementáciou databázy v systéme, je proces normalizácie. Podľa Microsoft Corporation, normalizácia je proces transformácie údajov, ktorý zahŕňa tvorbu tabuliek a ich usporiadanie podľa pravidiel s cieľom ochrany pred duplikáciou a anomáliami (2024). Existuje niekoľko pravidiel na konfiguráciu databázy. Tieto pravidla sa označujú ako normálne formy. Po splnení prvého pravidla, sa databáza nachádza v prvej normálnej forme. Ak databáza spĺňa prvé tri normálne formy, označuje sa ako databáza v tretej normálovej forme. Kolade vysvetlil, že keď sú všetky atribúty tabuľky atómové a je definovaný zložený primárny kľúč, tak takáto tabuľka je v prvej normálnej forme (1NF). V druhej normálnej forme (2NF) sa duplicitné skupiny a duplicity údajov eliminujú, ale krátkodobé závislosti sú stále prítomne. To znamená, že nepriamy atribút (prvok, ktorý nie je súčasťou kandidátskeho kľúča) závisí od iného nepriameho atribútu. Prave tieto závislosti sú predmetom eliminácie v tretej normálovej forme (3 NF). (2022)

### 3.3 Algoritmus zavedenia neurčitosti do relačného modelu

Po definovaní konceptu spracovania fuzzy množín nasleduje spôsob ich integrácie do relačného databázového modelu. Tento algoritmus je všeobecným pre relačne dátové

modely a popisuje ako zaviesť neurčitosť priamo do relačnej databázy na základe fuzzy metamodelu s pomocou jazyka SQL. Fuzzy metamodel je abstraktný model, ktorý slúži ako štruktúra pre integráciu fuzzy údajov do relačných databáz. Ide o „nadstavbu“ nad klasickým relačným modelom, ktorá umožňuje databázam pracovať s nepresnými, vágnymi a jazykovými údajmi, typickými pre fuzzy logiku. Profesor opísal algoritmus vloženia nepresných údajov do klasického relačného modelu nasledovne:

1. Vykonať kontrolu, či je atribút fuzzy:

Skontroluje sa, či je atribút evidovaný v tabuľke JE\_FUZZY. Táto tabuľka uchováva informáciu o tom, ktoré atribúty sú definované ako fuzzy.

2. Ak je atribút fuzzy:

(if attribute appears (is fuzzy = true) then). Ak sa v tabuľke JE\_FUZZY atribút má hodnotu je\_fuzzy = 1, znamená to, že ide o fuzzy atribút a musí sa spracovať špeciálnym spôsobom.

3. Generovanie identifikačného čísla pre cudzí kľúč:

(generate ID for foreign key). Ďalším krokom je vytvorenie jedinečného identifikátora, ktorý bude použitý ako cudzí kľúč. Keďže hodnota samotného atribútu sa nebude ukladať priamo – bude uložená v novej tabuľke a v hlavnej tabuľke ju nahradí tento kľúč.

4. Vybrať typ fuzzy množiny:

(check acceptable fuzzy sets for the attribute and offer them in a combo box). Tento krok algoritmu vyžaduje priradenie vhodného typu fuzzy množiny jednotlivým atribútom, ktoré boli označené ako fuzzy. Cieľom je po vykonaní tejto operácie zobrazit' používateľovi dostupné typy fuzzy množín, ako napríklad trojuholníkovú, lichobežníkovú alebo singletonovú, a to pre každý fuzzy atribút osobitne.

5. Zobrazit' vstupné polia podľa typu fuzzy množiny:

(according to selected type of fuzzy set provide input boxes for parameters). Na základe zvoleného typu množiny sa zobrazia polia, do ktorých používateľ zadá hodnoty: pre trojuholníkovú: a, m, b; lichobežníkovú: a, b, c, d; singleton: m.

6. Overenie a ukladanie hodnoty:

(verify input values and store them in relevant tables). Overenie správnosti vstupov (napr.  $a < m < b$ ), potom je potrebné uložiť záznam do príslušnej tabuľky (SINGLETON, TRIANGULAR.) a zároveň je potrebné uložiť cudzí kľúč do hlavnej tabuľky.

## 7. Ak atribút nie je fuzzy:

(else allow storing crisp value for attribute). Ak atribút nie je fuzzy, hodnota sa uloží klasickým spôsobom.

Algoritmus vychádza zo základných operácií SQL: čítanie, vkladanie, aktualizácia a prepojenie tabuliek, a je rozšírený o podporu fuzzy údajov prostredníctvom štruktúry fuzzy metamodelu (Hudec, 2016).

### 3.4 Postup tvorby databázy pomocou SQL v prostredí MySQL

V tejto podkapitole je popísaný presný postup implementácie relačnej databázy pomocou príkazov jazyka SQL v prostredí MySQL, ktoré bolo zvolené ako vývojové prostredie pre realizáciu úlohy.

Základným krokom pri tvorbe databázy v akomkoľvek prostredí je vytvorenie priestoru na ukladanie budúcej databázy. Pomocou príkazu `CREATE DATABASE` (názov\_databázy) sa inicializuje nový priestor, v ktorom budú následne vykonávané všetky operácie. Na prenesenie entít do vývojového prostredia MySQL sa využíva príkaz `CREATE TABLE` (názov\_tabuľky) (názov\_atribútu, dátový\_typ, obmedzenie (ak existuje)). V zátvorkách sa uvádza názov každého atribútu tabuľky, typ údajov: `INT`- pre celé čísla, `FLOAT` pre desatinné čísla, `VARCHAR (n)` pre textové reťazce, kde `n` určuje maximálnu dĺžku (20, 50, 100). Atribúty môžu mať definované obmedzenia ako primárny kľúč (`PRIMARY KEY`) alebo cudzí kľúč (`FOREIGN KEY`), ak sa na atribút vzťahujú. Tabuľky predstavujú základné dátové štruktúry systému, kde každý riadok reprezentuje jeden záznam. Taktiež, väzby medzi tabuľkami môžu byť definované aj po implementácii všetkých hlavných tabuliek pomocou cudzieho kľúča: `ALTER TABLE` (názov\_tabuľky) `ADD CONSTRAINT` (názov\_obmedzenia) `FOREIGN KEY` (názov\_atribútu). Po vykonaní daného kroku sa uloží cudzí kľúč, ktorý zabezpečuje referenčnú integritu medzi tabuľkami. Po vytvorení tabuľkovej štruktúry, sa do databázy vkladajú dáta pomocou príkazu `INSERT INTO` (názov\_tabuľky, názov\_atribútov) `VALUES` (hodnoty). Pri tvorbe dotazov nad viacerými tabuľkami sa často využíva operácia `JOIN`, ktorá umožňuje spojenie záznamov medzi tabuľkami na základe vzťahu medzi primárnym a cudzím kľúčom. (Oracle, 2024)

Uvedené SQL príkazy predstavujú základný a univerzálny postup pre vytvorenie a správu databázy v prostredí MySQL.

## 4 Výsledky práce a diskusia

Táto kapitola predstavuje súhrn dosiahnutých výsledkov vyplývajúcich z implementácie návrhu databázového riešenia využívajúceho fuzzy logiku. Cieľom bolo vytvoriť relačnú databázu schopnú pracovať s nepresnými údajmi. Riešenie umožňuje nielen ukladanie fuzzy údajov, ale aj ich následnú modifikáciu a vyhľadávanie záznamov na základe jazykovo vyjadrených požiadaviek používateľa.

Pri návrhu databázy boli využité dve typy funkcií príslušnosti – trojuholníková a lichobežníková – ktoré slúžia na reprezentáciu nepresných údajov ako teplota jednotlivých miest a cena ubytovania. Konceptuálny návrh bol spracovaný pomocou entitno-relačného modelovania (ER diagram), pričom boli navrhnuté všetky základné aj podporné entity potrebné pre fungovanie systému. Vyvinutá databáza zahŕňa celkovo 17 relácií, ktoré zahŕňajú základné prvky ako klient, smer destinácie, mesto, hotel, izba, účel cestovania, sezóna a servisný systém, ako aj doplnkové tabuľky potrebné pre modelovanie fuzzy metamodelu: trojuhol\_cislo, Prislusn\_t, Interpret\_t, lichob\_cislo, Prislusn\_l, Interpret\_l, JE\_FUZZY, FUZZY\_LINK a FUZZY\_TYP. Spolu vytvárajú komplexnú databázovú štruktúru, ktorá umožňuje spracovanie nepresných údajov pri vyhľadávaní záznamov.

Databáza bola implementovaná na platforme MySQL, ktorá poskytuje robustné riešenie pre prácu s relačnými dátami. Výpočty príslušnosti fuzzy množín boli realizované v prostredí Microsoft Excel, ktorý slúžil na prípravu a overenie transformačných funkcií pred ich vložením do databázy.

### 4.1 Získavanie a úprava údajov

V danej podkapitole sa opisuje pôvod údajov, využitých pri návrhu databázy. Taktiež boli opísané výsledky transformácie fuzzy množín prostredníctvom funkcií príslušnosti, aby dosiahnuté výsledky následne boli použité do relačného modelu.

#### 4.1.1 Typy použitých údajov

Na vývoj databázy sme bolo potrebné získať niekoľko druhov kvalitných údajov, vhodných pre ďalšie spracovanie a analýzu. Spoľahlivá a funkčná databáza si vyžaduje dôsledný výber konzistentných údajov, preto je táto fáza jednou z najdôležitejších.

Na vytvorenie prvej časti databázy sme použili vlastné modelové dáta, najmä pre vytvorenie tabuľky klient, kde údaje boli špeciálne náhodne vygenerované, aby sa predišlo podobnosti so skutočnými osobami. Také citlivé údaje, ako sú telefónne čísla, boli takisto vytvorené umelo, pričom sa maximálne približovali realite. Rovnako pri vytváraní tabuľky hotel, konkrétne v atribúte nazov\_hotel, boli použité neexistujúce názvy hotelov, teda fiktívne. Tieto testovacie údaje sme vytvorili sami pre potreby práce.

Sekundárne dáta boli použité na vytvorenie atribútov nazov\_mesto a nazov\_smer z tabuliek mesto a smer. Tieto typy údajov, teda geografické názvy miest, krajín a ostrovov, boli prevzaté z virtuálnych máp, napríklad Google Maps. Na opis servisných systémov v tabuľke Servis, ako aj na vymedzenie dátumov pre najväčší počet cestujúcich podľa destinácií v tabuľke Sezóna, boli údaje prevzaté z verejne dostupnej turistickej stránky tourism.com.de. Na priblíženie cenovej politiky hotelových izieb pri spracovaní nepresného atribútu cena bol použitý online rezervačný systém ubytovania Booking.com, ktorý poskytuje otvorený prehľad cien vo všetkých mestách spomenutých v databáze tejto bakalárskej práce. Na vytvorenie nepresného atribútu teplota boli použité meteorologické údaje o teplote vzduchu opisovaných miest v našej databáze zo stránky weatherspark.com, kde sa nachádzajú historické záznamy, ktoré sme použili na analýzu. Po získaní meteorologických údajov boli následne agregované a vložené do databázy ako spracované údaje.

#### 4.1.2 Výpočet fuzzy príslušnosti

V rámci našej bakalárskej práce sme sa rozhodli zaviesť dva typy neurčitých údajov: rozsah teplôt a rozsah cien hotelových izieb. Tieto údaje predstavujú špecifickú formu neurčitosti v prípadoch, keď nie je možné jednoznačne a presne opísať kategóriu „Teplota“ alebo „Cena“. V prípade teploty vzniká nejednoznačnosť v dôsledku spôsobu jej zaznamenávania - bežne sa uvádza maximálna a minimálna denná teplota, ktoré sa následne agregujú do priemernej hodnoty. Použitie samostatnej strednej hodnoty nie je vždy vhodné, keďže teploty sa výrazne líšia medzi dennými a nočnými hodinami. Pri

analýze kategórie cien sme taktiež identifikovali dynamiku cien hotelových izieb v závislosti od blížiacich sa sezónnych období, keď je počet turistov najvyšší. Tento jav spôsobuje prirodzené kolísanie cien v čase, čo predstavuje ďalší príklad neurčitosti údajov. Aby bolo možné zabezpečiť konkrétne a zmysluplné odpovede zo strany poskytovateľa turistických služieb pri spracovaní požiadaviek klientov, rozhodli sme o transformáciu údajov do fuzzy formátu.

Tabuľka 4. Zoznam teplôt 2024

Zoznam teplôt 2024								
id_mesto	Mesto	Mesiac		Sezóna		MIN	AVG	MAX
		November		Zima				
MST10	Dubaj	18	33	17,8	28	17,8	22,92	33
MST4	Saint-Bon-Tarentaise	1	12	-8	-4	-8	-1,4	12
MST1	Rovaniemi	-9	0	-14	-4	-14	-8,2	0
		August		Jeseň				
MST3	Pariz	13	26	3	23	3	13,6	26
		Máj		Leto				
MST2	Nice	12	22	16	28	12	18	28
MST5	Barcelona	12	22	16	29	12	18,2	29
MST6	Ibiza	13	24	17	30	13	19,4	30
MST7	Santa Eularia des Riu	13	24	17	29	13	19,2	29
MST8	Istanbul	11	24	16	29	11	18,2	29
MST9	Antalya	18	28	19	35	18	23,6	35
MST11	Denpasar	24	31	25	31	24	27	31
MST12	Singaradza	21	29	25	32	21	25,6	32

*Zdroj: vlastne spracovanie*

Pri prvej transformácii nepresných hodnôt teploty sme v prostredí Exceli vytvorili zoznam na základe nameraných teplôt jednotlivých miest. V *Tabuľke 4* sú uvedené identifikátory a názvy miest. Záznamy sme rozdelili podľa období pribúdania najväčšieho počtu turistov do miest. Mestá boli rozdelené do troch skupín podľa sezóny – každá sezóna zahŕňala celý trimester a jeden prechodný mesiac. Na základe dostupných záznamov boli identifikované tri sezóny: zimná, jesenná a letná. Pre každú sezónu sme uviedli charakteristické teploty: vľavo minimálnu priemernú a vpravo maximálnu priemernú.

Následne sme vykonali transformácie údajov na konkrétne hodnoty. Pomocou funkcie minima sme vypočítali priemernú minimálnu teplotu sezóny. Funkcia aritmetického priemeru poskytla hodnotu hrubej priemernej teploty za sezónu. Funkcia maxima určila maximálnu priemernú teplotu medzi maximálnymi teplotami.

Následne boli vypočítali hodnoty príslušnosti pomocou trojuholníkovej funkcie. Na výpočet príslušnosti hodnôt teploty bola zvolená trojuholníková funkcia, pretože sa bežne používa v prípadoch, keď je známa len približná hodnota. Premenné a, m, a b boli prevzaté ako výstupne hodnoty z *Tabuľky 4*. Hodnoty x (-5, 15, 20, 25) boli zvolené ako reprezentanti jazykových výrazov „zima“, „mierne teplo“, „teplo“, „horúce“. Pre jednotlivé hodnoty x boli v prostredí MS Excel vytvorené vzorce na výpočet stupňa príslušnosti na základe trojuholníkovej funkcie. Určovalo sa, do akej miery záznam teploty s hodnotami a,m,b, v s identifikátorom TEPxxx patrí k premennej x („zima“ a ďalším). Každá z teplôt bola porovnávaná so všetkými definovanými jazykovými termínmi v rámci hodnoty x. Výsledky sú uvedené v *Tabuľke 5*. Interpretácia výsledkov ukazuje, že teplota s identifikátorom TEP10 vykazuje najvyššiu mieru príslušnosti k jazykovému termínu „horúce“, miernu príslušnosť k termínu „teplo“ a žiadnu príslušnosť k termínom „zima“ alebo „mierne teplo“.

*Tabuľka 5. Trojuholníková funkcia príslušnosti*

teplota_id	a	m	b	x			
				zima	mierne teplo	teplo	horuce
				-5	15	20	25
TEP10	17,8	22,92	33	0,0	0,0	0,4	0,8
TEP4	-8	-1,4	12	0,5	0,0	0,0	0,0
TEP1	-14	-8,2	0	0,6	0,0	0,0	0,0
TEP3	3	13,6	26	0,0	0,9	0,5	0,1
TEP2	12	18	28	0,0	0,5	0,8	0,3
TEP5	12	18,2	29	0,0	0,5	0,8	0,4
TEP6	13	19,4	30	0,0	0,3	0,9	0,5
TEP7	13	19,2	29	0,0	0,3	0,9	0,4
TEP8	11	18,2	29	0,0	0,6	0,8	0,4
TEP9	18	23,6	35	0,0	0,0	0,4	0,9
TEP11	24	27	31	0,0	0,0	0,0	0,3
TEP12	21	25,6	32	0,0	0,0	0,0	0,9

Zdroj: vlastné spracovanie

### Funkcie:

- Pre termín „zima“ := IF (F\$4 <= \$C5; 0; IF (F\$4 >= \$E5; 1; IF (AND (F\$4 > \$C5; (F\$4 <= \$D5)); (F\$4 - \$C5) / (\$D5 - \$C5); IF (AND (F\$4 > \$D5; (F\$4 <=\$E5)); (\$E5 - F\$4) / (\$E5 - \$D5); 0))))
- Pre termín „mierne teplo“ := IF (G\$4 <= \$C5; 0; IF (G\$4 >= \$E5; 1; IF (AND (G\$4 > \$C5; (G\$4 <= \$D5)); (G\$4 - \$C5) / (\$D5 - \$C5); IF (AND (G\$4 > \$D5; (G\$4 <=\$E5)); (\$E5 - G\$4) / (\$E5 - \$D5); 0))))
- Pre termín „tepló“ := IF (H\$4 <= \$C5; 0; IF (H\$4 >= \$E5; 1; IF (AND (H\$4 > \$C5; (H\$4 <= \$D5)); (H\$4 - \$C5) / (\$D5 - \$C5); IF (AND (H\$4 > \$D5; (H\$4 <=\$E5)); (\$E5 - H\$4) / (\$E5 - \$D5); 0))))
- Pre termín „horúce“ := IF (I\$4 <= \$C5; 0; IF (I\$4 >= \$E5; 1; IF (AND (I\$4 > \$C5; (I\$4 <= \$D5)); (I\$4 - \$C5) / (\$D5 - \$C5); IF (AND (I\$4 > \$D5; (I\$4 <=\$E5)); (\$E5 - I\$4) / (\$E5 - \$D5); 0))))

Druhou transformovanou premennou bola fuzzy premenná cena. V prostredí Microsoft Exceli bol vytvorený zoznam na základe údajov o jednotlivých izbách v konkrétnych mestách. *Tabuľka 6* obsahuje názov mesta, servisný systém, identifikačné číslo hotela, identifikačné číslo izby, výmeru a identifikačné číslo ceny – teda faktory, ktoré priam ovplyvňujú cenotvorbu. Následne boli týmto izbám priradené fuzzy hodnoty ceny, reprezentované pomocou lichobežníkovej funkcie príslušnosti so štyrmi parametrami: a, b, c a d. Hodnota a predstavuje počiatočnú najnižšiu cenu mimo sezóny, b označuje dolnú hranicu jadra funkcie, c hornú hranicu jadra a d maximálnu cenu počas hlavnej sezóny. Hodnoty boli zvolené na základe pozorovaných cenových rozpätí pre daný typ a lokalitu. Týmto spôsobom bolo možné vyjadriť nepresné hodnoty ceny pomocou jazykových výrazov ako „lacná“, „stredne drahá“, „drahá“, „luxusná“ a následne ich spracovať pomocou fuzzy logiky. V *Tabuľke 6* sú uvedené ukázkové údaje o izbách pre jednotlivé mestá. Celkový počet záznamov predstavuje 75 riadkov.

Tabuľka 6. Zoznam cien

nazov_mesta	serv_system	id_hotel	id_izba	typ_izby	vymera	cena_id	a	d	b	c
Antalya	AI	HOT001	I050	dvojlozkova	25	FC50	85	105	140	170
Barcelona	RO	HOT006	I027	apartman	47	FC27	95	115	160	190
Denpasar	AI	HOT009	I069	apartmán	47	FC69	100	150	170	180
Dubaj	AI	HOT010	I063	apartmán	48	FC63	100	110	120	120
Ibiza	BB	HOT014	I058	apartman	59	FC58	98	118	165	195
Istanbul	RO	HOT018	I044	dvojlozkova	21	FC44	80	100	130	160
Nice	AI	HOT021	I053	apartman	50	FC53	100	120	170	200
Pariz	RO	HOT023	I018	dvojlozkova	23	FC18	80	100	130	160
Rovaniemi	RO	HOT027	I051	apartman	53	FC51	200	230	370	400
Saint-Bon-Tarentaise	AI	HOT028	I055	dvojlozkova	24	FC55	85	105	140	170
Santa Eularia des Riu	AI	HOT031	I040	jednolozkova	18	FC40	75	95	150	180
Singaradza	RO	HOT035	I074	dvojlozkova	26	FC74	80	100	130	160

Zdroj: vlastne spracovanie

Následne boli vypočítané hodnoty príslušnosti pre lichobežníkové fuzzy čísla reprezentujúce cenu. Premenné a, b, c a d boli prevzaté z *Tabuľky 6* ako hranice lichobežníkovej funkcie pre každú konkrétnu hodnotu *cena\_id*. Hodnoty x boli zvolené ako typické reprezentanti pre jazykové výrazy: „lacná“ (23), „mierne drahá“ (100), „drahá“ (200) a „luxusná“ (340). Pre každú hodnotu x bola pomocou vzorcov v prostredí Microsoft Excel vypočítaná miera príslušnosti danej ceny k zvolenej jazykovej premennej. Výsledkom je číselný údaj (v rozsahu od 0 po 1), ktorý určuje, do akej miery sa daná fuzzy cena zodpovedá danému lingvistickému výrazu. Tento spôsob umožnil opísať každú cenu pomocou jazykovo zrozumiteľných kategórií, čo umožňuje využitie fuzzy logiky pri spracovaní nepresných údajov. Výsledky výpočtov sú uvedené v *Tabuľke 7*. Interpretácia výsledkov ukazuje, že cena s identifikátorom FC01 vykazuje najvyššiu mieru príslušnosti k jazykovému termínu „mierne drahá“, a žiadnu príslušnosť k termínom „lacná“, „drahá“ alebo „luxusná“. V *Tabuľke 7* sú uvedené ukážkové údaje o izbách pre jednotlivé mestá. Celkový počet záznamov predstavuje 75 riadkov

Tabuľka 7. Lichobežníková funkcia prislušnosti

cena_id	a	b	c	d	x			
					lacna	mierne draha	draha	luxusna
					23	100	200	340
FC01	70	90	140	170	0,0	1,0	0,0	0,0
FC02	80	100	130	140	0,0	1,0	0,0	0,0
FC03	300	400	430	470	0,0	0,0	0,0	0,4
FC04	290	300	300	350	0,0	0,0	0,0	0,2
FC05	270	300	450	500	0,0	0,0	0,0	1,0
FC06	270	300	370	400	0,0	0,0	0,0	1,0
FC07	75	95	150	180	0,0	1,0	0,0	0,0
FC08	85	105	140	170	0,0	0,8	0,0	0,0
FC09	70	90	140	170	0,0	1,0	0,0	0,0
FC10	95	115	160	190	0,0	0,3	0,0	0,0

Zdroj: vlastne spracovanie

Funkcie:

- Pre termín „lacna“ :  

$$=IF(G\$4 \leq \$C5; 0; IF(G\$4 \geq \$F5; 0; IF(AND(G\$4 > \$C5; (G\$4 \leq \$D5)) ; (G\$4 - \$C5) / (\$D5 - \$C5); IF(AND(G\$4 > \$D5; G\$4 < \$E5); 1; IF(AND(G\$4 > \$D5; G\$4 < \$F5); (\$F5 - G\$4) / (\$F5 - \$E5); 0))))$$
- Pr termín „mierne draha“ :  

$$=IF(H\$4 \leq \$C5; 0; IF(H\$4 \geq \$F5; 0; IF(AND(H\$4 > \$C5; (H\$4 \leq \$D5)) ; (H\$4 - \$C5) / (\$D5 - \$C5); IF(AND(H\$4 > \$D5; H\$4 < \$E5); 1; IF(AND(H\$4 > \$D5; H\$4 < \$F5); (\$F5 - H\$4) / (\$F5 - \$E5); 0))))$$
- Pre termín „draha“ :  

$$=IF(I\$4 \leq \$C5; 0; IF(I\$4 \geq \$F5; 0; IF(AND(I\$4 > \$C5; (I\$4 \leq \$D5)) ; (I\$4 - \$C5) / (\$D5 - \$C5); IF(AND(I\$4 > \$D5; I\$4 < \$E5); 1; IF(AND(I\$4 > \$D5; I\$4 < \$F5); (\$F5 - I\$4) / (\$F5 - \$E5); 0))))$$

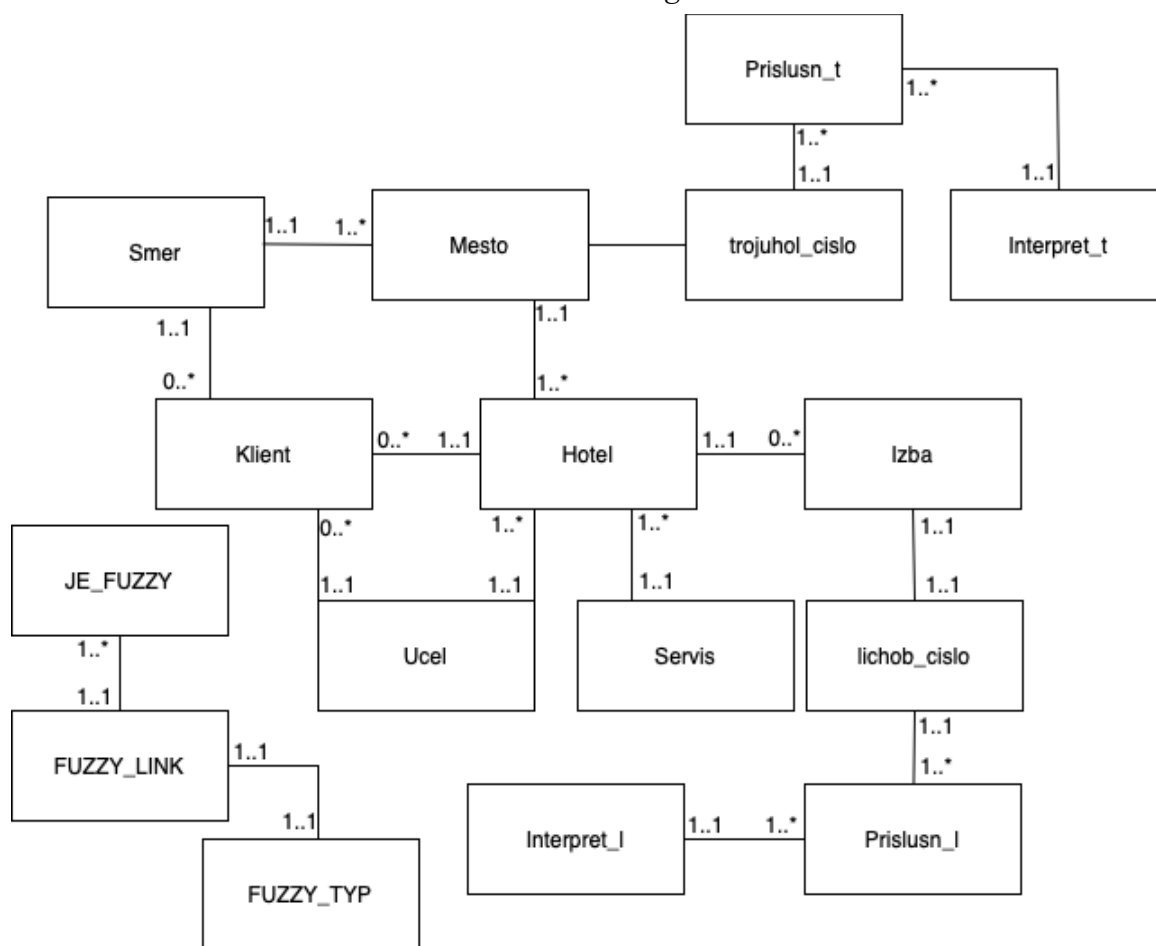
- Pre termín „luxusna“ :

$$=IF(J\$4\leq\$C5;0;IF(J\$4\geq\$F5;0;IF(AND(J\$4>\$C5;(J\$4\leq\$D5));(J\$4-\$C5)/(\$D5-\$C5);IF(AND(J\$4>\$D5;J\$4<\$E5);1;IF(AND(J\$4>\$D5;J\$4<\$F5);(\$F5-J\$4)/(\$F5-\$E5);0))))$$

Výsledkom daných výpočtov je možnosť vloženia modifikovaných údajov o teplote a cene do databázy zo zachovaním jej integrity, možnosti aktualizácie a interpretácie údajov

#### 4.2 Návrh databázovej štruktúry. ER diagram

Obrázok 8. ER diagram



Zdroj: vlastne spracovanie

Na Obrázku 8 je znázornený konceptuálny model databázy, navrhnutý prostredníctvom entitno-relačného diagramu. Daný model vytvorený pre účely cestovnej

agentúry, ktorý zároveň integruje podporu fuzzy logiky. V tejto fáze vývoja sme opísali databázu na úrovni relácií, atribútov a vzťahov medzi nimi. Model pozostáva zo súboru 17 entít, ktoré sa rozlišujú podľa hlavných entít (napríklad klient, hotel, mesto, izba ) a entít vysvetľujúcich fuzzy množiny (Prislusn\_l, Prislusn\_t, Interpret\_t, Interpret\_l). Každé Mesto patrí do určitého Smeru (napr. krajiny alebo ostrovu) a obsahuje jeden alebo viac Hotelov. Tie ponúkajú rôzne Izby, pričom každá Izba má priradenú fuzzy Cenu, reprezentovanú entitou lichob\_cislo. Analogicky, každé Mesto má priradenú fuzzy Teplotu, uchovávanú v entite trojuhol\_cislo. Na evidenciu toho, ktoré atribúty sú fuzzy, bol použitý fuzzy metamodel tvorený entitami JE\_FUZZY, FUZZY\_LINK a FUZZY\_TYP. Tento model umožňuje flexibilne pridávať ďalšie fuzzy atribúty. Vzťahy medzi entitami sú určené pomocou kardinalít, pričom model zachováva referenčnú integritu.

Na *Obrázku 9* predstavená schéma 1 z tabuľkami Klient, Mesto, Smer, Ucel a Sezona. Nasleduje podrobnejší popis jednotlivých entít.

1. Klient (id\_klient, meno, priezvisko, vek, pohlavie, tel\_cislo, ucel\_id, smer\_id, mesto\_id, hotel\_id, izba\_id)  
Primárny kľúč: id\_klient  
Cudzí kľúč: ucel\_id odkazuje na Ucel(ucel\_id)  
Cudzí kľúč: smer\_id odkazuje na Smer(smer\_id)  
Cudzí kľúč: mesto\_id odkazuje na Mesto(id\_mesto)  
Cudzí kľúč: hotel\_id odkazuje na Hotel(id\_hotel)  
Cudzí kľúč: izba\_id odkazuje na Izba(id\_izba)

V danej tabuľke je popísala entita Klient, ktorá obsahuje všetky dôležité informácie o klientoch cestovnej agentúry. Atribút id\_klient odkazuje na identifikačné číslo každého klienta, čo je unikátnym identifikátorom. Meno a priezvisko klienta zapísané v atribútoch Meno a Priezvisko. Atribút vek reprezentuje číselný vek každého klienta. Pohlavie je pre každého klienta zapísané ako „muz“ alebo „zena“. Atribút tel\_cislo reprezentuje telefónne číslo klienta, vhodné pre kontakt s klientom. Ucel\_id a smer\_id odkazujú na identifikačné číslo účelu a smeru cestovania. Mesto\_id, hotel\_id a izba\_id sú taktiež unikátne odkazy pre samostatné údaje príslušných tabuliek.

Obrázok 9. Schéma 1

id_klient	meno	priezvisko	vek	pohlavie	tel_cislo	ucel_id	smer_id	mesto_id	hotel_id	izba_id
kl001	Marek	Slavik	34	muz	0904123456	UC01	1	MST1	HOT002	I004
kl002	Andrea	Mala	29	zena	0907111222	UC02	2	MST2	HOT006	I011
kl003	Peter	Cerny	41	muz	0908111333	UC03	2	MST3	HOT007	I013
kl004	Lucia	Valentova	37	zena	0910555666	UC01	2	MST4	HOT012	I023
kl005	Jozef	Novotny	31	muz	0910123456	UC02	3	MST5	HOT013	I025

Tabuľka "Klient"

id_smer	id_mesto	nazov_mesto
1	MST1	Rovaniemi
2	MST2	Nice
2	MST3	Pariz
2	MST4	Saint-Bon-Tarentaise
3	MST5	Barcelona
4	MST6	Ibiza
4	MST7	Santa Eularia des Riu
5	MST8	Istanbul
5	MST9	Antalya
6	MST10	Dubaj
7	MST11	Denpasar
7	MST12	Singaradza

Tabuľka "Mesto"

id_ucel	popis_ucel
UC01	zabava
UC02	romantika
UC03	zdravie

Tabuľka "Ucel"

smer_id	nazov_smer
1	Finsko
2	Francuzsko
3	Spanielso
4	Ibiza
5	Turecko
6	Spojene arabske emiraty
7	Bali

Tabuľka "Smer"

id_sezon	zaciatok	koniec
SZN001	1.11.2025	28.2.2026
SZN002	1.11.2025	28.2.2026
SZN003	1.11.2025	28.2.2026
SZN004	1.8.2025	31.10.2025
SZN005	1.5.2025	31.7.2025
SZN006	1.5.2025	31.7.2025
SZN007	1.5.2025	31.7.2025
SZN008	1.5.2025	31.7.2025
SZN009	1.5.2025	31.7.2025
SZN010	1.5.2025	31.7.2025
SZN011	1.5.2025	31.7.2025
SZN012	1.5.2025	31.7.2025

Tabuľka "Sezona"

Zdroj: vlastne spracovanie

## 2. Ucel (ucel\_id, popis\_ucel)

Primárny kľúč: ucel\_id

Tato tabuľka reprezentuje konkrétny účel cestovania klienta. Účelom môže byť: zábava a oddych, romantický výlet alebo zlepšenie zdravia.

## 3. Smer (smer\_id, nazov\_smer)

Primárny kľúč: smer\_id

V tabuľke Smer sa nachádzajú údaje o možných destináciách ponúkaných cestovnou agentúrou. Nejde o krajiny, ale konkrétne o smery, pretože v možnostiach ponuky sú predstavené aj názvy ostrovov, ako napríklad Bali (Indonézia) alebo Ibiza (Španielsko). Pre lepšie pochopenie boli tieto destinácie rozdelené osobitne. Existuje sedem destinácií: Fínsko, Francúzsko, Španielsko, Ibiza, Turecko, Spojené arabské emiráty a ostrov Bali.

4. Mesto (id\_mesto, nazov\_mesto, smer\_id, teplota\_id)

Primárny kľúč: id\_mesto

Cudzí kľúč: smer\_id odkazuje na Smer(smer\_id)

Cudzí kľúč: id\_sezon odkazuje na Sezona(id\_sezon)

Cudzí kľúč: teplota\_id odkazuje na trojuhol\_cislo(teplota\_id)

V tabuľke Mesto je uvedených 12 miest: Rovaniemi (Fínsko), Nice (Francúzsko), Paríž (Francúzsko), Saint-Bon-Tarentaise (Francúzsko), Barcelona (Španielsko), Ibiza (Španielsko, Ibiza), Santa Eulària des Riu (Španielsko, Ibiza), Istanbul (Turecko), Antalya (Turecko), Dubaj (Spojene arabské emiraty), Denpasar (Indonézia, Bali) a Singaradza (Indonézia, Bali). Okrem identifikácie mesta tabuľka obsahuje aj odkazy na fuzzy atribút teplota\_id a na atribút definujúci začiatok a koniec turistickej sezóny, čím je zabezpečené logické prepojenie s ďalšími entitami databázy.

5. Sezona (id\_sezon, zaciatok, koniec)

Primárny kľúč: id\_hotel

V danej tabuľke sú uvedené dátumy začiatku a konca turistickej sezóny pre každé mesto samostatne. Sú dôležitými informáciami pre pokračovanie tvorby logickej databázy.

Na *Obrázku 10* predstavená schéma 2 z tabuľkami Hotel a Izba (prvých 25 záznamov). Nasleduje podrobnejší popis jednotlivých entít.

6. Hotel (id\_hotel, nazov\_hotel, id\_mesto, ucel\_id, serv\_system, popis\_hotel)

Primárny kľúč: id\_hotel

Cudzí kľúč: id\_mesto odkazuje na Mesto(id\_mesto)

Cudzí kľúč: ucel\_id odkazuje na Ucel(ucel\_id)

Cudzí kľúč: serv\_system odkazuje na Servis(serv\_system)

Ku každému Mestu patria 2 až 4 hotely. Každý Hotel má identifikačné číslo, názov, prepojenie na mesto, prepojenie na účel, určený servisný systém a popis. Existuje 27 hotelov.

7. Izba (id\_izba, id\_hotel, typ\_izby, vymera, cena\_id)

Primárny kľúč: id\_izba

Cudzí kľúč: id\_hotel odkazuje na Hotel(id\_hotel)

Cudzí kľúč: cena\_id odkazuje na lichob\_cislo(cena\_id)

Každý Hotel má niekoľko (2 až 4) Izieb, rozdelených podľa štandardov. Štandardy sú: jednolôžková, dvojlôžková izba a apartmán. Taktiež entita obsahuje výmeru v metroch štvorcových a odkaz na fuzzy atribút cena\_id. Existujú 75 izieb.

Obrázok 10. Schéma 2

Tabuľka "Hotel"

id_hotel	nazov_hotel	id_mesto	ucel_id	serv_system	popis_hotel
HOT001	Santa Hotel	MST1	UC03	RO	Ubytovanie pri zjazdovke so skibusom
HOT002	Blue Hotel	MST1	UC01	AI	Hotel v blízkosti lyžiarskeho strediska
HOT003	Aurora Hotel	MST1	UC02	UAI	Zimný rezort s wellness centrom
HOT004	La vie el bel	MST2	UC02	AI	Resort s bazenom a sukromnou plazou
HOT005	Dream Hotel	MST2	UC03	RO	Hotel s vyhladom na more a slnečnou terasou
HOT006	Royal Hotel	MST2	UC03	AI	Resort s bazenom a sukromnou plazou
HOT007	Eiffel	MST3	UC02	RO	Hotel s vyhladom na vezu a slnečnou terasou
HOT008	La mour	MST3	UC01	AI	Ihrisko je pri hoteli
HOT009	Croissant Inn	MST3	UC01	RO	Moderne ubytovanie v centre mesta
HOT010	Sun Hotel	MST4	UC01	AI	Chata s kozubom a saunou vhodna pre zimnu dovolenku
HOT011	Alpine Hotel	MST4	UC02	AI	Ubytovanie pri zjazdovke so skibusom
HOT012	Snow Hotel	MST4	UC03	RO	Zimný rezort s wellness centrom a vyhladom na hory
HOT013	Barsa Hotel	MST5	UC02	AI	Moderne ubytovanie v centre prímorského mesta
HOT014	Park wellness	MST5	UC03	RO	Moderne ubytovanie v centre prímorského mesta s wellnessom
HOT015	Grazie Hotel	MST5	UC01	RO	Romantický hotel pre pary blízko pláže
HOT016	Ibiza at	MST6	UC02	RO	Moderne ubytovanie v centre prímorského mesta
HOT017	Raven club resort	MST6	UC02	UAI	Hotel s vlastnou plazou
HOT018	Royal Hotel	MST6	UC01	BB	Moderne ubytovanie v centre prímorského mesta
HOT020	Santa Eulalia	MST7	UC01	UAI	Romantický hotel pre pary blízko pláže
HOT021	Rio Lala	MST7	UC01	AI	Resort s bazenom a sukromnou plazou
HOT022	Kebab Inn	MST8	UC02	UAI	Moderne ubytovanie v centre mesta
HOT023	Lover Hotel	MST8	UC01	RO	Romantický hotel pre pary
HOT025	Royal Hotel	MST9	UC01	AI	Hotel s vyhladom na more a slnečnou terasou
HOT026	Swimmers Club Hotel	MST9	UC03	UAI	Moderne ubytovanie v centre prímorského mesta
HOT027	Dolphine Hotel	MST9	UC02	AI	Hotel s vyhladom na more a slnečnou terasou

Tabuľka "Izba"

id_izba	id_hotel	typ_izby	vymera	cena_id
I001	HOT001	jednolozkova	18	FC01
I002	HOT001	dvojlozkova	25	FC02
I003	HOT002	apartman	40	FC03
I004	HOT002	dvojlozkova	26	FC04
I005	HOT003	apartman	45	FC05
I006	HOT003	dvojlozkova	22	FC06
I007	HOT004	jednolozkova	16	FC07
I008	HOT004	dvojlozkova	24	FC08
I009	HOT005	jednolozkova	15	FC09
I010	HOT005	apartman	48	FC10
I011	HOT006	jednolozkova	17	FC11
I012	HOT006	apartman	50	FC12
I013	HOT007	dvojlozkova	26	FC13
I014	HOT007	jednolozkova	15	FC14
I015	HOT008	dvojlozkova	24	FC15
I016	HOT008	apartman	43	FC16
I017	HOT009	jednolozkova	18	FC17
I018	HOT009	dvojlozkova	23	FC18
I019	HOT010	apartman	52	FC19
I020	HOT010	dvojlozkova	27	FC20
I021	HOT011	jednolozkova	16	FC21
I022	HOT011	dvojlozkova	25	FC22
I023	HOT012	apartman	46	FC23
I024	HOT012	dvojlozkova	22	FC24
I025	HOT013	jednolozkova	19	FC25

Zdroj: vlastne spracovanie

Na Obrázku 11 predstavená schéma 3 z tabuľkami Servis, lichob\_cislo, Interpret\_1, Prislsn\_1. Nasleduje podrobnejší popis jednotlivých entít.

Obrázok 11. Schéma 3

servisny_system_id	servisny_system_popis
RO	Room only. Iba izba - ubytovanie bez stravy.
BB	Bed & Breakfast. Ubytovanie s raňajkami
AI	All Inclusive. Tri jedla denne, dalsie občerstvenie, dezerty a napoje
UAI	Ultra All Inclusive. Rozsirena moznost AI

**Tabuľka "Servis"**

cena_id	a	b	c	d
FC01	70	90	140	170
FC02	80	100	130	140
FC03	300	400	430	470
FC04	250	270	300	350
FC05	270	300	450	500
FC06	270	300	370	400
FC07	75	95	150	180
FC08	85	105	140	170
FC09	70	90	140	170
FC10	95	115	160	190

**Tabuľka "lichob\_cislo"**

cena_id	id_int_l	hodnota
FC01	1intl	0.00
FC01	2intl	1.00
FC01	3intl	0.00
FC01	4intl	0.00
FC02	1intl	0.00
FC02	2intl	1.00
FC02	3intl	0.00
FC02	4intl	0.00
FC03	1intl	0.00
FC03	2intl	0.00

**Tabuľka "Prislsn\_l"**

id_int_l	termin	hodnota
1intl	lacna	23
2intl	mierne draha	100
3intl	draha	200
4intl	luxusna	340

**Tabuľka "Interpret\_l"**

Zdroj: vlastne spracovanie

#### 8. Servis (serv\_system, popis)

Primárny kľúč: serv\_system

Každý hotel má pridelený servisný systém: RO, BB, AI alebo UAI. RO je servis „Iba izba“, čo znamená, že hotel poskytuje iba ubytovanie bez stravy. BB je „Lôžko a raňajky“ – hotel poskytuje ubytovanie a v cene sú zahrnuté aj raňajky pre každého hosťa. AI znamená „Všetko v cene“, teda hotel poskytuje neobmedzenú stravu, nápoje a izbový servis. UAI predstavuje „Úplne všetko v cene“ – väčšinou ide o luxusné hotely, ktoré hosťom ponúkajú aj drahé importované alkoholické nápoje, stravu, nápoje a izbový servis.

#### 9. lichob\_cislo (cena\_id, a, b, c, d)

Primárny kľúč: cena\_id

V tejto tabuľke je predstavené uchovávanie fuzzy údajov ceny, na opis ktorého najlepšie vyhovovala lichobežníková funkcia. Údaje sa ukladajú podľa takéhoto princípu: „a“ – počiatočná cena mimo turistickej sezóny – dolná hranica lichobežníkovej funkcie, „b“ – konečná cena pred začiatkom turistickej sezóny – dolná hranica jadra lichobežníkovej funkcie, „c“ – cena na začiatku turistickej sezóny – horná hranica jadra lichobežníkovej funkcie, „d“ – cena v rozkvetu turistickej sezóny – horná hranica lichobežníkovej funkcie.

10. Interpret\_1 (id\_int\_1, termin, hodnota)

Primárny kľúč: id\_int\_1

Táto tabuľka bola vytvorená na interpretáciu x-hodnôt lichobežníkovej funkcie. Údaje o jazykovej premennej sú uložené v atribúte termín. Atribút hodnota bol vytvorený umelo.

11. Prislusn\_1 (cena\_id, id\_int\_1, hodnota)

Primárny kľúč: zložený kľúč (cena\_id, id\_int\_1)

Cudzí kľúč: cena\_id odkazuje na lichob\_cislo(cena\_id)

Cudzí kľúč: id\_int\_1 odkazuje na Interpret\_1(id\_int\_1)

V tejto tabuľke sme opísali výsledky lichobežníkovej funkcie príslušnosti, ktoré sú uložené v atribúte hodnota. Každé hodnote z atribútu cena\_id bola priradená hodnota id\_int\_1 a z atribútu hodnota.

Na *Obrázku 12* predstavená schéma 4 z tabuľkami trojuhol\_cislo, Interpret\_t, Prislusn\_t. Nasleduje podrobnejší popis jednotlivých entít.

12. trojuhol\_cislo (teplota\_id, a, m, b)

Primárny kľúč: teplota\_id

V tejto tabuľke je predstavené uchovávanie fuzzy údajov teplota, na opis ktorého najlepšie vyhovovala trojuholníková funkcia. Údaje sa ukladajú podľa takéhoto princípu: „a“ – minimálna priemerná teplota počas sezóny – dolná hranica trojuholníkovej funkcie. „m“ – hrubá priemerná teplota počas sezóny – modálna hodnota (vrcholová hodnota) trojuholníkovej funkcie. „b“ – maximálna priemerná teplota počas sezóny – horná hranica trojuholníkovej funkcie.

13. Interpret\_t (id\_int\_t, termin, hodnota)

Primárny kľúč: id\_int\_t

Táto tabuľka bola vytvorená na interpretáciu x-hodnôt pre trojuholníkovú funkciu. Údaje o jazykovej premennej sú uložené v atribúte termín. Atribút hodnota bol vytvorený umelo.

Obrázok 12. Schéma 4

teplota_id	a	m	b
TEP1	-14.00	-8.20	0.00
TEP2	12.00	18.00	28.00
TEP3	3.00	13.60	26.00
TEP4	-8.00	-1.40	12.00
TEP5	12.00	18.20	29.00
TEP6	13.00	19.40	30.00
TEP7	13.00	19.20	29.00
TEP8	11.00	18.20	29.00
TEP9	18.00	23.60	35.00
TEP10	17.80	22.92	33.00
TEP11	24.00	27.00	31.00
TEP12	21.00	25.60	32.00

**Tabuľka "trojuhol\_cislo"**

id_int_t	termin	hodnota
1int	zima	-5
2int	mierne teplo	15
3int	teplo	20
4int	horuco	25

**Tabuľka "Interpret\_t"**

teplota_id	id_int_t	prislusnost
TEP1	1int	0.60
MST1	2int	0.00
MST1	3int	0.00
MST1	4int	0.00
MST10	1int	0.00
MST10	2int	0.00
MST10	3int	0.40
MST10	4int	0.80
MST11	1int	0.00
MST11	2int	0.00

**Tabuľka "Prislusn\_t"**

Zdroj: vlastne spracovanie

14. Prislusn\_t (id\_mesto, id\_int\_t, prislusnost)

Primárny kľúč: zložený kľúč (id\_mesto, id\_int\_t)

Cudzí kľúč: id\_mesto odkazuje na Mesto(id\_mesto)

Cudzí kľúč: id\_int\_t odkazuje na Interpret\_t(id\_int\_t)

V tejto tabuľke sme opísali výsledky trojuholníkovej funkcie príslušnosti, ktoré sú uložené v atribúte hodnota. Každý hodnote z atribútu teplota\_id bola priradená hodnota id\_int\_t a z atribútu prislusnost.

Na Obrázku 13 predstavená schéma 5 z tabuľkami JE\_FUZZY, FUZZY\_LINK, FUZZY\_TYP. Nasleduje podrobnejší popis jednotlivých entít.

15. JE\_FUZZY (tabulka, atribut, je\_fuzzy, atribut\_id)

Primárny kľúč: z (atribut\_id)

V tejto tabuľke sme identifikovali, ktoré atribúty sú fuzzy. Uviedli sme tabuľky (atribút tabuľka), z ktorých boli prevzaté atribúty (atribut) a ich vzťah (je\_fuzzy), kde 0 znamená „nie“ a 1 znamená „áno“. Atribút atribut\_id je unikátnym identifikátormi v danej tabuľke.

Obrázok 13. Schéma 5

**Tabuľka "JE\_FUZZY "**

tabuľka	atribut	je_fuzzy	atribut_id
Izba	cena_id	1	FUZ01
Mesto	teplota_id	1	FUZ02

**Tabuľka  
"FUZZY\_LINK"**

atribut_id	fuzzy_id
FUZ01	1f
FUZ02	2f

**Tabuľka "FUZZY\_TYP"**

fuzzy_id	typ
1f	trojuholníkové fuzzy číslo
2f	lichobežníkové fuzzy číslo

Zdroj: vlastne spracovanie

16. FUZZY\_LINK (atribut\_id, fuzzy\_id)

Primárny kľúč: zložený kľúč (atribut\_id, fuzzy\_id)

Cudzí kľúč: fuzzy\_id odkazuje na FUZZY\_TYP(fuzzy\_id)

Cudzí kľúč: atribut\_id odkazuje na JE\_FUZZY (atribut\_id)

Táto tabuľka bola vytvorená za účelom identifikácie funkcií neurčitosti pri fuzzy atribútoch. Atribút atribut\_id predstavuje odkaz na konkrétny fuzzy atribút a atribút fuzzy\_id slúži ako identifikátor použitej fuzzy funkcie

17. FUZZY\_TYP (fuzzy\_id, typ)

Primárny kľúč: fuzzy\_id

Táto tabuľka bola vytvorená za účelom evidencie typov fuzzy funkcií používaných pri reprezentácii neurčitých údajov. Atribút fuzzy\_id slúži ako identifikátor konkrétneho typu fuzzy množiny a atribút typ popisuje formu fuzzy čísla, napríklad trojuholníkové alebo lichobežníkové fuzzy číslo.

Pred realizáciou databázy prebehlo overenie jej konceptuálneho modelu pomocou normalizačných pravidiel. Výsledkom overenia bolo, že model spĺňa všetky požiadavky normálnych foriem (1NF, 2NF, 3NF).

1. Prvá normálna forma bola splnená tým, že každý údaj bol atomický; atribúty relácií obsahujú len jednotlivé a nedeliteľne hodnoty.
2. Druhá normálna forma je zabezpečená celkovou závislosťou nekľúčových atribútov od kľúčových (Primárnych kľúčov).
3. Tretia normálna forma je dodržaná odstránením, tranzitívnych závislostí.

#### 4.3 Implementácia a testovanie databázy

Implementácia databázy predstavuje fázu, v ktorej bol navrhnutý logický model prevedený do konkrétneho relačného databázového systému. V rámci tejto práce bola databáza implementovaná v systéme MySQL, konkrétne na serveri poskytovanom webovým hosťiteľom [sql4.webzdarma.cz](http://sql4.webzdarma.cz).

##### 4.3.1 Implementácia na MySQL serveri. Logický model databázy

Na implementáciu relačného modelu databázy boli použité štandardne SQL príkazy. Najprv boli vytvorené tabuľky pomocou príkazu `CREATE TABLE`. `lichob_cislo` alebo `Prislusn_l` boli prevedené do CSV formátu, ktorý bol neskôr priamo vložený na SQL server. Táto možnosť umožňuje obísť dlhé skripty SQL jazyka a jednoducho vytvárať tabuľky. Pomocou SQL príkazov boli vytvorené tabuľky s presne definovaným primárnym kľúčom, napríklad pre entitu Hotel:

```
CREATE TABLE Hotel (  
  id_hotel VARCHAR(10) PRIMARY KEY,  
  nazov_hotel VARCHAR(100) NOT NULL,  
  id_mesto VARCHAR(10) NOT NULL,
```

```

ucel_id VARCHAR(10) NOT NULL,
serv_system VARCHAR(10) NOT NULL,
popis_hotel VARCHAR(255);

```

Na vloženie údajov do tabuliek bol použitý príkaz: INSERT napríklad:

```

INSERT INTO Hotel (id_hotel, nazov_hotel, id_mesto, ucel_id,
serv_system, popis_hotel)
VALUES ('HOT001', 'Santa Hotel', 'MST1', 'UC03', 'RO', 'Ubytovanie
pri zjazdovke so skibusom');

```

Na vytvorenie cudzieho kľúča bol použitý SQL príkaz ADD CONSTRAINT napríklad:

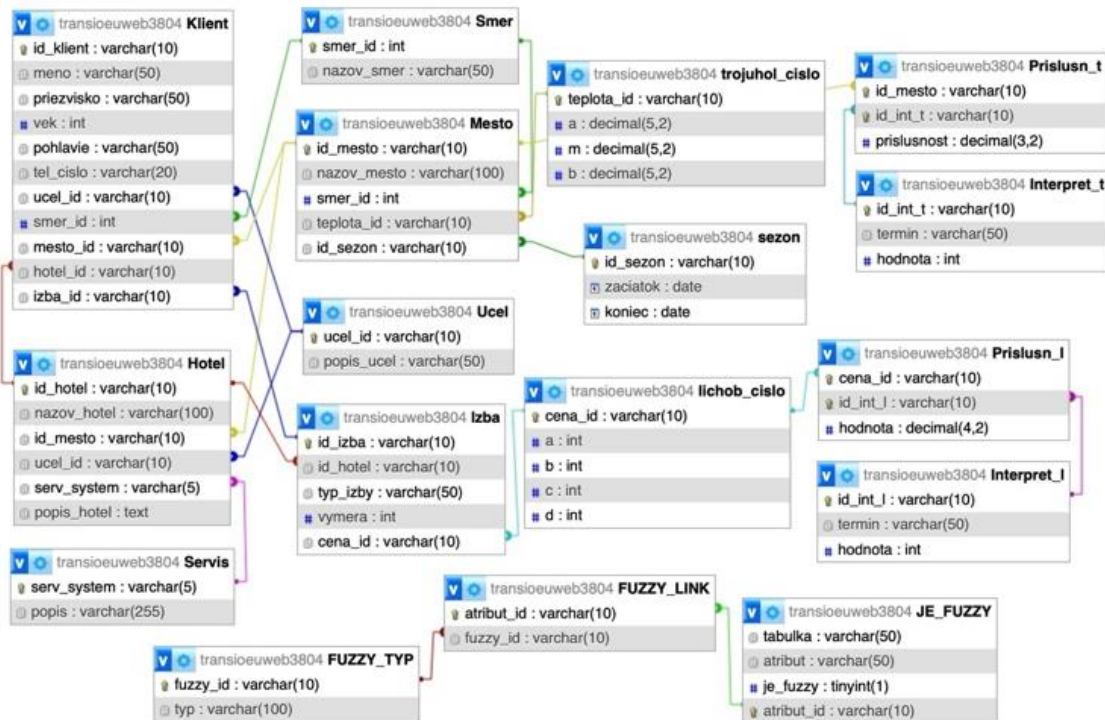
```

ALTER TABLE Hotel
ADD CONSTRAINT fk_hotel_mesto
FOREIGN KEY (id_mesto)
REFERENCES Mesto(id_mesto);

```

Na implementáciu fuzzy časti sme vytvorili tabuľky lichob\_cislo, trojuhol\_cislo, kde sa uchovávajú údaje o cenách a teplotách vzduchu, opísané vyššie. Tabuľky interpretácie Interpret\_l, Interpret\_t. Tabuľky Prislusn\_l, Prislusn\_t, kde boli prenesené výsledky výpočtov funkcií. Tabuľky JE\_FUZZY, FUZZY\_LINK a FUZZY\_TYP boli vytvorené na prepojenie na fuzzy spracovanie.

Obrázok 14. Logický dizajn databázy



Zdroj: vlastne spracovanie

Vo výsledku sme implementovali plne funkčnú databázu so zabezpečením všetkých pravidiel ukladania údajov a zachovania ich konzistencie.

#### 4.3.2 Výsledky dotazovania

V tejto podkapitole prezentujeme výsledky dotazovania nad implementovanou databázou. Cieľom bolo otestovať funkčnosť systému pomocou SQL dotazov. Pre kompletný prehľad možností získavania dôležitých informácií z vytvorenej databázy sme postupovali podľa nasledujúcich pokynov:

1. Možnosti základných funkcií databáz CREATE, READ, UPDATE, DELETE
2. Kontrola správnosti výsledkov pomocou zložitých SQL príkazov

Sme vygenerovali náhodné dotazy na základe definovaných úloh pri práci s databázou použité zo strany pracovníkov cestovnej kancelárie a skontrolovali ich výsledky.

Úloha: Vybrať všetky smery, mestá, hotely a izby, kde teplota je „zima“.

SQL príkaz:

```
SELECT Smer.nazov_smer, Mesto.nazov_mesto, Hotel.nazov_hotel,
Izba.typ_izby, Izba.vymera
FROM Prislusn_t
INNER JOIN Mesto ON Prislusn_t.teplota_id = Mesto.teplota_id
INNER JOIN Smer ON Mesto.smer_id = Smer.smer_id
INNER JOIN Hotel ON Mesto.id_mesto = Hotel.id_mesto
INNER JOIN Izba ON Hotel.id_hotel = Izba.id_hotel
INNER JOIN Interpret_t ON Prislusn_t.id_int_t =
Interpret_t.id_int_t
WHERE
Interpret_t.termin = 'zima' AND Prislusn_t.prislusnost > 0
ORDER BY
Smer.nazov_smer, Mesto.nazov_mesto, Hotel.nazov_hotel;
```

Výsledok: podľa *Obrázku 15* dotaz správne identifikuje všetky relevantné smery, mestá, hotely a izby, ktorých teplota spadá do jazykovej kategórie „zima“. Výsledok potvrdzuje schopnosť databázy spracovať fuzzy údaj na základe hodnoty príslušnosti k jazykovému termínu. V databáze boli definované dva smery s fuzzy teplotou „zima“: Francuzsko, Saint-Bon-Tarentaise a Fincko, Rovaniemi. Presne dane destinácie sú odpoveďou na požiadavku „zimnej“ teploty.

Obrázok 15. Dotaz 1

nazov_smer	nazov_mesto	nazov_hotel	typ_izby	vymera
Finsko	Rovaniemi	Aurora Hotel	apartman	45
Finsko	Rovaniemi	Aurora Hotel	dvojlozkova	22
Finsko	Rovaniemi	Blue Hotel	apartman	40
Finsko	Rovaniemi	Blue Hotel	dvojlozkova	26
Finsko	Rovaniemi	Santa Hotel	jednolozkova	18
Finsko	Rovaniemi	Santa Hotel	dvojlozkova	25
Finsko	Rovaniemi	Santa Hotel	apartman	53
Francuzsko	Saint-Bon-Tarentaise	Alpine Hotel	jednolozkova	16
Francuzsko	Saint-Bon-Tarentaise	Alpine Hotel	dvojlozkova	25
Francuzsko	Saint-Bon-Tarentaise	Snow Hotel	apartman	46

Zdroj: výsledok behu systému

Úloha: Vybrať všetky izby, ktorých cena je „drahá“ alebo „mierne drahá“, teplota je „zima“ alebo „mierne teplo“, a ukázať názov hotela, mesta, smeru, maximálnu cenu za noc pričom účel cestovania, je zlepšenie zdravia.

SQL príkaz:

```
SELECT Hotel.nazov_hotel, Mesto.nazov_mesto, Smer.nazov_smer,
Ucel.popis_ucel AS ucel_cestovania,
Izba.id_izba, Izba.typ_izby, lichob_cislo.d AS max_cena_noc,
Interpret_l.termin AS cenova_kategoria, Interpret_t.termin AS
teplotna_kategoria
FROM Hotel
JOIN Mesto ON Hotel.id_mesto = Mesto.id_mesto
JOIN Smer ON Mesto.smer_id = Smer.smer_id
JOIN Ucel ON Hotel.ucel_id = Ucel.ucel_id
JOIN Izba ON Hotel.id_hotel = Izba.id_hotel
JOIN lichob_cislo ON Izba.cena_id = lichob_cislo.cena_id
JOIN Prislusn_l ON Izba.cena_id = Prislusn_l.cena_id
JOIN Interpret_l ON Prislusn_l.id_int_l = Interpret_l.id_int_l
JOIN Prislusn_t ON Mesto.id_mesto = Prislusn_t.id_mesto
JOIN Interpret_t ON Prislusn_t.id_int_t = Interpret_t.id_int_t
WHERE Interpret_l.termin IN ('mierne draha', 'draha')
AND Interpret_t.termin IN ('zima', 'mierne teplo') AND
Ucel.popis_ucel = 'zdravie'
AND Prislusn_l.hodnota > 0 AND Prislusn_t.prislusnost > 0
ORDER BY max_cena_noc ASC;
```

Výsledok: podľa Obrázku 16, dotaz vrátil správne filtrované výsledky podľa zadaných lingvistických podmienok. Vďaka použitiu fuzzy logiky boli úspešne spracované nepresné výrazy ako „mierne teplo“ a „drahá“, čím sa demonštruje praktická využiteľnosť fuzzy

údajov v cestovnej agentúre. Touto operáciou sme získali komplexný prehľad údajov týkajúcich sa definovanej úlohy. Takto môžeme zistiť maximálnu cenu za noc pri zadaných obmedzeniach.

Obrázok 16. Dotaz 2

Zobrazíť všetko   Počet riadkov: 100   Filtrovať riadky: Hľadať v tabuľke   Sort by key: Žiadny								
Extra options								
nazov_hotel	nazov_mesto	nazov_smer	ucel_cestovania	id_izba	typ_izby	max_cena_noc	cenova_kategoria	teplotna_kategoria ▲ 1
Dream Hotel	Nice	Francuzsko	zdravie	I010	apartman	190	mierne draha	mierne teplo
Dream Hotel	Nice	Francuzsko	zdravie	I009	jednolozkova	170	mierne draha	mierne teplo
Royal Hotel	Nice	Francuzsko	zdravie	I012	apartman	200	mierne draha	mierne teplo
Royal Hotel	Nice	Francuzsko	zdravie	I011	jednolozkova	180	mierne draha	mierne teplo
Park wellness	Barcelona	Spanielsko	zdravie	I028	jednolozkova	170	mierne draha	mierne teplo
Park wellness	Barcelona	Spanielsko	zdravie	I027	apartman	190	mierne draha	mierne teplo
Royal Hotel	Nice	Francuzsko	zdravie	I053	apartman	200	draha	mierne teplo
Santa Hotel	Rovaniemi	Finsko	zdravie	I002	dvojlozkova	140	mierne draha	zima
Santa Hotel	Rovaniemi	Finsko	zdravie	I001	jednolozkova	170	mierne draha	zima
Snow Hotel	Saint-Bon-Tarentaise	Francuzsko	zdravie	I056	jednolozkova	170	mierne draha	zima
Snow Hotel	Saint-Bon-Tarentaise	Francuzsko	zdravie	I024	dvojlozkova	160	mierne draha	zima
Snow Hotel	Saint-Bon-Tarentaise	Francuzsko	zdravie	I023	apartman	190	mierne draha	zima

Zdroj: výsledok behu systému

Na základe realizovaných testovacích dopytov môžeme potvrdiť, že vyvinutá databáza správne podporuje spracovanie neurčitých údajov pomocou fuzzy logiky. Dotazy umožňujú kombinovať viaceré podmienky založené na lingvistických hodnotách, pričom výstupy spĺňajú požiadavky na kvalitu, presnosť a flexibilitu vyhľadávania.

#### 4.3.3 Manipulácie s fuzzy údajmi

Vďaka riešeniu navrhnutému v tejto bakalárskej práci bola umožnená priama manipulácia s nepresnými údajmi v databáze. Na príklade fuzzy reprezentácie teploty a ceny z tabuliek trojuhol\_cislo a lichob\_cislo bolo možné vložiť jednu z týchto premenných priamo do databázy bez potreby predchádzajúceho presného zaokrúhľovania. V súlade s pravidlami fungovania relačných databázových systémov bola zároveň zabezpečená možnosť neskoršej úpravy týchto údajov alebo ich odstránenia zo systému, čím bola zachovaná plná flexibilita práce s dátami.

Príkaz:

```
INSERT INTO `trojuhol_cislo` (`teplota_id`, `a`, `m`, `b`)
VALUES ('TEP13', 18.00, 24.00, 30.00);
INSERT INTO `Mesto` (`id_mesto`, `nazov_mesto`, `smer_id`,
`teplota_id`, `id_sezon`)
VALUES ('MST13', 'Madrid', 3, 'TEP13', 'SZN006');
INSERT INTO `Prislusn_t` (`id_mesto`, `id_int_t`,
```

```

`prislusnost`) VALUES
('MST13', '1int', 0.00),
('MST13', '2int', 0.00),
('MST13', '3int', 0.30),
('MST13', '4int', 0.80);

```

Daným SQL príkazom sme do tabuľky trojuhol\_cislo vložili nový fuzzy údaj o teplote. Trojuholníkový interval s identifikačným číslom TEP13, kde intervalom je  $a = 18.00$  (začiatok prípustnej teploty),  $m = 24.00$  (najideálnejšia teplota),  $b = 30.00$  (koniec prípustnej teploty). Tento záznam definuje, aká teplota je „ideálna“ pre Madrid. Zároveň sme do tabuľky Mesto pridali nové mesto Madrid s identifikačným číslom MST13, ktoré patrí do smeru 3 (Španielsko) a patrí k sezóne SZN006 (letná sezóna). Do tabuľky Prislusn\_t na základe trojuholníkovej funkcie príslušnosti sme priradili Madridu preferencie k typom teplôt: 1int (zima)– príslušnosť 0.00 (nepríslušná), 2int (mierne teplo)– príslušnosť 0.00 (nepríslušná), 3int (teplo)– príslušnosť 0.30 (slabá príslušnosť), 4int (horúce) – príslušnosť 0.80 (silná príslušnosť). Vo výsledku sme dostali nový záznam o fuzzy teplote konkrétne pre mesto Madrid s príslušnosťou ku “teplej” teplote aj silnou príslušnosťou ku “horúcej” teplote.

Príkaz:

```

SELECT
Mesto.nazov_mesto, trojuhol_cislo.a, trojuhol_cislo.m,
trojuhol_cislo.b, Interpret_t.termin, Prislusn_t.prislusnost
FROM Mesto m
JOIN trojuhol_cislo ON Mesto.teplota_id =
trojuhol_cislo.teplota_id
JOIN Prislusn_t ON Mesto.teplota_id = Prislusn_t.teplota_id
JOIN Interpret_t ON Prislusn_t.id_int_t =
Interpret_t.id_int_t
WHERE Mesto.id_mesto = 'MST13';

```

Daným SQL príkazom sme požiadali o kompletný prehľad dosiahnutých výsledkov pri zadaní novej hodnoty fuzzy teploty. Ďalej pre mesto Madrid teplota bude definovaná

Obrázok 17. Dotaz 3

nazov_mesto	a	m	b	termin_pocasia	prislusnost
Madrid	18.00	24.00	30.00	zima	0.00
Madrid	18.00	24.00	30.00	mierne teplo	0.00
Madrid	18.00	24.00	30.00	teplo	0.30
Madrid	18.00	24.00	30.00	horuco	0.80

ako “tepla” alebo “horúce” pri lingvistickom ohodnotení, lebo pre tieto hodnoty ma najvyšší príslušnosť.

#### 4.4 Simulácia výberu zázjazdu pre klienta

V danej podkapitole chceme popísať kompletnú procedúru rozhodovania klienta na základe fuzzy logiky, ktorú ovplyvni náš systém. Klientom je Ivan Ivanov, 25-ročný muž, ktorého účelom cestovania je zábava. Ani smer ani konkrétne mesto klient si momentálne nezvolil. Jeho kontaktné telefónne číslo je 0876543210. Na základe účelu „zabava“ bol vyhládaný vhodný zoznam hotelov pomocou nasledujúceho SQL dopytu:

```
SELECT h.id_hotel, h.nazov_hotel, m.nazov_mesto,
s.nazov_smer, h.popis_hotel
FROM Hotel h
JOIN Mesto m ON h.id_mesto = m.id_mesto
JOIN Smer s ON m.smer_id = s.smer_id
WHERE h.ucel_id = 'UC01';
```

Výsledok systému: zobrazil sa zoznam všetkých hotelov s účelom „zábava“ v rôznych mestách a krajinách. (Obrázok 18)

Obrázok 18. Zoznam hotelov

Extra options				
id_hotel	nazov_hotel	nazov_mesto	nazov_smer	popis_hotel
HOT002	Blue Hotel	Rovaniemi	Finsko	Hotel v blízkosti lyžiarskeho strediska
HOT008	La mour	Pariz	Francuzsko	Ihrisko je pri hoteli
HOT009	Croissant Inn	Pariz	Francuzsko	Moderne ubytovanie v centre mesta
HOT010	Sun Hotel	Saint-Bon-Tarentaise	Francuzsko	Chata s kozubom a saunou vhodna pre zimnu dovolenk...
HOT015	Grazie Hotel	Barcelona	Spanielsko	Romanticky hotel pre pary blizko plaze
HOT018	Royal Hotel	Ibiza	Ibiza	Moderne ubytovanie v centre prímorského mesta
HOT020	Santa Eulalia	Santa Eularia des Riu	Ibiza	Romanticky hotel pre pary blizko plaze
HOT021	Rio Lala	Santa Eularia des Riu	Ibiza	Resort s bazenom a sukromnou plazou
HOT023	Lover Hotel	Istanbul	Turecko	Romanticky hotel pre pary
HOT025	Royal Hotel	Antalya	Turecko	Hotel s vyhladom na more a slnečnou terasou
HOT029	Park Villa Hotel	Dubaj	Spojene arabske emiraty	Resort s bazenom a sukromnou plazou
HOT034	BlueHouse Hotel	Singaradza	Bali	Moderne ubytovanie v centre prímorského mesta
HOT035	Junggles	Singaradza	Bali	Moderne ubytovanie v centre prímorského mesta

Zdroj: výsledok behu systému

Klient si želal, aby destinácia spĺňala klimatickú podmienku „mierne teplo“, „teplo“ a „horúce“ podnebie. Na výber hotelov spĺňajúcich tieto kritériá bol použitý nasledujúci dopyt:

```

SELECT Hotel.id_hotel, Hotel.nazov_hotel,
Mesto.nazov_mesto, Smer.nazov_smer,
Hotel.popis_hotel, Ucel.popis_ucel
AS ucel_cestovania, Interpret_t.termin
FROM Hotel
JOIN Mesto ON Hotel.id_mesto = Mesto.id_mesto
JOIN Smer ON Mesto.smer_id = Smer.smer_id
JOIN Ucel ON Hotel.ucel_id = Ucel.ucel_id
JOIN Prislusn_t ON Mesto.id_mesto = Prislusn_t.id_mesto
JOIN Interpret_t ON Prislusn_t.id_int_t =
Interpret_t.id_int_t
WHERE Hotel.ucel_id = 'UC01'
AND Interpret_t.termin IN ('mierne teplo', 'teplo',
'horuco')
AND Prislusn_t.prislusnost > 0
ORDER BY Hotel.nazov_hotel, Prislusn_t.prislusnost
DESC;

```

Výsledok systému: Systém vrátil zoznam hotelov zodpovedajúcich požiadavke na teplotu. (Obrázok 19)

Obrázok 19. Zoznam hotelov podľa teploty

id_hotel	nazov_hotel	nazov_mesto	nazov_smer	popis_hotel	ucel_cestovania	termin
HOT034	BlueHouse Hotel	Singaradza	Bali	Moderne ubytovanie v centre prímorského mesta	zabava	horuco
HOT009	Croissant Inn	Pariz	Francuzsko	Moderne ubytovanie v centre mesta	zabava	mierne teplo
HOT009	Croissant Inn	Pariz	Francuzsko	Moderne ubytovanie v centre mesta	zabava	teplo
HOT009	Croissant Inn	Pariz	Francuzsko	Moderne ubytovanie v centre mesta	zabava	horuco
HOT015	Grazie Hotel	Barcelona	Spanielsko	Romantický hotel pre pary blízko pláže	zabava	teplo
HOT015	Grazie Hotel	Barcelona	Spanielsko	Romantický hotel pre pary blízko pláže	zabava	mierne teplo
HOT015	Grazie Hotel	Barcelona	Spanielsko	Romantický hotel pre pary blízko pláže	zabava	horuco
HOT035	Junggles	Singaradza	Bali	Moderne ubytovanie v centre prímorského mesta	zabava	horuco
HOT008	La mour	Pariz	Francuzsko	Ihrisko je pri hoteli	zabava	mierne teplo
HOT008	La mour	Pariz	Francuzsko	Ihrisko je pri hoteli	zabava	teplo
HOT008	La mour	Pariz	Francuzsko	Ihrisko je pri hoteli	zabava	horuco
HOT023	Lover Hotel	Istanbul	Turecko	Romantický hotel pre pary	zabava	teplo
HOT023	Lover Hotel	Istanbul	Turecko	Romantický hotel pre pary	zabava	mierne teplo
HOT023	Lover Hotel	Istanbul	Turecko	Romantický hotel pre pary	zabava	horuco
HOT029	Park Villa Hotel	Dubaj	Spojene arabske emiraty	Resort s bazenom a sukromnou plazou	zabava	horuco

Zdroj: výsledok behu systému

Ďalšou požiadavkou od klienta bolo určiť cenu izby. Podľa pána Ivanova, cena izby za noc musí byť iba „mierne drahá“.

```

SELECT Hotel.id_hotel, Hotel.nazov_hotel,
Mesto.nazov_mesto, Smer.nazov_smer,

```

```

Ucel.popis_ucel AS ucel_cestovania,
Interpret_t.termin AS teplotny_termin, Izba.id_izba,
Izba.typ_izby, Interpret_l.termin
FROM Hotel
JOIN Mesto ON Hotel.id_mesto = Mesto.id_mesto
JOIN Smer ON Mesto.smer_id = Smer.smer_id
JOIN Ucel ON Hotel.ucel_id = Ucel.ucel_id
JOIN Prislusn_t ON Mesto.id_mesto = Prislusn_t.id_mesto
JOIN Interpret_t ON Prislusn_t.id_int_t =
Interpret_t.id_int_t
JOIN Izba ON Izba.id_hotel = Hotel.id_hotel
JOIN lichob_cislo ON Izba.cena_id =
lichob_cislo.cena_id
JOIN Prislusn_l ON Izba.cena_id = Prislusn_l.cena_id
JOIN Interpret_l ON Prislusn_l.id_int_l =
Interpret_l.id_int_l
WHERE Hotel.ucel_id = 'UC01' AND Prislusn_t.id_int_t =
'2int' AND Prislusn_t.prislusnost > 0 AND
Prislusn_termin = 'mierne draha' AND
Prislusn_l.hodnota > 0;

```

Výsledok: zobrazili sa hotely s izbou „mierne drahou“ a požadovaným podnebím. (Obrázok 20)

Obrázok 20. Zoznam hotelov podľa teploty a ceny

id_hotel	nazov_hotel	nazov_mesto	nazov_smer	ucel_cestovania	teplotny_termin	id_izba	typ_izby	termin
HOT008	La mour	Pariz	Francuzsko	zabava	mierne teplo	I015	dvojlozkova	mierne draha
HOT008	La mour	Pariz	Francuzsko	zabava	mierne teplo	I054	jednolozkova	mierne draha
HOT009	Croissant Inn	Pariz	Francuzsko	zabava	mierne teplo	I017	jednolozkova	mierne draha
HOT009	Croissant Inn	Pariz	Francuzsko	zabava	mierne teplo	I018	dvojlozkova	mierne draha
HOT015	Grazie Hotel	Barcelona	Spanielsko	zabava	mierne teplo	I029	dvojlozkova	mierne draha
HOT015	Grazie Hotel	Barcelona	Spanielsko	zabava	mierne teplo	I030	apartman	mierne draha
HOT015	Grazie Hotel	Barcelona	Spanielsko	zabava	mierne teplo	I057	dvojlozkova	mierne draha
HOT020	Santa Eulalia	Santa Eularia des Riu	Ibiza	zabava	mierne teplo	I037	jednolozkova	mierne draha
HOT020	Santa Eulalia	Santa Eularia des Riu	Ibiza	zabava	mierne teplo	I038	dvojlozkova	mierne draha
HOT021	Rio Lala	Santa Eularia des Riu	Ibiza	zabava	mierne teplo	I040	jednolozkova	mierne draha
HOT021	Rio Lala	Santa Eularia des Riu	Ibiza	zabava	mierne teplo	I059	jednolozkova	mierne draha
HOT018	Royal Hotel	Ibiza	Ibiza	zabava	mierne teplo	I035	dvojlozkova	mierne draha
HOT018	Royal Hotel	Ibiza	Ibiza	zabava	mierne teplo	I036	apartman	mierne draha
HOT018	Royal Hotel	Ibiza	Ibiza	zabava	mierne teplo	I058	apartman	mierne draha
HOT023	Love Hotel	Istanbul	Turecko	zabava	mierne teplo	I043	jednolozkova	mierne draha

Zdroj : výsledok behu systému

Po analýze možnosti predstavených miest (Mesto.nazov\_mest), klient si všimol že ma až tri možnosti hotelov v meste Barcelona , čo bolo jeho obľúbené mesto. Tak si rozhodol pozrieť možnosti cien daných izieb. Bol to hotel s identifikátorom HOT015. Ďalej si klient požiadal o prehľad možnosti izieb v danom hoteli.

```
SELECT Izba.id_izba, Izba.typ_izby, Izba.vymera,
Interpret_1.termin AS cenovy_termin
FROM Izba
JOIN lichob_cislo ON Izba.cena_id =
lichob_cislo.cena_id
JOIN Prislusn_1 ON Izba.cena_id = Prislusn_1.cena_id
JOIN Interpret_1 ON Prislusn_1.id_int_1 =
Interpret_1.id_int_1
WHERE Izba.id_hotel = 'HOT015'
AND Interpret_1.termin = 'mierne draha'
AND Prislusn_1.hodnota > 0
ORDER BY Izba.id_izba;
```

Výsledok: Zobrazili sa možnosti ubytovania. (Obrázok 21)

*Obrázok 21. Zoznam izieb*

id_izba	typ_izby	vymera	cenovy_termin
I029	dvojizbova	26	mierne draha
I030	apartman	55	mierne draha
I057	dvojizbova	26	mierne draha

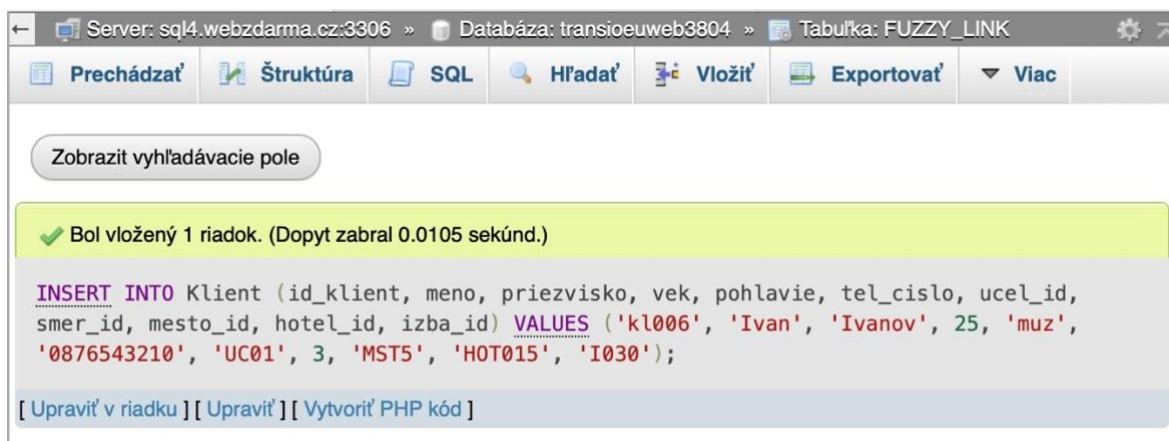
*Zdroj: výsledok behu systému*

Po kompletnej analýze poskytnutých výsledkov, klient si všimol že za rovnakú cenu má možnosť si zvolit' izbu s typom „apartman“. Preto sa rozhodol sa ubytovať na izbe s identifikačným číslom I030. Ďalším krokom bolo zavedenie klienta do databázy cestovnej agentúry. (Obrázok 22)

Výsledky preukázali, že databázový systém správne implementuje fuzzy logiku pre spracovanie neurčitých údajov. Vyhľadávanie na základe lingvistických hodnôt teploty a ceny bolo úspešné a umožnilo klientovi efektívne rozhodovanie. Celý proces potvrdil, že

navrhnutá databáza je schopná flexibilne reagovať na požiadavky obsahujúce neurčité informácie a správne ich spracovať do konkrétnych odporúčaní.

Obrázok 22. . Evidencia nového klienta



Zdroj: výsledok behu systému

#### 4.5 Diskusia a možnosti zlepšenia

Na zlepšenie tejto úlohy by mohli prispieť nasledovné techniky:

- Rozšírenie databázy na zvýšenie možností vyhľadávania v rámci cestovnej agentúry. Napríklad rozšírenie tabuliek o rezervačný systém, kde by bolo možné zapisovať určité dátumy v súlade so sezónou. Rovnako by bolo možné vytvoriť tabuľku, v ktorej by sa vypočítavala konkrétna suma, ktorú klient minul počas celého svojho pobytu. To by pomohlo lepšie sledovať výdavky klientov.
- Pre fuzzy dáta by sa mohla pridať možnosť rozšíreného fuzzy vyhľadávania na základe lingvistických hodnôt. Napríklad by bolo možné prepojiť systém s UML systémami, ktoré by dokázali interpretovať konkrétne slová v ľudských vetách ako napríklad: „Neviem, chcem, aby to bolo dobré.“ Pre štandardné systémy by bola takáto požiadavka veľmi náročná vzhľadom na komplexnosť ľudskej filozofie. Na tomto princípe by sa dala vytvoriť tabuľka „Hodnotenie hotelov“, kde by bolo možné posudzovať a vyberať hotely na základe fuzzy hodnotení. Toto hodnotenie by bolo založené na ľudských recenziách ako lingvistických premenných, z ktorých by UML systém čítal počet pozitívnych a negatívnych slov a prevádzal ich do

tabuľky, z ktorej by bolo možné čerpať údaje. Takéto riešenia by pozitívne ovplyvnili spokojnosť klientov, a tým by sa zvýšil ich počet v cestovnej agentúre.

- V súvislosti s touto prácou by sa dali fuzzy atribúty lichob\_cislo a trojuhol\_cislo (opisujúce ceny a teploty) ďalej rozšíriť o nové možné hodnoty. Napríklad pridaním termínov ako „veľká zima“ alebo „veľmi lacná“, čo by podporilo rozšírenie fuzzy množín. Toto by rozšírilo možnosti vyhľadávania a vytvorilo ešte jednoduchší systém na spracovanie nepresných údajov od klientov.
- Pomocou automatizovaných systémov, ako je napríklad Desktop Power Automate (nástroj na tvorbu webových riešení bez potreby kódovania, založený na UX princípoch), by sa dali automatizovať tabuľky uchovávajúce ceny a teploty. Tieto systémy dokážu automaticky čítať informácie z webových stránok alebo iných aplikácií a vkladať ich do príslušných polí v tabuľke. Tým by sa znížila pracovná vyťaženosť človeka a zároveň by sa zjednodušila údržba systému v aktuálnom stave.

## Záver

V tejto bakalárskej práci sme sa zaoberali problematikou spracovania neurčitých údajov v relačných databázach s využitím fuzzy logiky. Cieľom tejto práce bolo navrhnúť databázové riešenie schopné efektívne uchovávať a spracovávať nepresné údaje bez porušenia pravidiel návrhu relačného modelu.

V teoretickej časti tejto práce sme skúmali možnosti a štruktúru relačných databáz, predstavili kľúčové myšlienky z literatúry o prístupoch fuzzy logiky a diskutovali o súčasnom stave technológií implementácie fuzzy databáz vo svete. Diskutovali sme o výhodách a nevýhodách klasických a fuzzy riešení a charakterizovali súčasné problémy nepresnosti. Riešenou problematikou bolo potvrdenie zložitosti integrácie fuzzy logiky do databázového prostredia, čo v reálnom svete obmedzuje možnosti spracovania fuzzy údajov pomocou relačných modelov.

V praktickej časti sme vytvorili relačný model databázy pre cestovnú agentúru s prvkami fuzzy logiky. Nepresnými atribútmi na spracovanie boli cena izieb a teplota. Na úspešnú implementáciu tejto logiky v databáze sme použili trojuholníkové a lichobežníkové fuzzy množiny, pričom sme zachovali referenčnú integritu a normalizačné pravidlá. Pomocou v práci definovaných postupov sme transformovali definíciu nepresných údajov do formátov kompatibilných so štandardnými relačnými databázami.

Po testovaní systému sme získali presné a logické odpovede vhodné na poskytnutie klientovi cestovnej agentúry. Výsledky potvrdili, že kombinácia relačného modelu s prvkami fuzzy logiky výrazne zlepšuje schopnosť databáz pracovať s neurčitými, nepresne špecifikovanými údajmi. Dané riešenie urýchlilo spracovanie jazykovo formulovaných požiadaviek klientov pri vyhľadávaní turistických zájazdov.

Prínosom práce je praktické ukázanie, že fuzzy prístup môže byť úspešne implementovaný do relačného prostredia bez straty výhod klasických databázových systémov. Týmto sme úspešne prezentovali návrh práce s nepresnými údajmi, ktorý je možné ďalej rozvíjať a vylepšovať.

## Zoznam použitej literatúry

Abu-Al Dahab, A., Haggag, R. M., & Abu-Al Fotouh, S. (2023). Enhancing customer relationship management using fuzzy association rules and the evolutionary genetic algorithm. *International Journal of Advanced Computer Science and Applications*, 14(4).

Bojanowska, A., & Kulisz, M. (2023). Using fuzzy logic to make decisions based on the data from customer relationship management systems. *Advances in Science and Technology Research Journal*, 17(5), pp. 270–279.

<https://doi.org/10.12913/22998624/172374>

Chen, G. Q. (1998). *Fuzzy logic in data modeling, semantics constraints, and databases design*. In A. K. Elmagarmid (Ed.), *The Kluwer international series on advances in database systems*. Kluwer Academic Publishers.

Chodorow, K. (2019). *MongoDB: The definitive guide*, 3rd ed., O'Reilly Media.

Churcher, C. (2007). *Beginning database design: from novice to professional*. New York: Apress.

Chukwu, E. G., Okoronkwo, M. C., Ezema, M. E., Achi, I. K., & Nwosu, P. C. (2024). A systematic review of fuzzy database approaches and its state of the art. *International Journal of Advanced Research in Computer and Communication Engineering*, 13(2).

<https://doi.org/10.17148/IJARCCCE.2024.13201>

Codd, E. F. (1970). *A relational model of data for large shared data banks*. *Communications of the ACM*, 13(6), pp. 377–379.

Codasyl. (1971). *Data base task group: Data base management systems standards*.

Connolly, T. M., & Begg, C. E. (2004). *Database systems: A practical approach to design, implementation, and management*, 4th ed., Addison-Wesley.

Connolly, T. M., & Begg, C. E. (2015). *Database systems: A practical approach to design, implementation, and management*, 6th ed., Pearson.

DeCandia, G., Hastorun, D., Jampani, M., Kakulapati, G., Lakshman, A., Pilchin, A., Swaminathan, S., Voshall, P., & Vogels, W. (2007). Dynamo: Amazon's highly available

key-value store. *Proceedings of the Twenty-First ACM SIGOPS Symposium on Operating Systems Principles (SOSP'07)*, pp. 205–220. <https://doi.org/10.1145/1294261.1294281>

Dezhic, E. (2018, May 21). *Fuzzy vs classical logic in AI*. Medium.

<https://edezhic.medium.com/fuzzy-vs-classical-logic-in-ai-1d8c618b2495>

Elmasti, R., & Navathe, S. B. (2016). *Fundamentals of database systems*, 7th ed., Pearson.

Farokhi, N., Vahid, M., Nilashi, M., & Ibrahim, O. (2016). A multi-criteria recommender system for tourism using fuzzy approach. *Journal of Soft Computing and Decision Support Systems*, 3(4), pp. 19–26.

Fegaras, L., & Maier, D. (1995). *Towards an effective calculus for object query languages*. Oregon Graduate Institute of Science & Technology.

Galindo, J. (2005). New characteristics in FSQL, a fuzzy SQL for fuzzy databases. *In Proceedings of the 4th WSEAS International Conference on Artificial Intelligence, Knowledge Engineering and Data Bases (AIKED'05)*, p. 8, WSEAS Press.

Galindo, J. (2008). Introduction and trends to fuzzy logic and fuzzy databases. *In J. Galindo (Ed.), Handbook of research on fuzzy information processing in databases*. IGI Global.

Galindo, J., Urrutia, A., & Piattini, M. (2005). *Fuzzy databases: Modeling, design and implementation*. Idea Group Publishing.

Galindo, J., Urrutia, A., & Piattini, M. (2006). *Fuzzy databases: Modeling, design and implementation*. Springer.

Garcia-Molina, H., Ullman, J. D., & Widom, J. (2008). *Database systems: The complete book*, 2nd ed., Pearson.

Hudec, M. (2016). *Fuzziness in information systems: How to deal with crisp and fuzzy data in selection, classification, and summarization*. Springer.

Ichimura, T., & Tachibana, I. (2018). *Emotion orientated recommendation system for Hiroshima tourist by fuzzy Petri net*. arXiv preprint. Cornell University.

Jimenez, L., Urrutia, A., Galindo, J., & Zaraté, P. (2025). *Implementation of a fuzzy relational database: Case study: Chilean cardboard industry in the Maule region*. arXiv preprint. Cornell University.

Klir, G. J. & Yuan, B. (1995) *Fuzzy sets and fuzzy logic: theory and applications*. Prentice Hall PTR.

Kolade, C. (2022, December). *Database normalization – Normal forms 1NF 2NF 3NF table examples*. freeCodeCamp. <https://www.freecodecamp.org/news/database-normalization-1nf-2nf-3nf-table-examples/>

Kim, W. (1990). *Introduction to object-oriented databases*. MIT Press.

Lertworasirikul, S. (2002). *Fuzzy data envelopment analysis (DEA)*. Idea Group Publishing.

Ma, Z. M., & Yan, L. (2010). *A literature overview of fuzzy conceptual data modeling*. *Journal of Information Science and Engineering*, 26(3), pp. 427–441.

Medina, J. M., Pons, O., & Vila, M. A. (1994). *GEFRED: A generalized model of fuzzy relational databases*. *Information Sciences*, 76, pp. 87–109.

Mego, M., Svetlovská, D., & Mardiak, J. (2014). *Translačný výskum v onkológii – prvé skúsenosti Jednotky translačného výskumu*. *Onkológia (Bratislava)*, 9(5), pp. 305–307.

Microsoft Corporation. (2024, June). *Database normalization description*. Microsoft Learn. <https://learn.microsoft.com/en-us/office/troubleshoot/access/database-normalization-description>

Oracle. (2024). *MySQL 8.0 Reference Manual – SQL Statement Syntax*. <https://dev.mysql.com/doc/refman/8.0/en/sql-statements.html>

Rajeev, S. (2015). Fuzzy data and its applications. *International Journal of Engineering Research and Applications*, 5(6), pp. 76–79.

Redgate Software. (2025, April). *DB-Engines ranking*. <https://db-engines.com/en/ranking>

- Scott, G. (2023). *Fuzzy logic: Definition, meaning, examples, and history*. Investopedia. <https://www.investopedia.com/terms/f/fuzzy-logic.asp>
- Shields, W. (2019). *SQL QuickStart guide: The simplified beginner's guide to managing, analyzing, and manipulating data with SQL*. ClydeBank Media LLC.
- Silberschatz, A., & Kedem, Z. (1980). *Consistency in hierarchical database systems*. *Journal of the ACM*, 27(1), pp. 72–80.
- Štrbo, M. (2020). *Modelovanie databázových systémov*. Trnavavská univerzita v Trnave: Katedra matematiky a informatiky, Pedagogická fakulta Trnavskej univerzity.
- Visconti, E., Anderson, D. T., & Kerley, J. (2023). Fuzzy database for language-driven procedurally generated simulated datasets. *Applied Imagery Pattern Recognition Workshop (AIPR)*, pp. 1–11. IEEE. <https://doi.org/10.1109/AIPR60534.2023.10440714>
- Vučetić, M. (2011). Functional dependencies analyse in fuzzy relational database models. *Journal of Information Technology and Applications*, 1(2), pp. 90–104.
- Wang, Y., & Liu, X. (2009). *Fuzzy conceptual data modeling in fuzzy systems*. Springer, pp. 427–446.
- Zadeh, L. A. (1965). *Fuzzy sets*. *Information and Control*, 8(3), pp. 338–353. [https://doi.org/10.1016/S0019-9958\(65\)90241-X](https://doi.org/10.1016/S0019-9958(65)90241-X)
- Zadeh, L. A. (1975). The concept of a linguistic variable and its application to approximate reasoning. *Information Sciences*, 8(3), pp. 199–249.
- Ziaei Ghahnavieh, M., Habibi Manesh, H., & Sheikmoradi, S. (2024). Comparative comparison of fuzzy logic and classical logic. *International Journal of Nonlinear Analysis and Applications*, 15(12), pp. 435–452. <https://doi.org/10.22075/ijnaa.2024.32978.4905>