



VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
BRNO UNIVERSITY OF TECHNOLOGY

FAKULTA INFORMAČNÍCH TECHNOLOGIÍ
FACULTY OF INFORMATION TECHNOLOGY

MODEL INFORMAČNÍHO SYSTÉMU – HOTEL
PROJECT TITLE

DATABÁZOVÉ SYSTÉMY
COURSE

AUTORI **TOMÁŠ KAZÍK, DENIS HELIENEK**
AUTHORS

BRNO 2018

Kontrolná sekcia dokumentu

Názov dokumentu	xhelie00_xkazik03.pdf
Dátum vytvorenia	7.3. 2018
Dátum poslednej úpravy	25.4. 2018
Počet strán	8
Abstrakt Tento dokument bol vytvorený za účelom obhajoby modelu informačného systému hotelu, ktorý je projektom do predmetu Databázových systémov.	

Dátum	Verzia	Info
7.3. 2018	1.0	Vytvorenie dokumentu
25.4. 2018	1.1	Pridané posledné kapitoly pre 4. a 5. časť projektu

Meno	Rola
Denis Helienek	Autor dokumentu
Tomáš Kazík	Autor dokumentu

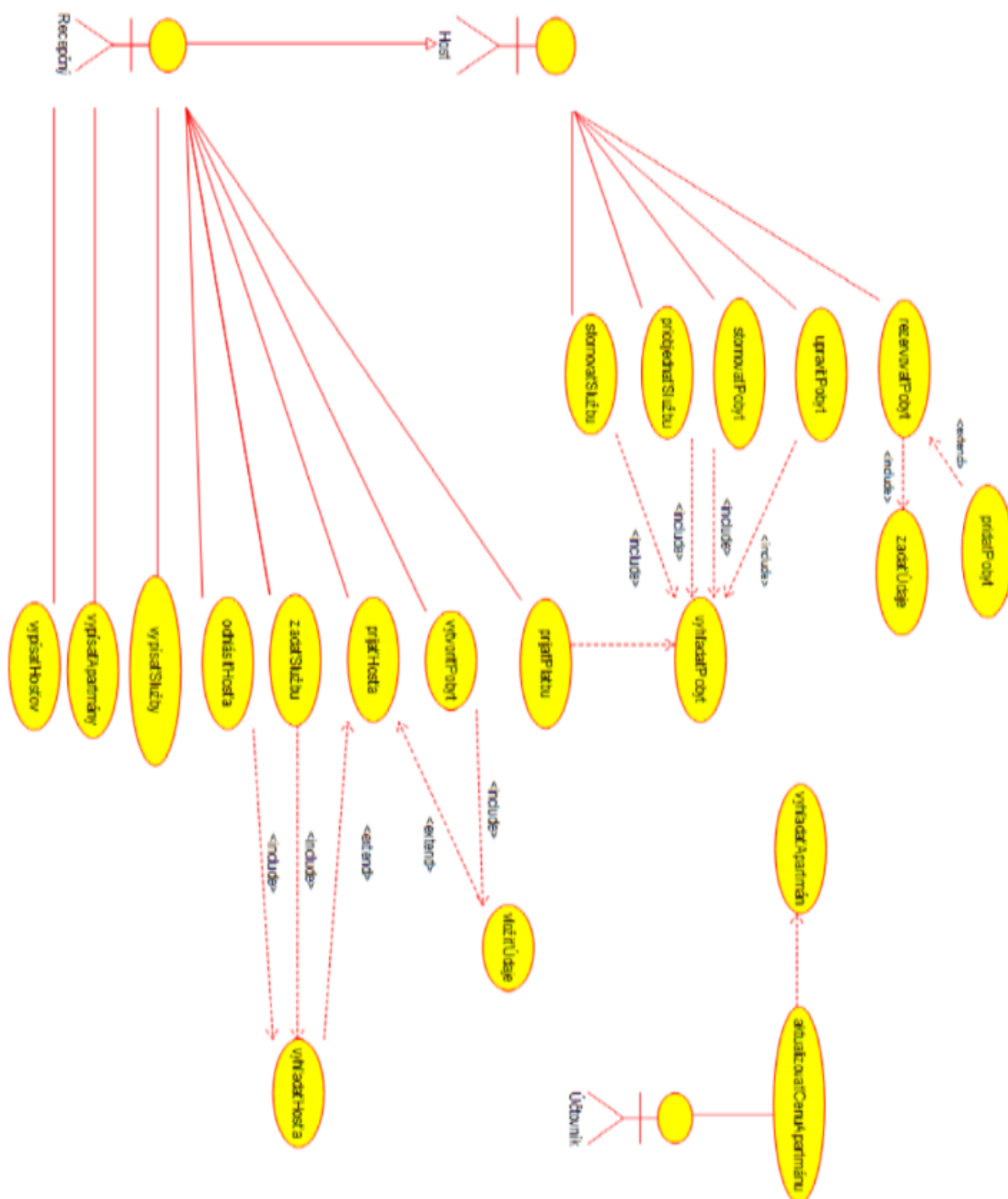
Obsah

1. Zadanie.....	4
2. Use case diagram.....	5
3. ER Diagram.....	6
4. Implementácia.....	7
4.1. Triggery.....	7
4.2. Procedúry.....	7
4.3. Explain plan.....	7
4.4. Materializovaný pohľad.....	8
5. Záver.....	8

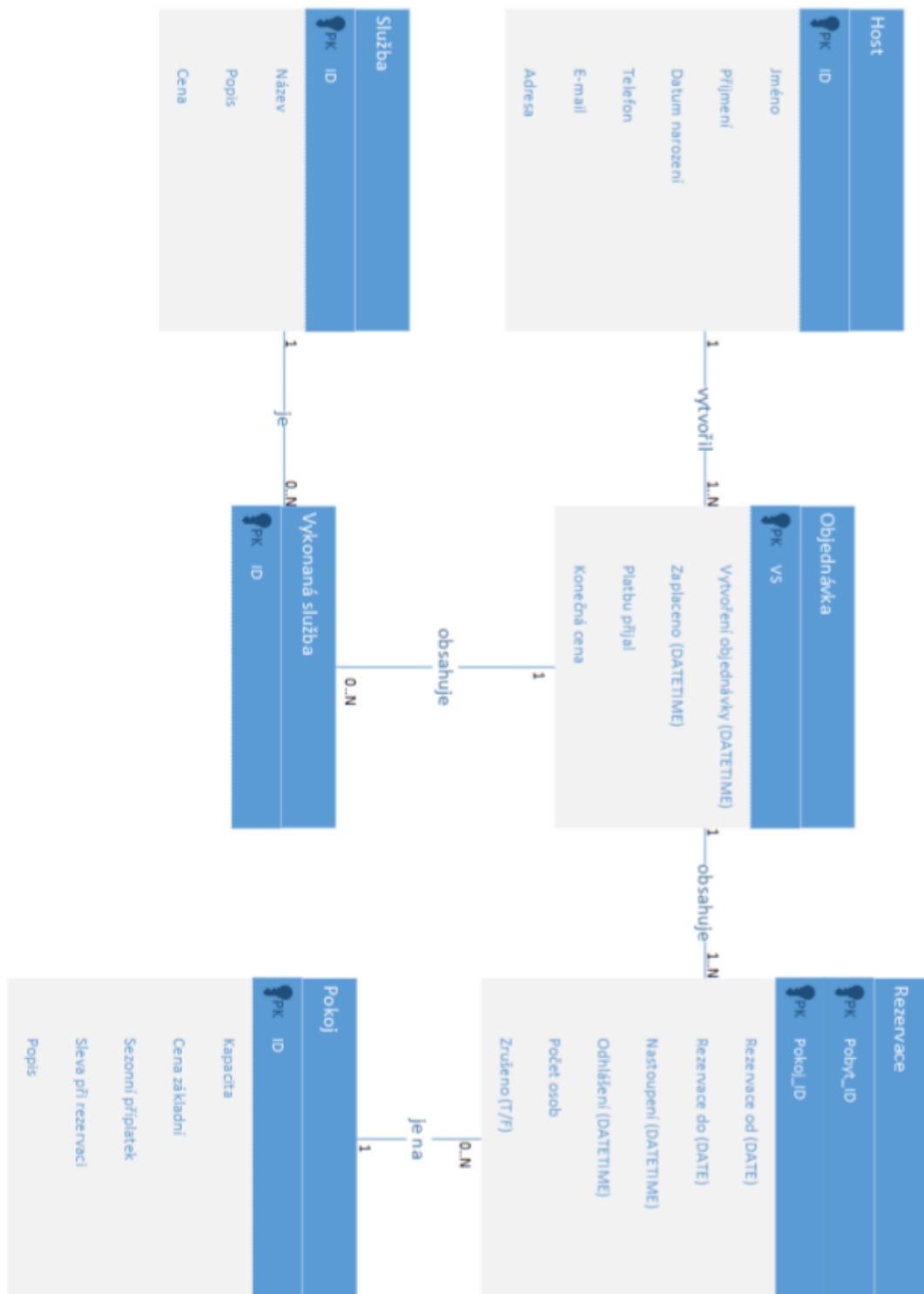
1. Zadanie

Navrhněte IS hotelu, který by poskytoval přehled o dostupnosti pokojů, údaje o hostech, jejich pobytech v hotelu, požadavcích na služby, platby za pokoje, atd. Zákazníci mohou provádět rezervace pokojů (tím pádem musí zadat své osobní údaje). Klient si může v rámci jedné rezervace objednat více pokojů, třeba i na jiné datum. Hotelu stačí mít informace pouze o jednom klientovi, který zaštiťuje celý pobyt v hotelu (o ostatních účastnících pobytu nemusí být dostupné žádné informace). Pobyt může být vytvořen na základě rezervace, nebo k rezervaci pokoje vůbec nemusí dojít, pokud klient přijde přímo na recepci hotelu. Jednotlivé typy pokojů mají různé ceny, cena pokoje se navíc může lišit podle období (turistická sezóna), na které si klient pokoj objednává. Cena pokoje se zároveň odvíjí od toho, zda si host pokoj rezervuje dopředu, nebo až na místě. Klient si může přiobjednat k danému pobytu služby v hotelu navíc, jako např. pronájem bazénu, a to i vícekrát. IS dále bude schopen evidovat skutečně absolvované pobyty - tj. které pokoje klient od kdy do kdy obýval a jaké služby skutečně využil (od x do x naKteremPokoji). U každého pobytu je evidováno, který zaměstnanec převzal platbu za pobyt.

2. Use case diagram



3. ER Diagram



4. Implementácia

Najskôr sú vymazané predošlé tabuľky a trigger. Následne sú vytvorené tabuľky, trigger, a procedúry. Ku koncu je vytvorený explain plan a materializovaný pohľad. V skripte sa taktiež nachádzajú okomentované selecty zobrazujúce ukázkové dáta.

4.1. Triggery

V skripte je 6 triggerov. Tieto trigger sú implementované pomocou konštrukcie SEQUENCE. Niektoré majú viac ukázkovú formu, ako napríklad trigger na kontrolu dĺžky mena a priezviska, ktoré kontrolujú či dĺžka nieje menšia ako 1. Taktiež sa v skripte nachádza trigger, ktorý kontroluje predvoľbu čísla, ktorá musí byť 421.

4.2. Procedúry

V implementaci se nachází 4 procedury. Dvě z nich jsou menší a určené pro využití v triggerech jako usnadnění práce s SQL příkazy. Zbylé dvě jsou obsáhlejší, kde první z nich slouží k výpočtu procentuálního podílu pokoje, zadaného pomocí jeho ID, na veškerých rezervacích pokojů a druhá procedura počítá, kolik klient, zadaný pomocí jeho ID, celkově zaplatil za všechny své návštěvy. Obě procedury využívají CURSOR, který pomocí cyklu prochází a zjišťují, které řádky splňují kritérium. Poté výsledek vypíše na výstup.

4.3. Explain plan

Explain plan spája tabuľku HOST a OBJEDNAVKA a využíva agregáčnú funkciu GROUP BY. Tento dotaz je implementovaný dva krát, kde druhý raz je využitý index. Tieto dva dotazy nám umožňujú znázorniť spracovanie toho istého dotazu po optimalizácii a bez.

PLAN_TABLE_OUTPUT

Plan hash value: 1927733912

Id	Operation	Name	Rows	Bytes	Cost (%CPU)	Time
0	SELECT STATEMENT		25	1000	13 (8)	00:00:01
1	HASH GROUP BY		25	1000	13 (8)	00:00:01
2	MERGE JOIN CARTESIAN		25	1000	12 (0)	00:00:01
3	TABLE ACCESS FULL	OBJEDNAVKA	5	65	3 (0)	00:00:01
4	BUFFER SORT		5	135	10 (10)	00:00:01
5	TABLE ACCESS FULL	HOST	5	135	2 (0)	00:00:01

Bez indexu

PLAN_TABLE_OUTPUT

Plan hash value: 3102556400

Id	Operation	Name	Rows	Bytes	Cost (%CPU)	Time
0	SELECT STATEMENT		25	1000	7 (15)	00:00:01
1	HASH GROUP BY		25	1000	7 (15)	00:00:01
2	MERGE JOIN CARTESIAN		25	1000	6 (0)	00:00:01
3	TABLE ACCESS FULL	OBJEDNAVKA	5	65	3 (0)	00:00:01
4	BUFFER SORT		5	135	4 (25)	00:00:01
5	INDEX FAST FULL SCAN	I	5	135	1 (0)	00:00:01

S indexom

Z vyššie uvedených výstup je možné vidieť, že sa hodnota Cost znížila. Takmer všetky operácie až na poslednú sú rovnaké.

Operácia s indexom 0 naznačuje, že sa vykonal dotaz. Druhá operácia naznačuje, že sa zoskupili položky podľa hashovacieho kľúča. Merge join cartesian je náročná operácia, ktorá má zmysel pri malom počte riadkov. Následný Table access full je prechodom celou tabuľkou. Štvrtý index ukazuje bufferovaciu techniku. Najväčší rozdiel môžeme vidieť pri posledných operáciach kedy je prvý raz využitý celý prechod tabuľkou a druhý raz pomocou indexu.

4.4. Materializovaný pohľad

Pohled je ve skriptu využit pro usnadnění přístupu k celkovému množství financí, které pracovníci přijali od klientů. Po vypracování dotazu je uložen v cache paměti jako zamestnanecPrijalCelkem a poté stačí dotaz na výpis obsahu daného pohledu a není třeba dalších specifikací.

5. Záver

Projekt bol implementovaný za použitia dvoch rôznych prostredí a to pomocou Datagrip a SQL Developer. Spúšťaný bol na fakultnom serveri Oracle.