

VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
FAKULTA INFORMAČNÍCH TECHNOLOGIÍ

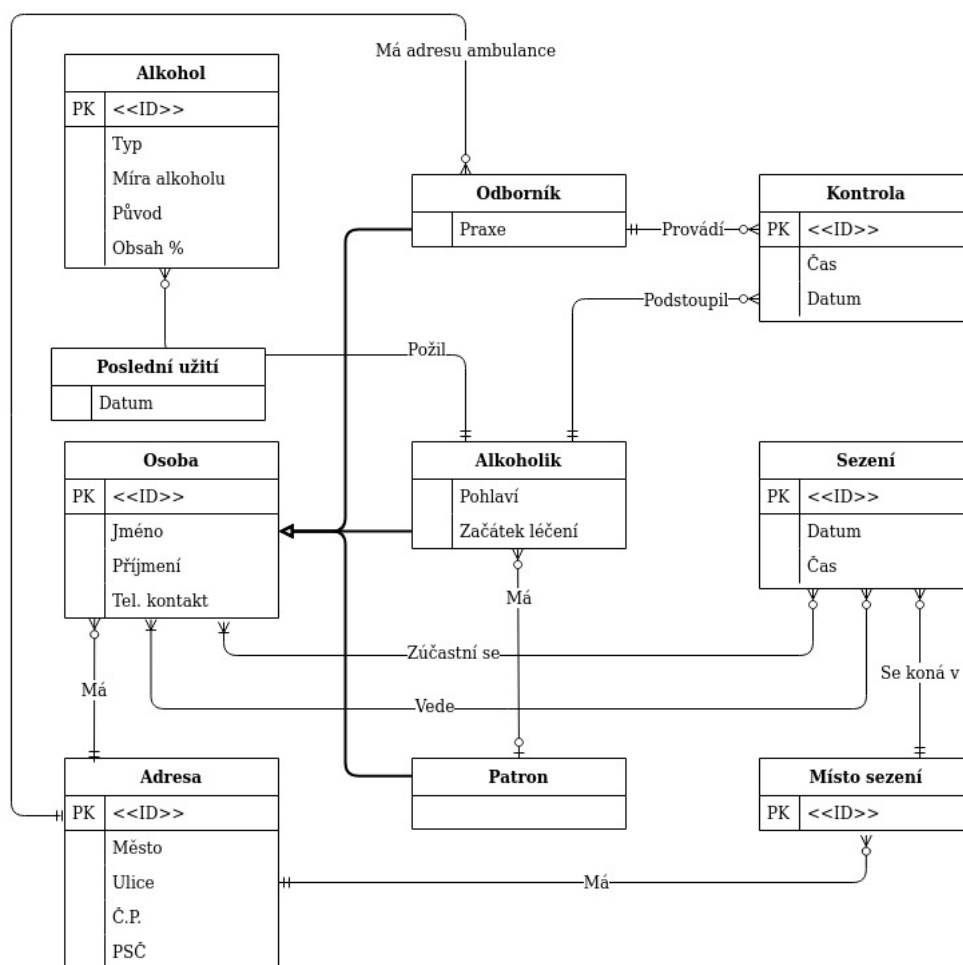
Databázové systémy
2019/2020

KLUB ANONYMNÍCH ALKOHOLIKŮ

1 Zadání

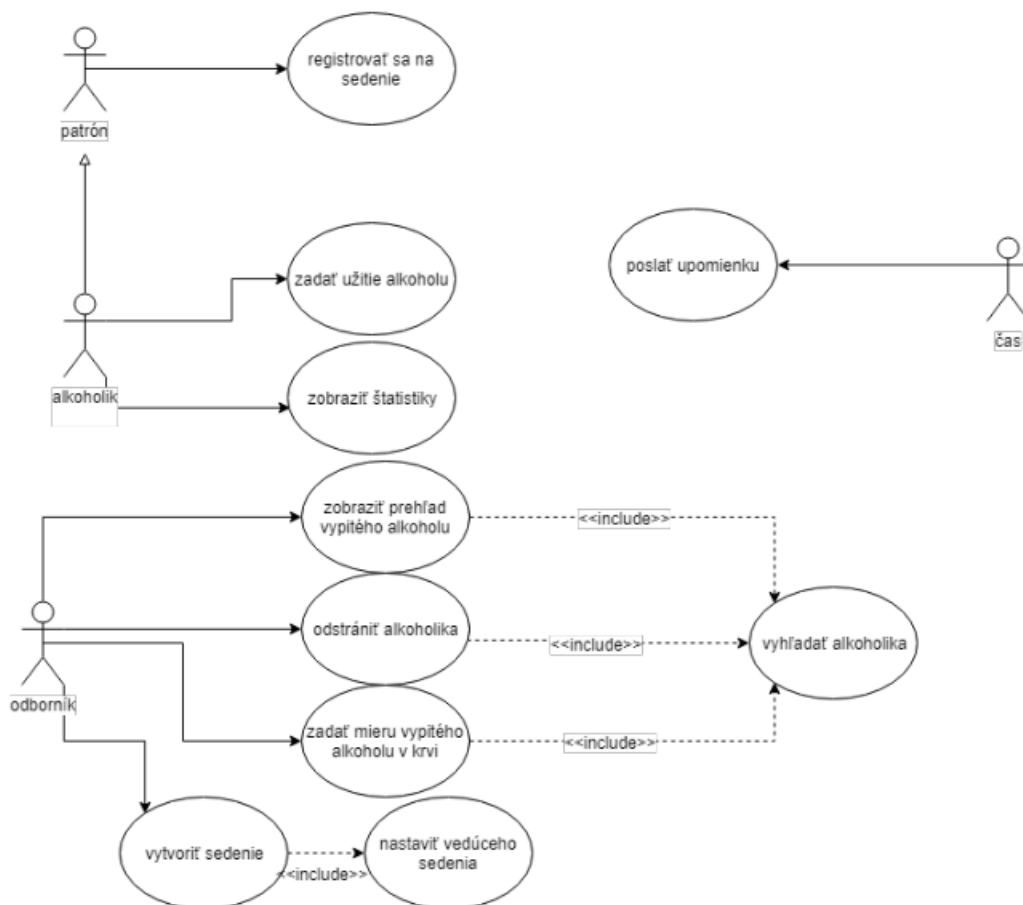
Navrhněte informační systém, který bude podporovat anonymní alkoholiky k organizaci sezení a evidenci vypitého alkoholu. Systém uchovává základní informace o alkoholících, jako je jejich věk, pohlaví, patrony, kteří je podporují a se kterými se nepravidelně scházejí na různých místech a v různých datech a rovněž i informace o odbornících, kteří na ně lékařsky dohlíží. Odborníci musí mít patřičnou expertízu pro pečování o alkoholiky, a mít minimální lékařskou praxi, která je v systému evidována. Patronem však může být kdokoliv. Pravidelně se konají sezení, kterých se účastní až dvanáct alkoholiků a navíc můžou být přítomni jak patroni tak i odborníci a dohlížet nad diskuzí. U každého sezení nás zajímá datum, čas, a místo konání. Těchto míst je pouze několik oficiálních a dedikovaných. Každé sezení je vedeno jednou osobou. Neformální schůzky s patrony však mohou být organizovány v libovolné lokalitě. Alkoholici se musí alespoň třikrát ročně účastnit nějakého sezení, a v případě, že se více jak tři měsíce nedostaví na žádné sezení je jim systémem zaslána upomínka. U alkoholiků jsou pravidelně (i nepravidelně a nečekaně) prováděny kontroly odborníky, na kterých se měří míra alkoholu v jejich krvi. Tato míra vypitého alkoholu je pak evidována do systému, rovněž s původem a typem vypitého alkoholu. Alkoholici však mohou sami zaevidovat (ze špatného svědomí), že alkohol požili (tedy mimo prováděné kontroly) a tuto informaci rovněž přidat do systému.

2 ER diagram



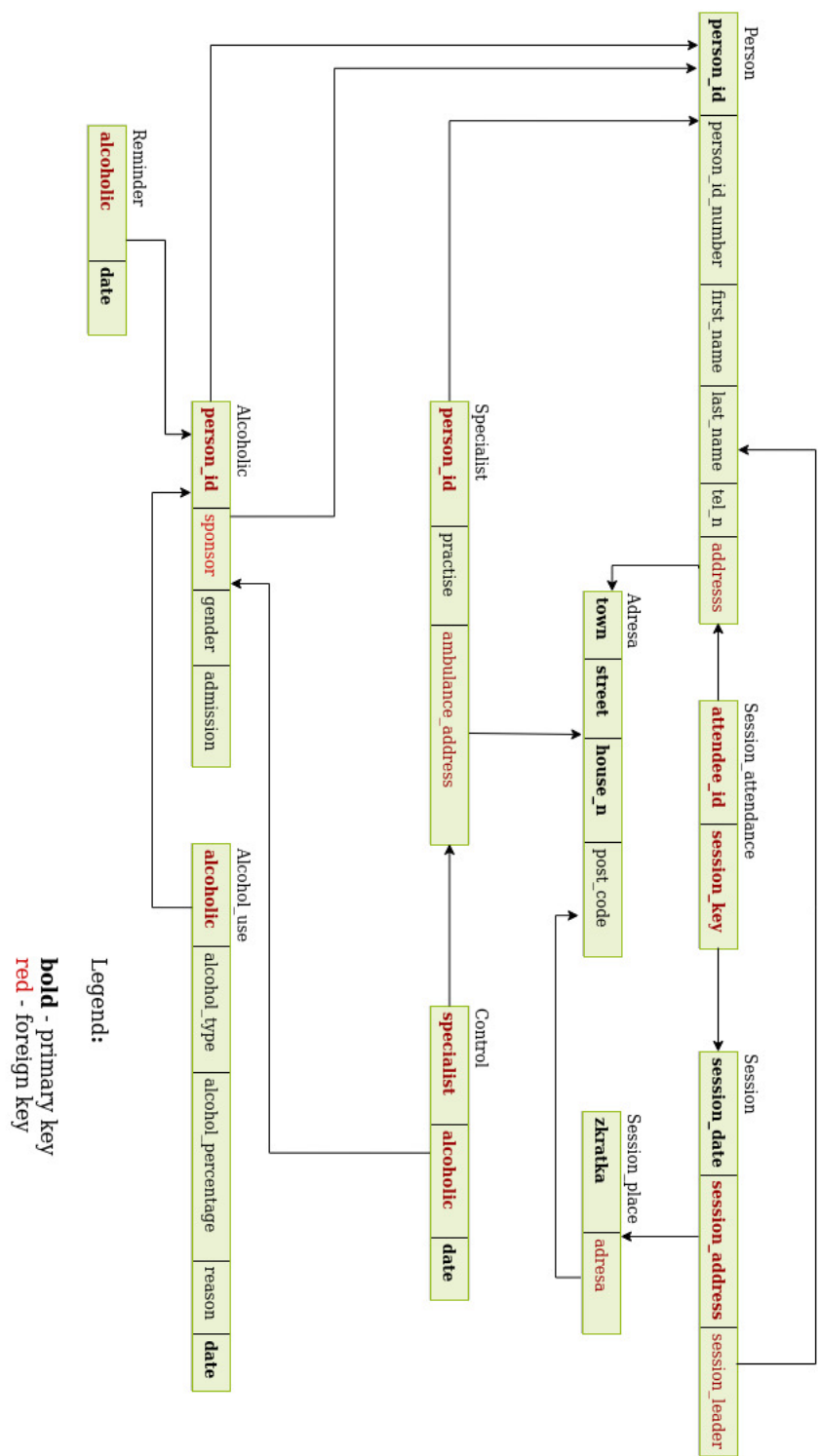
Obrázek 1: ERD

3 UC diagram



Obrázek 2: UCD

4 Návrh databáze



Obrázek 3: DB

5 Implementace

5.1 Generalizace - Specializace

Generalizaci specializaci jsme využily u vztahu `osoba` \leftarrow `alkoholik`, `specialista`, `sponzor`. Máme zde jednu tabulku pro nadtyp `osoba` a tři tabulky pro podtypy.

5.2 Trigger

Všechny trigery jsou spuštěny při vložení, nebo modifikaci řádku.

Check_person_id_number

Tento trigger kontroluje zadaný formát rodného čísla osoby v tabulce `PERSON`. Konkrétně podle délky, roku, dne, měsíce a dělitelnosti jednácti.

V případě špatného formátu je vyhozena chyba číslo -20001.

Check_session_collision

Trigger kontrolující kolize sezení v tabulce `SESSION`.

Za kolizi se považuje sezení, které má stejné místo a datum se liší maximálně o hodinu. Pokud nastane kolize, je vyhozena chyba číslo -20001.

Generate_user_id

Trigger na autoinkrementaci jsme použili u tabulky `Person`, který má stejnou funkci jako `GENERATED AS IDENTITY NOT NULL` - generuje počínaje jedničkou unikátní ID.

5.3 Procedury

Count_alcoholics_at_session (p_session_name number)

Dle zadání je maximální počet alkoholiků na sezení dvanáct. Tato procedura tiskne počet alkoholiků přihlášených na daném sezení (`p_session_name`). Pokud sezení není nalezeno, vytiskne se "Session has not been found."

Add_reminder (p_months_num number)

V zadání je specifikováno, že pokud se alkoholik nezúčastní sezení tři měsíce, má mu být zaslána upomínka. K vytvoření upomínek právě slouží tato procedura.

U každého alkoholika se zkontroluje, zda v posledních `n` měsících (`p_months_num`) navštívil sezení. Pokud ne, je vygenerována nová upomínka.

5.4 Indexy

Ke každé tabulce jsme přidaly indexy na primární klíče. Podle primárních klíčů se totiž většinou v tabulkách vyhledává a proto se tímto vyhledávání urychlí.

5.5 EXPLAIN PLAN

V našem kódu explain plan zobrazuje sekvenci operací u příkazu `SELECT` přes dvě tabulky.

Select zobrazí seznam alkoholiků, kteří požili alkohol a zároveň kolikrát sa to stalo. Po vytvoření indexu `alcohol_use_i` by mělo být vyhledávání rychlejší.

| ID | Operation | Name | Rows | Bytes | Cost | Time |
|-----|-----------------------------|----------------|------|-------|--------|----------|
| 0 | SELECT STATEMENT | | 1 | 18 | 6 (34) | 00:00:01 |
| 1 | SORT ORDER BY | | 1 | 18 | 6 (34) | 00:00:01 |
| * 2 | FILTER | | | | | |
| 3 | HASH GROUP BY | | 1 | 18 | 6 (34) | 00:00:01 |
| 4 | NESTED LOOPS | | 1 | 18 | 4 | 00:00:01 |
| 5 | NESTED LOOPS | | 1 | 18 | 4 | 00:00:01 |
| 6 | TABLE ACCESS FULL | ALCOHOL_USE | 1 | 3 | 3 | 00:00:01 |
| * 7 | INDEX UNIQUE SCAN | SYS_C001168092 | 1 | 0 | 0 | 00:00:01 |
| 8 | TABLE ACCESS BY INDEX ROWID | PERSON | 1 | 15 | 1 | 00:00:01 |

| ID | Operation | Name | Rows | Bytes | Cost | Time |
|-----|-----------------------------|----------------|------|-------|--------|----------|
| 0 | SELECT STATEMENT | | 1 | 18 | 4 (50) | 00:00:01 |
| 1 | SORT ORDER BY | | 1 | 18 | 4 (50) | 00:00:01 |
| * 2 | FILTER | | | | | |
| 3 | HASH GROUP BY | | 1 | 18 | 4 (50) | 00:00:01 |
| 4 | NESTED LOOPS | | 1 | 18 | 2 | 00:00:01 |
| 5 | NESTED LOOPS | | 1 | 18 | 2 | 00:00:01 |
| 6 | INDEX FULL SCAN | ALCOHOL_USE | 1 | 3 | 1 | 00:00:01 |
| * 7 | INDEX UNIQUE SCAN | SYS_C001168092 | 1 | | 0 | 00:00:01 |
| 8 | TABLE ACCESS BY INDEX ROWID | PERSON | 1 | 15 | 1 | 00:00:01 |

2 filter(COUNT(*)>0)

7 access(P.PERSON_ID=AU.ALCOHOLIC)

Vyhledávání by se dalo teoreticky ještě urychlit s pomocí materializovaného pohledu.

5.6 Přístupová práva

Přístupová práva jsme využily pro přístup druhého člena k databázi.

5.7 Materializovaný pohled

Materializovaný pohled zobrazuje alkoholiky a jejich počet kontrol.

Pohled je vytvořen okamžitě (`BUILD IMMEDIATE`).

Je možné ho aktualizovat pomocí příkazu (`DBMS_MVIEW.REFRESH('alcoholic_control_count')`).

6 Závěr

Pro přístup k databázi jsme využívali DataGrip od JetBrainsu.

Projekt se vyplatilo dělat souběžně s přednáškami, protože si člověk hned vyzkoušel probrané učivo v praxi.

7 Zdroje

<https://www.w3schools.com/sql/>

<https://www.cssz.cz/web/cz/standardni-kontrola-rodneho-cisla-a-evidencni-ho-cisla-pojistence>

<https://docs.oracle.com/database/121/index.htm>