

Vysoké učení technické v Brně

Fakulta informačních technologií



Databázové systémy - IDS

Dokumentácia

Téma č. 19

Přihlašování na projekty

Autor:

Marcel Feiler (xfeile00)

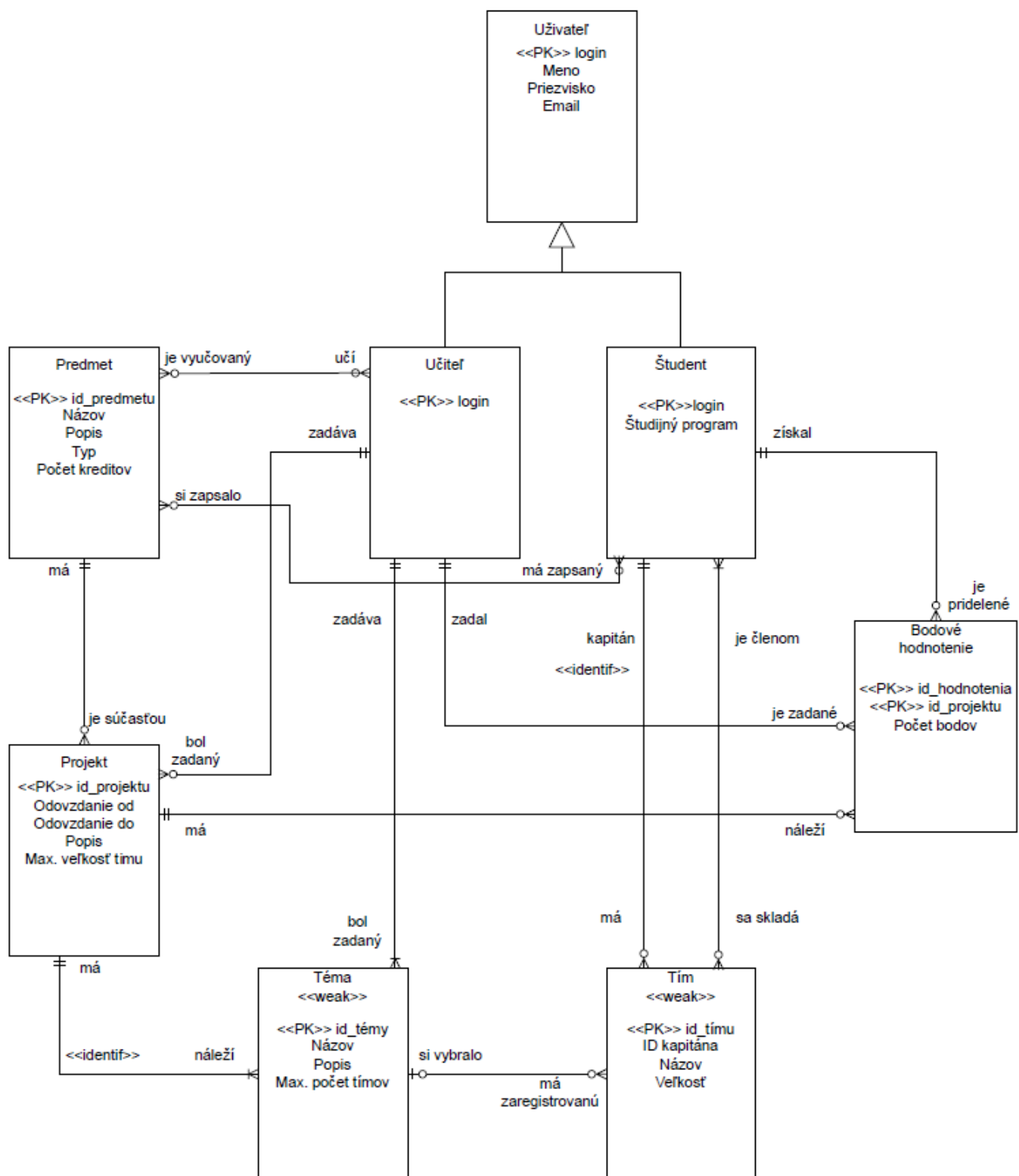
Zadanie témy

Navrhňte aplikáciu, ktorá umožní študentom prihlasovať sa na projekty z predmetov. Systém musí umožniť vložiť informácie o projektoch, vrátane veľkosti tímov a maximálneho počtu tímov, ktoré sa môžu na jednu tému prihlásiť. Učiteľovi musia poskytovať prehľad obsadenia projektov a musia umožniť zadať bodové hodnotenie jednotlivých študentov.

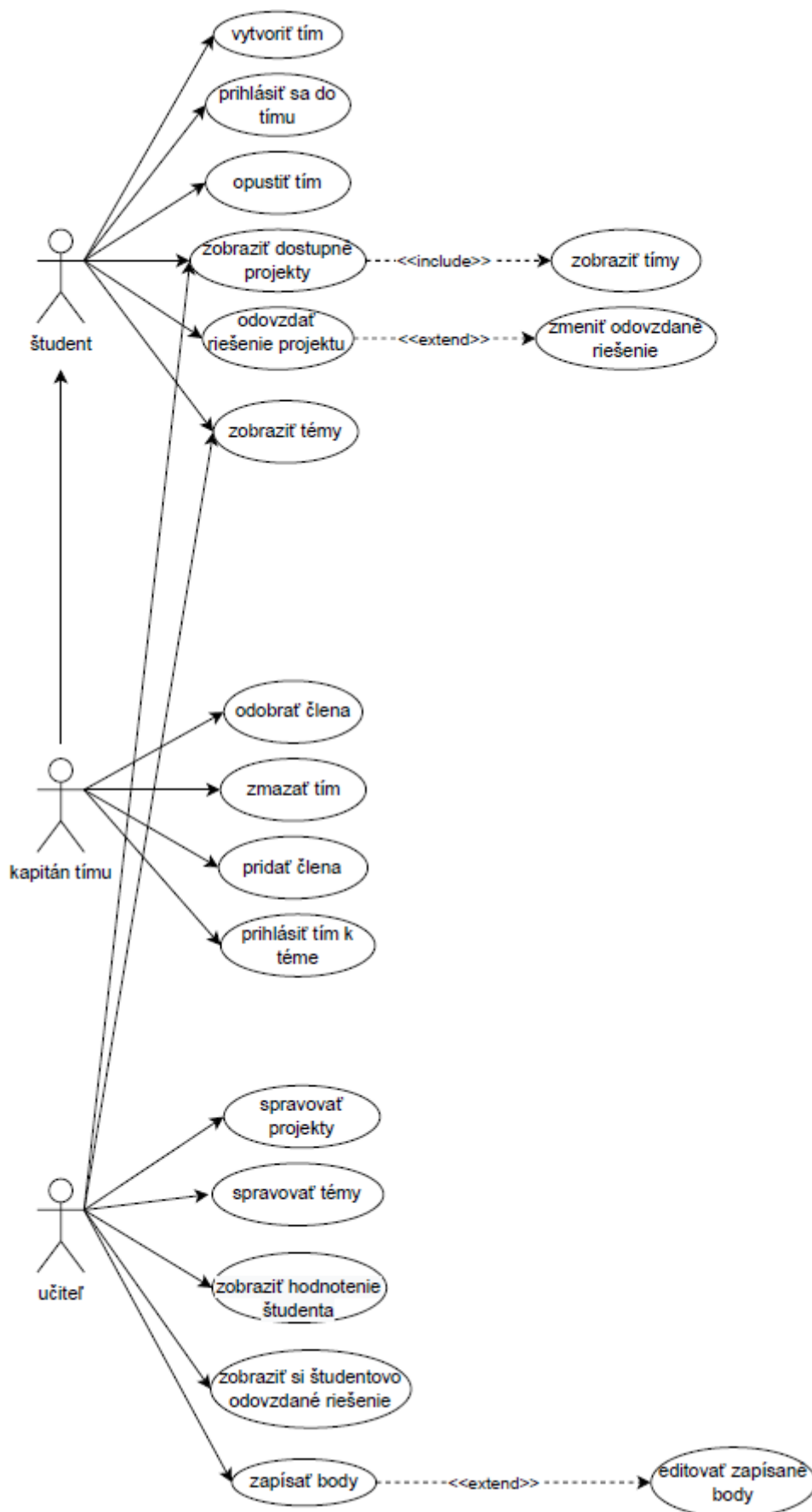
Realizácia ER diagramu

Pre svoj dátový model sme sa rozhodli vytvoriť celkom 8 typov entít, kde sme sa rozhodli špecializovať typ entity užívateľ na študent a učiteľ. Učiteľ môže, ale nemusí vyučovať predmet. Každý študent môže mať zapísaný nejaký predmet, ktorého súčasťou sú projekty. Tie sú zadané a špecifikované učiteľom. Projekt musí mať vždy na výber aspoň jednu tému, ktorú si následne volí každý tím, ktorý zastupuje kapitán. Každý študent si buď môže založiť vlastný tím a automaticky bude kapitán, alebo sa pripojí do nejakého tímu so statusom člena tímu. Projektu náleží určité bodové hodnotenie a to je pridelené každému študentovi zvlášť. Bodové hodnotenie zadáva učiteľ.

Vzhľad ER diagramu



UC diagram



Základné objekty schématu databázy

Pri vytvorení základných objektov sme potrebovali implementovať nasledujúce tabuľky pre:

- užívateľa (uzivatel)
- učiteľa (ucitel)
- študenta (student)
- predmet (predmet)
- tému (tema)
- project (projekt)
- tím (tým)
- bodové hodnotenie (bodove_hodnoceni)
- predmet a študenta (predmet_student) kvôli vzťahu M:N
- predmet a učiteľa (predmet_ucitel) kvôli vzťahu M:N

Tie sme následne vyplnili prvotnými dátami.

SQL dotazy

Ako hlavné dotazy, ktoré sme chceli realizovať sme si najskôr určili ich slovnú reprezentáciu a tá bola nasledovná:

- Ktorí študenti majú zapísaný predmet IZP? (2 tabuľky)
- Ktorí učitelia vyučujú predmet IFF? (2 tabuľky)
- Ktorým študentom (xlogin) zapísal učiteľ s loginom xkrena00 aspoň 10 bodov z nejakého projektu? (3 tabuľky)
- Koľko projektov zadal učiteľ s loginom xkrena00 z predmetu IFF? (využitie klauzule GROUP BY)
- Koľko bodov má celkovo zo všetkých predmetov študent s loginom xnovak00? (využitie klauzule GROUP BY)
- Ktorí študenti majú zapísaný len predmet IFF? (využitie predikátu EXISTS)
- Ktorí študenti sú v time sami? (využitie predikátu EXISTS)
- Ktoré projekty sa odovzdávajú v apríli? (predikát IN)

Po implementácii spomenutých selektov sme sa presunuli na poslednú časť.

Skript pre vytvorenie pokročilých objektov schématu databázy

Trigger č. 1

Daný trigger kontroluje správnosť dátumu odovzdania novo pridaného projektu. Po vložení nového projektu porovná dátum odovzdania so súčasným dátumom, v prípade ak zadaný dátum už nie je aktuálny, vypíše sa chybová hláška: „Datum odevzdani nesmi byt v minulosti.“

Trigger č. 2

Po aktualizácii počtu kreditov sa aktivuje trigger, ktorý vypíše neaktuálny, aktuálny počet kreditov daného predmetu a rozdiel medzi novou a starou hodnotou v tvare: „Pocet kreditov predty: ... Pocet kreditov teraz: ... Rozdiel v pocte kreditov: ...“

Procedúra č. 1

Nasledujúca procedúra po jej vykonaní vypočíta a vypíše celkový počet tímov, projektov a užívateľov s tým, že pre rýchlejší prístup k užívateľom sme použili kurzor kde sme ho definovali ako výber mien užívateľa. Použili sme ešte jednu premennú jmeno_uziv, ktorá nám neskôr pomôže v LOOP sekcii pri príkaze FETCH a následnom inom spôsobe inkrementácie a celkového spočítania počtu užívateľov, nie pomocou tradičného COUNT(*) ako tomu bolo pri ostatných výsledkoch. Čo sa týka output časti, procedúra nám vypíše: „Počet tímov: ... Počet projektov: ... Počet užívateľov ...“

Procedúra č. 2

V tejto procedúre počítame a vypíšeme priemerný bodový zisk na študenta v danom projekte. Používame pomocné premenné typu NUMBER pre uloženie hodnoty sumy bodov a celkového počtu študentov v danom projekte. Pri implementácii sme využili EXCEPTION v prípade ZERO_DIVIDE, pri ktorej vypíše chybovú hlášku: „Chyba: deleni nulou.“ Ak takýto prípad nenastane, na output sa vypíše správa: „Prumerny pocet bodu za projekt ..(projekt_id).. je ..(prumerny_pocet_bodu)..“

Explain plan

Výsledok pred vytvorením indexu:

Plan hash value: 3850025777

Id	Operation	Name	Rows	Bytes	Cost (%CPU)	Time
0	SELECT STATEMENT		1	12	3 (0)	00:00:01
1	SORT GROUP BY NOSORT		1	12	3 (0)	00:00:01
* 2	TABLE ACCESS FULL	BODOVE_HODNOCENI	2	24	3 (0)	00:00:01

Predicate Information (identified by operation id):

2 - filter("BODOVE_HODNOCENI"."XLOGIN_FK"='xnovak00')

Výsledok po vytvorení indexu:

Plan hash value: 1637535460

Id	Operation	Name	Rows	Bytes	Cost (%CPU)	Time
0	SELECT STATEMENT		1	12	2 (0)	00:00:01
1	SORT GROUP BY NOSORT		1	12	2 (0)	00:00:01
2	TABLE ACCESS BY INDEX ROWID	BODOVE_HODNOCENI	2	24	2 (0)	00:00:01
* 3	INDEX RANGE SCAN	XLOGIN_INDEX	2		1 (0)	00:00:01

Predicate Information (identified by operation id):

3 - access("BODOVE_HODNOCENI"."XLOGIN_FK"='xnovak00')

Využitie indexácie

Po vytvorení indexu sa zmenšila celková cena operácia z 9 jednotiek na 7. Zaindexovaný bol foreign key xlogin studenta z tabuľky bodove_hodnoceni.

Cost (cena) je odhadovaná náročnosť operácie, ktorá úzko súvisí s dĺžkou prevádzania tejto operácie. Na databáze s veľkým množstvom dát by sa tento rozdiel pravdepodobne značne prejavil v kolónke Time.

Materialized view

V odovzdanom skriptu je ukážka použitia z pohľadu druhého študenta. Tento študent má udelený prístup k tabuľkám kolegu, z ktorých následne vytvorí mat. pohľad, ktorý sa dotazuje na to, ktorí študenti majú zapísaný iba predmet IFF. Po vytvorení mat. pohľadu je k nemu udelený prístup prvému členovi tímu (autorovi a vlastníkovi tabuliek).