

Pokyny pro vypracování projektu

Radim Bača

18. září 2023

1 Téma projektu

Zadání by mělo být originální a netriviální. Zkuste se vyhnout zadání:

- o kterém máte minimální znalosti (např. půjčky v bance, evidence zaměstnanců v řetězci fast-foodů),
- kde by se celé řešení dalo jednoduše nahradit nějakým content management systémem (např. databáze letadel 2. světové války a jejich nasazení).

Vyberte si zadání, které je Vám nějakým způsobem **blízké**, máte o něm hlubší znalosti, zároveň jsou předmětem **dynamická data** a aplikace pomůže řešit nějaký problém. Příkladem může být situace, kdy se podílíte na organizaci/pořádání nějaké události a aplikace Vám může tuto organizaci usnadnit. Pokud si nejste vhodností svého tématu jistí, zkonzultujte jej se svým cvičícím.

Konceptuální model každého projektu musí obsahovat minimálně pět entitních typů a pět vazeb. Maximální počet tabulek výsledného relačního modelu je deset. Za optimální počet považujeme zhruba sedm tabulek.

Myslete na to, že v budoucnu (v jiných předmětech jako DS II) se po Vás bude chtít nad datovým modelem podrobně rozebrat netriviální funkce. Netriviální funkcí jsou operace, které nejsou pouze CRUD operace (Create, Read, Update, Delete). Takže tím, že nyní vymyslíte netriviální model, si ulehčíte práci v dalším předmětu.

Projekt musí student vypracovat samostatně. Jakékoli odhalení pokusu o odevzdání plagiátu nějakého jiného projektu, bude mít za následek neudělení zápočtu a postoupení přestupku před komisi. Naopak se doporučuje postup řešení konzultovat s kolegy (ve smyslu porovnání použitých technik a jejich aplikace na daný problém). Každý student musí být schopen o svém řešení plynule mluvit a zodpovědět případné dotazy během závěrečné prezentace. Použití stejného projektu napříč více předměty je povoleno, samozřejmě při splnění podmínek projektu pro daný předmět.

2 Rozdělení projektu

Projekt bude rozdělen do několika dílčích kroků včetně prezentace na posledním cvičení. Každý krok má dva termíny odevzdání. Každý termín je povinný a nebude možné odevzdat v druhém termínu pokud jste v prvním neodevzdali nic! Odevzdání bude probíhat **přes stránky dbedu.cs.vsb.cz**. Po přihlášení se na stránce objeví menu **Profil**, kde naleznete položku **Odevzdat projekt**. Vyberete **předmět Databázové systémy I, projekt Datová analýza pro informační systém** a následně jeden z dílčích úkolů:

- Specifikace zadání & Datové modely
(termíny 12.11. / 26.11.)
(10b)
- SQL skripty (termín 3.12.)
(15b)
- Formální analýza (termín 4.12.) – nepovinný dílčí úkol projektu
(2b)
- Závěrečná prezentace projektu (v 13. týdnu na cvičení)
(3b)
- Finální verze (termín v den prezentace)

Každý dílčí úkol kromě formální analýzy je **povinný**. Není tedy možné bez omluvy například nedojít na prezentaci, nebo neodevzdat specifikaci

a datové modely a očekávat udělení bodů za projekt. Pokud byly během prezentace nalezeny nedostatky a budete vyzváni k doděláním, pak není možné odevzdávat textovou část bez těchto oprav.

2.1 Specifikace

Specifikace bude obsahovat následující části:

- MOTIVACE – stručné uvedení do kontextu a jasný popis toho, jaký problém Vám aplikace pomůže řešit.
- KDO – budou alespoň dva typy uživatelů (administrátor aplikace se nepočítá). Tito uživatelé pak budou zahrnuti v dalších úvahách o vstupech, výstupech a funkcích.
- VSTUPY – jasná a kompletní definice toho, co budeme evidovat (entity a jejich atributy) a kdo bude mít za to zodpovědnost. Z popisu vstupů musí jednoznačně vyplývat datová analýza (viz kap. 2.2).
- VÝSTUPY – slovně popište dva až tři konkrétní výstupy (např. tiskové sestavy, statistiky, grafy apod.), které bude uživatel potřebovat. Uvažujte, že základem implementace výstupu bude jeden nebo více netriviálních SQL SELECT příkazů. Při popisu buďte konkrétní – které informace ve výstupu uvidíme, podle čeho bude možné výstup filtrovat apod. Uveďte rovněž, pro které typy uživatelů bude výstup dostupný.
- FUNKCE – krátká rozvaha nad tím jaké netriviální funkce bude aplikace mít a jak Vám pomohou tyto funkce řešit problémy popsané v sekci MOTIVACE. Netriviální funkcí je myšlena funkce, která se nedá provést jedním SQL příkazem typu INSERT, UPDATE, nebo DELETE.
- Do specifikace bude zahrnuta i rozvaha nad historií změn klíčových částí řešení (tzn. zajímají nás historie změn určitých entit v databázi?).

Očekává se srozumitelný slohový text na jednu-dvě A4. Základním účelem specifikace je, aby se zadavatel projektu dorozuměl s programátorem.

Ve specifikaci se tedy **nesmí vyskytnout technické pojmy** jako např. tabulka, primární klíč, ID a podobně. Jinými slovy, specifikace musí být formulována tak, aby jí rozuměl i člověk, který není odborníkem v IT.

2.2 Datová analýza

Datová analýza musí obsahovat:

- Grafický tvar konceptuálního modelu (ERD, UML).
- Grafický tvar relačního datového modelu.
- Úplné tabulky atributů (tj. datový slovník).
- Seznam integritních omezení.

2.3 SQL skripty

V rámci tohoto úkolu je potřeba vytvořit komprimovaný archiv se třemi soubory (žádné podadresáře v zipu): skript_schema.sql, skript_data.sql, skript_dotazy.sql Pojd' me si nyní popsat co se v těchto souborech má nacházet:

- První dva soubory jsou SQL skripty pro vytvoření a naplnění databáze.
- (15 bodů) Třetí soubor je SQL skript s validními SELECT dotazy nad Vaší databází. Select dotazy budou v celkovém počtu 22, kde každý bude z jedné šesti kategorií:
 - 4 dotazy se základním výpisem hodnot z tabulek a třídění záznamů
 - alespoň dva dotazy budou obsahovat aritmetickou funkci nebo jinou funkci pro práci hodnotami (viz. přednáška o datových typech). Dva dotazy budou obsahovat ORDER BY, přičemž jeden dotaz bude třídit na základě vypočtené hodnoty, nebo hodnoty získané nějakou funkcí. Jedno třídění by mělo být navíc sestupně.

- 4 dotazy s výběrem řádků - všechny budou obsahovat klauzuli WHERE. Jeden dotaz bude kombinovat operátory AND a OR. Jeden dotaz bude obsahovat negaci. Jeden dotaz bude obsahovat operátor LIKE a regulární výrazy. Jeden dotaz bude provádět manipulaci s atributem (aritmetické operace, nebo funkce)
 - 4 dotazy s množinovými operacemi a poddotazy v podmínce - V této kategorii mají být čtyři dotazy, kde každý z dotazů má být napsán jinak (s použitím jiné konstrukce), ale sémanticky půjde o totožné dotazy. Budou tedy vracet stejný výsledek. Půjde tedy o to si zkusit přepis mezi operátory IN, EXISTS, ALL a množinovými operacemi jako UNION, INTERSECT a EXCEPT.
 - 4 dotazy s agregačními funkcemi a shlukováním - všechny dotazy musí obsahovat nějakou agregační funkci. Tři dotazy musí obsahovat GROUP BY a jeden dotaz musí obsahovat HAVING.
 - 4 dotazy s operací spojení a agregačními funkcemi - první dotaz bude obsahovat běžné vnitřní spojení zapsané s pomocí konstrukce JOIN. Druhý dotaz bude jeho ekvivalent zapsaný s pomocí IN. Tedy druhý dotaz JOIN obsahovat nebude. Oba dotazy musí vracet stejný výsledek. Další dotaz bude obsahovat vnější spojení v kombinaci s GROUP BY a poslední dotaz bude obsahovat vnější spojení v kombinaci s GROUP BY a WHERE. Alespoň u jednoho dotazu s vnějším spojením použijte i agregační funkci.
 - 2 komplikovanější dotazy - budou kombinovat minimálně dva dotazy. Jeden poddotaz bude za FROM nebo SELECT. Dotaz bude koncipován tak, aby se nedal přepsat na jednoduchý dotaz bez poddotazů.
- Před každým SELECT příkazem dejte do blokového komentáře čtveřici: `Kat;Por;Vel;slovní vyjádření`. Kat je kategorie dotazu, Por je pořadí dotazu v kategorii, Vel je očekávaný počet řádků a slovní vyjádření je textový popis, který musí odpovídat sémantice dotazu. Jinak řečeno, musíte být schopni slovně přesně vyjádřit význam Vašeho dotazu. Tzn. komentář pro první SQL dotaz z první kategorie, který bude vracet 3 výsledky bude vypadat takto:


```
/* 1;1;3;Popis SQL dotazu*/
```

- Žádný dotaz nebude obsahovat redundantní části a použité konstrukce musí dávat v dotazu smysl. Tzn. například nebude uznán SQL dotaz s vnějším spojením, když dané vnější spojení je eliminováno nevhodnou podmínkou za WHERE.
- První tři skupiny dotazů jsou bodovány dvěma body a další tři třemi body. Za každé porušení nějakého pravidla je strhnutý v rámci skupiny jeden bod.

2.4 Formální analýza

Formální analýza intuitivního návrhu bude obsahovat:

- Sestavení množiny funkčních závislostí pro jednotlivé relace databáze
- Minimalizace množiny funkčních závislostí algoritmem pro minimální neredundantní pokrytí
- Stanovení typu normální formy pro každou relaci a případná normalizace (pokud je potřeba).

2.5 Závěrečná prezentace projektu

Budou trvat přibližně 3 až 5 minut (3 minuty na přednes studentem a zbytek na otázky). Každou prezentaci si připravte ve formátu PDF s pomocí nástrojů jako je MS PowerPoint, LaTeX/Beamer nebo LibreOffice Impress. Za prezentaci nelze považovat dokumentaci ve formátu A4.

2.6 Finální část

Finální část projektu bude složená z textové části, prezentace a SQL skriptů. Textová část projektu bude mít tyto části:

1. Specifikace zadání
2. Datová analýza

3. Formální analýza (nepovinně)