

# Pokyny pro vypracování projektu

Radim Bača

6. listopadu 2019

Každý projekt musí obsahovat minimálně pět tabulek a pět vazeb. Maximální počet tabulek je deset. Za optimální počet považujeme zhruba sedm tabulek.

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 celé řešení by se 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 a máte o něm hlubší znalosti a zároveň jsou předmětem nějaká 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.

Specifikace a výsledný datový model bude mít následující parametry:

- PROČ - motivace bude mít více než jednu větu.
- KDO - budou alespoň dva typy uživatelů (admin aplikace se nepočítá). Tito uživatelé pak budou zahrnuti v dalších úvahách o vstupech, výstupech a funkcích.

- Do specifikace bude zahrnuta i rozvaha nad historií změn klíčových částí řešení (tzn. zajímají nás historie změn určitých entit v databázi?).

Myslete na to, že v budoucnu (v jiných předmětech jako DAIS) se po Vás bude chtít nad datovým modelem vymyslet netriviální funkce, které nebudou provádět pouze CRUD operace, či nějaké složené SELECT příkazy. Takže tím, že nyní vymyslíte netriviální model si ulehčíte práci v dalších předmětech.

Student může odevzdat pouze zadání, které odsouhlasí jeho cvičící. 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.

# 1 Odevzdání projektu

Odevzdání projektu bude probíhat ve třech krocích + prezentace na posledním cvičení. Každý krok má dva termíny odevzdání. Nebude možné odevzdat v druhém termínu pokud jste v prvním neodevzdali nic! Odevzdání bude probíhat přes stránky [dbedu.cs.vsb.cz](http://dbedu.cs.vsb.cz). Po přihlášení se na stránce objeví menu Profil, kde naleznete položku Odevzdat projekt. Vyberete předmět Úvod do databázových systémů, projekt UDBS a následně jeden z dílčích úkolů:

- Specifikace zadání (termin 23.11. / 30.11. / 7.12.) - odevzdání specifikace zadání projektu (PROČ, KDO, Popis základních funkcí aplikace). Očekává se srozumitelný slohový text na jednu A4.
- Datové modely (termin 30.11. / 7.12.) - odevzdání datových modelu (konceptuální model a relační datový model).
- SQL Skripty (termín 7.12. / 14.12.) - vypracování SQL skriptů, které vytvoří databázi, naplní ji daty a také seznam SQL dotazů. Podrobnější instrukce k tomuto bodu lze nalézt na konci tohoto textu.
- Závěrečná prezentace projektu, které proběhnou na závěrečném cvičení a budou trvat cca. 3 - 5 minut (3 minuty na přednes studentem a zbytek na otázky). Každou prezentaci si připravte v pdf formátu s pomocí nástrojů jako je MS PowerPoint, LaTeX/Beamer nebo LibreOffice Impress. Za prezentaci nelze považovat dokumentaci ve formátu A4.
- Finální verze textu včetně teoretického rozboru datového modelu bude odevzdána v den, kdy budete prezentovat finální verzi.

Všechny body odevzdání projektu **povinné**. Není tedy možné bez omluvy nedojít na prezentaci, nebo očekávat udělení bodů za projekt bez odevzdání některé části. Pokud byly během prezentace nalezeny nedostatky a budete vyzváni k dodělání, pak není možné odevzdávat textovou část bez těchto oprav.

## 2 Textová část

Finální textová část projektu bude mít tyto části:

1. **Specifikace zadání** - Funkční požadavky (KDO, PROČ, ...), Use Case diagram, Tabulka událostí a reakcí, Kontextový diagram
2. **Datová analýza**
  - Lineární zápis seznamu typů entit a jejich atributů.
  - Grafický tvar konceptuálního modelu (ERD, UML).
  - Grafický tvar relačního datového modelu.
  - Úplné tabulky atributů (t.j. datový slovník).
  - Seznam integritních omezení.
3. **Funkční závislosti a normální formy**
  - 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 v případě potřeby.

### 3 SQL skripty

SQL skripty budou mít tyto části:

- SQL skript pro vytvoření a naplnění databáze
- (15 bodů) 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 třízení 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 třízení 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 funkce.

- 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 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.
- 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 strhnut v rámci skupiny jeden bod.