

KIT/BDAS2 - Databázové systémy II

[Titulní stránka](#) / [Moje kurzy](#) / [KIT/BDAS2 \(ZS, 2025/2026\)](#) / [Zápočet, Zkouška](#) / [PROJEKT - ZADÁNÍ](#)

PROJEKT - ZADÁNÍ

Zadání projektu BDAS2

Cílem projektu je prokázat, že si student osvojil prezentované teoretické znalosti a je tyto znalosti schopen aplikovat v praxi. Studenti budou vytvářet **databázovou aplikaci v libovolném vyšším programovacím jazyku - C++, C#, PHP (ideálně objektový přístup), JAVA nebo je možné využít ASP.NET nebo ASP.NET MVC případně jiný framework dle domluvy se cvičícím. ORM je možné využívat pouze jako prostředek pro spouštění vlastních SQL skriptů.** Jejich práce bude zahrnovat interakci se skutečným uživatelem. V rámci projektu musí projít všemi částmi návrhu databázové aplikace - analýzou, modelováním, logickým i fyzickým návrhem. Součástí projektu je tedy i vytvoření funkční aplikace, **nestačí implementace požadavků pouze do databáze.**

Rozsah projektu

Cílem projektu je vytvoření databázové aplikace, ideálně webové aplikace.

Předpokládá se znalost návrhu a tvorby aplikací studentem dle již absolvovaných předmětů. Aplikace bude využívat min. tři různé role – přičemž povinný je administrátor, registrovaný uživatel, uživatel (bez přihlášení), dále to může být zaměstnanec libovolného oddělení, vývojář, školitel, vyučující apod.

Každá dvojice (případně jednotlivec) musí pro získání hodnocení ze zápočtového projektu:

1. provést stručnou prezentaci svého projektu (v rozsahu 10 - 15 minut), prezentaci demonstrujete na své vlastní aplikaci (nevytvářejte žádné powerpointové prezentace) – každý z dvojice by měl prezentovat tu část, kterou sám vytvářel;
2. odevzdat zprávu o projektu– odevzdává se do moodlu a postačí, aby odevzdal jeden z dvojice;
3. odevzdat všechny zdrojové kódy (!okomentované!), skripty a databázové skripty – postačí odevzdává jeden z dvojice;
4. mít implementované použité tabulky v databázovém schématu studenta, toto platí pro každého z dvojice!

Zpráva o projektu

Zpráva o projektu zahrnuje uživatelskou a programovou dokumentaci **viz šablona projektu**, kde naleznete i doprovodný text k jednotlivým položkám.

Uživatelská dokumentace

Obsahuje informace o instalaci aplikace a jejím užití.

Programová dokumentace

Datová část:

Popis zadavatelské dokumentace (zadání projektu), analýza, konceptuální datový model (ERD), logický datový model, datový slovník (fyzický model dat). Zkopírujte strukturální a procedurální pravidla z BDAS1. Zároveň uveďte, jak se liší vaše databáze od BDAS1, např. co jste změnili, jaká jste provedli rozšíření, apod.

Aplikace

Použité prostředí, instalace, popis celé aplikace, popis jednotlivých modulů. Připraveno tak, aby mohl další vývojář pokračovat ve vaší práci.

!!! Jakékoliv nedodržení zadání vede k neuznání projektu !!!

Databázová aplikace (DA) - funkční požadavky k databázové aplikaci BDAS2 (projekt navazuje na projekt z BDAS1)

Výsledná DA z BDAS2 musí splňovat MINIMÁLNĚ následující:

Orientačně je možné říci, že výsledný projekt (databáze + aplikace) bude zahrnovat následující, **vše bude vytvořeno řádně v databázi a zobrazeno a využito v aplikaci!:**

- 1. min. **10 tabulek** navrženého datového modelu, včetně 2 číselníků. V dokumentaci bude uvedeno, proč byly tabulky vybrány jako číselníky;
- 2. Každý umělý **primární klíč** bude mít vytvořenou **sekvenci**;
- 3. min. **3 pohledy** – logicky využité, různého typu; **pohledy je nutné využít pro výpis hodnot v DA;**

- 4. min. **3 funkce** různého typu s odpovídající složitostí, triviální a podobné funkce nebudou uznány, každá z funkcí musí mít odlišný výstup, tedy pracovat s různými operacemi. Do těchto funkcí nebudou počítány funkce převzaté ze cvičení;
- 5. min. **4 uložené procedury** různého typu s odpovídající složitostí, triviální a podobné procedury nebudou uznány, každá z procedur musí mít odlišný výstup, tedy pracovat s různými operacemi, procedura může data zpracovávat i dávkově;
- 6. min. **2 trigger** různého typu opět odpovídající složitostí, triviální a podobné spouště nebudou uznány;
- 7. Bude umožňovat **uložit vybraný binární obsah** do databáze a následně jej i z databáze získat (a pokud se bude jednat o obrázek, tak i v rámci aplikace zobrazit). **Binární obsah bude možné skrz DA vložit, změnit či odstranit.** Pro tento úkol vytvořte ve svém schématu speciální tabulku. Tabulku navrhnete tak, aby kromě samotného binárního obsahu umožnila uložit doplňkové informace, jako např.: název souboru, typ souboru, přípona souboru, datum nahrání, datum modifikace, kdo provedl jakou operaci. Binárním obsahem může být kromě obrázku i jakýkoliv soubor např. PDF či DOCX apod.
- 8. Bude využívat **minimálně 3 plnohodnotné formuláře** (ošetření vstupních polí, apod.) pro vytváření nebo modifikaci dat v tabulkách, ostatní potřebné formuláře jsou samozřejmostí.
- 9. Datová vrstva aplikace bude v rámci vybraných PL/SQL bloků pracovat minimálně s jedním **implicitním kurzorem** a jedním **explicitním kurzorem**.
- 10. DA mohou plnohodnotně využívat pouze **registrovaní uživatelé**, **neregistrovaný uživatel** má velmi omezený výpis obsahu.
- 11. DA umožňuje **vyhledávat a zobrazovat výsledky o všech přístupných datech** v jednotlivých tabulkách dle svého oprávnění. V případě tabulky obsahující **BLOB pak zobrazí název dokumentu/obrázku/jiného binárního souboru** dle zvoleného tématu a návazné informace alespoň ze dvou tabulek.
- 12. DA **umožňuje vkládání či modifikaci dat skrz uložené procedury! Tyto procedury na vkládání/modifikaci nelze započítat do nutných 4 procedur.**
- 13. V DA **nejsou viditelné ID a ani nelze vyhledávat a ani vyplňovat jakékoliv ID**, aplikace je uživatelsky přívětivá. Ani pro roli administrátora není nutné ve výpisu tabulek mít ID jednotlivých záznamů.
- 14. Aplikace **využívá trigger** k logování, zasílání zpráv mezi uživateli, apod. **Trigger k logování nelze započítat mezi 2 povinné trigger.**
- 15. **Grafické rozhraní DA bude funkční** a bude umožňovat editovat jakýkoliv záznam, který je načtený z databáze.
- 16. **DA řádně pracuje s transakcemi** a má ošetřenou práci tak, aby nedošlo k nepořádku v datech.
- 17. DA využívá z datové vrstvy **vlastní hierarchický dotaz**, který je vhodně využit dle tématu semestrální práce.
- 18. DA **eviduje a spravuje údaje o všech uživateli**, kteří mají do aplikace přístup.
- 19. Všechny číselníky se v DA chovají jako číselníky, tzn. že bude využit např. **combobox**, apod. Data z tabulky označená jako číselník nebude uživatel ručně zapisovat.
- 20. **DA má implementované veškerá pravidla, omezení, apod., která byla popsána v projektu z předmětu BDAS1 a nebyla řešena na datové vrstvě.**
- 21. DA je navržena tak, aby **uchovávala historii o vkládání či úpravách jednotlivých záznamů**, toto je zobrazeno pouze uživatelům s rolí Administrátor.
- 22. V DA existuje **funkcionalita, které umožňuje nezobrazovat osobní údaje jiným uživatelům** jako například rodné číslo, telefon, číslo účtu, apod. Toto neplatí pro uživatele v roli Administrátor, ty mají plný přístup všude.
- 23. DA umožňuje **přidávat, modifikovat a mazat záznamy ve všech tabulkách** dle oprávnění uživatele.
- 24. Aplikace bude mít **menu** nastaveno tak, že je možné **z jedné karty přepnout na všechny ostatní**, tak aby byla zaručena příjemná uživatelská správa.
- 25. **Všechny tabulky musí být naplněny řádnými daty**, nikoliv zkušebními.
- 26. Aplikace se skládá z **hlavního okna**, kde má možnost neregistrovaný uživatel procházet povolené položky menu. Hlavní okno aplikace také umožňuje přihlásit registrovaného uživatele.
- 27. **Administrátor může spravovat jakákoliv data a zároveň se může přepnout (emulovat) na jakéhokoliv jiného uživatele.**
- 29. **Uživatel si nemůže sám zvolit při registraci svoji roli, vždy obdrží roli s nejnižšími právy** a poté jej může změnit administrátor. Neregistrovaný uživatel musí vždy mít možnost registrace.
- 30. Databázová aplikace bude umožňovat výpis všech použitých databázových objektů v semestrální práci (využijte **systémový katalog**).

Naposledy změněno: sobota, 20. září 2025, 18.24

◀ Podmínky zápočtu a zkoušky a hodnocení

Přejít na...

PROJEKT - ŠABLONA ▶

✉ [Kontaktujte podporu stránek](#)