

UNIVERZITA PARDUBICE

Fakulta elektrotechniky a informatiky

Dokumentace aplikace

Lebedová Kateřina

Obsah

Úvod	10
Zadání.....	11
Uživatelská dokumentace	12
Základní popis používané aplikace.....	13
Instalace	13
Přístupová oprávnění	13
Použití aplikace.....	13
Programová dokumentace	14
Datová část	15
Analýza	15
Indexy	15
Aplikace	16
Použité prostředí	16
Moduly.....	16
Orientace ve zdrojovém kódu.....	23
Závěr	25

Úvod

Tato práce je přílohou k bakalářské práci na téma In-Memory Database Cache – analýza a zhodnocení. Obsahuje základní seznámení s aplikací pro testování databázových systémů.

Zadání

Mezi cíle bakalářské práce patří i vytvoření aplikace pro testování databázových systémů z pohledu sledování časů potřebných pro zpracování dotazů, které budou také výstupem této práce. Daná aplikace by měla obsahovat obsluhu připojení k danému databázovému systému, tvorbu testovacího prostředí, jeho smazání a provedení příslušného sledovacího testu v jazyce Java, obsluhující SQL dotazy.

Uživatelská dokumentace

Základní popis používané aplikace

Tato aplikace je vytvořena za účelem testování čtyř databázových systémů – Oracle Database 12c, Microsoft SQL Server 2016, MySQL 5.7 a PostgreSQL 9.6. Celé uživatelské prostředí bylo implementováno v jednoduchém designu, aby i méně zasvěcený uživatel pochopil, co která část vykonává.

Instalace

Spuštění aplikace je možné dvěma způsoby. V prvním případě je nutné mít nainstalováno prostředí pro spuštění Java aplikace přímo s přístupem ke zdrojovému kódu, např. prostředí NetBeans. V druhém případě, kdy dochází ke spuštění již spustitelného .jar souboru bez přístupu k aplikačnímu kódu uživateli plně postačí nainstalované Java Runtime Environment.

Přístupová oprávnění

Samotná aplikace neřeší přístupová oprávnění k databázovému systému. Je tedy nutné při připojení přihlašovat uživatele s oprávněním pro vytváření databáze.

Použití aplikace

Aplikace je rozdělena do tří částí. První část obsluhuje výběr požadovaného databázového systému, připojení k němu a následně i ukončení spojení. Druhá část obsluhuje naplnění databázového systému – tvorba testovacího databázového schématu, jeho naplnění testovacími daty a nakonec i jeho celé smazání z databázového systému. Třetí část má za úkol spuštění testu. Po spuštění testu už ho nelze přerušit jiným způsobem, než je ukončení spojení s daným databázovým systémem.

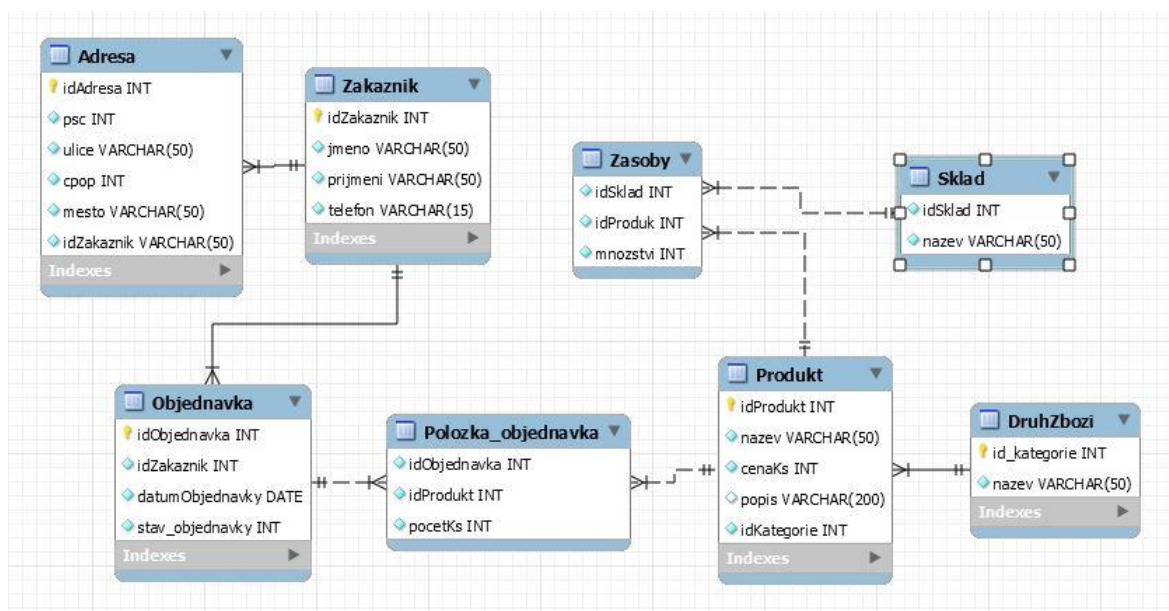
Programová dokumentace

Datová část

Analýza

Pro tvorbu testovacího prostředí v databázovém systému je použito schéma skládající se z osmi tabulek – Adresa, Zákazník, Zásoby, Sklad, Objednávka, PoložkaObjednávka, Produkt a DruhZboží. Dané tabulky obsahují převážně číselné a řetězcové datové typy pro ukládání požadovaných informací. Příložený ukázkový model z obrázku č. 1 byl vytvořen v programu MySQL Workbench 6.3, obsahuje tedy příslušné datové typy pro tento databázový systém. Veškeré vztahy mezi tabulkami jsou také demonstrační.

Tabulky Zákazník, Sklad a DruhZboží jsou samostatné tabulky, které je možno vytvořit bez závislosti na nějaký primární klíč. Adresa, Produkt a Objednávka již obsahují cizí klíče odkazující na primární klíče z tabulky Zákazník, popř. DruhZboží. Poslední dvě tabulky – Zásoby a PoložkaObjednávka slouží jako spojovací mezi příslušnými tabulkami a obsahují cizí klíče odkazující na dvě různé tabulky.



Obrázek 1 - Ukázkové schéma (vlastní zpracování)

Indexy

Celá aplikace nevyužívá indexů, které by však v mnohých případech urychlily vyhledávání v příslušných tabulkách. Použity jsou však v případě vytváření optimalizovaných tabulek pro SQL Server 2016, kdy jsou použity ve formě nonclustered¹ při vytváření primárních klíčů dané tabulky.

¹ Obsahují pouze hodnoty daného sloupce. Opakem jsou clustered indexy, který pokrývá všechny sloupce.

Aplikace

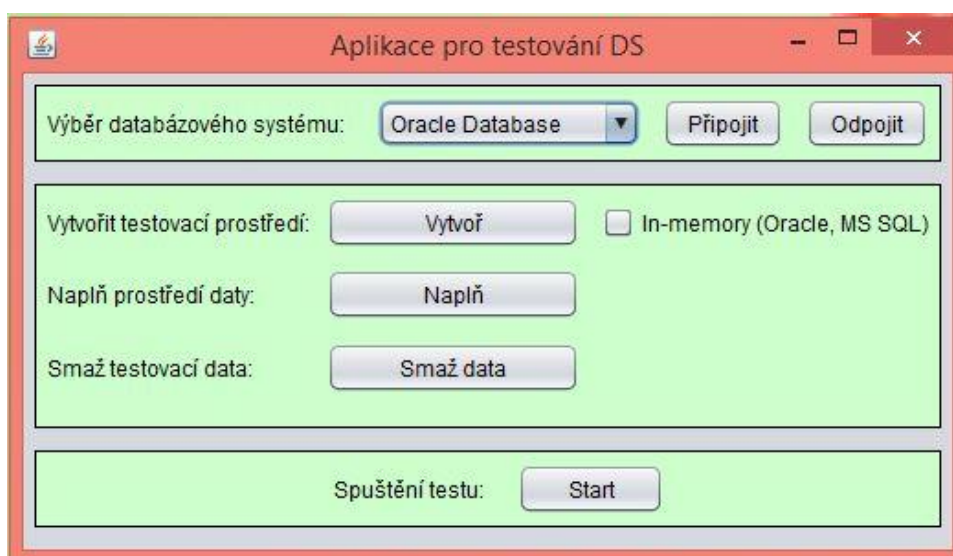
V následující části dokumentace se nachází popis jejích jednotlivých částí, tedy uživatelské části a základní orientace ve zdrojovém kódu.

Použité prostředí

Pro vytvoření aplikace bylo využito prostředí NetBeans IDE 8.2 s podporou JDK 1.8. Pro vytvoření spojení s databázovým systémem prostřednictvím aplikace bylo využito knihoven patřících k danému databázovému systému a JDBC (Java Database Connectivity).

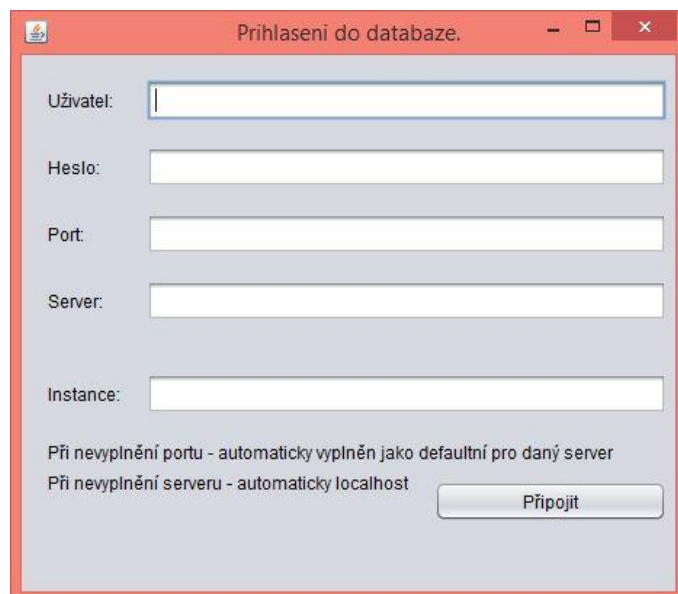
Moduly

První dialogové okno, které se zobrazí po spuštění dané aplikace a je zobrazeno na obrázku č. 2, je hlavní okno obsluhující celou aplikaci. V horní části uživatel nalezne obsluhu připojení k vybranému databázovému systému. Po rozkliknutí nabídky databázových systémů přistoupí k připojení k dané databázi.



Obrázek 2 - Aplikace (vlastní zpracování)

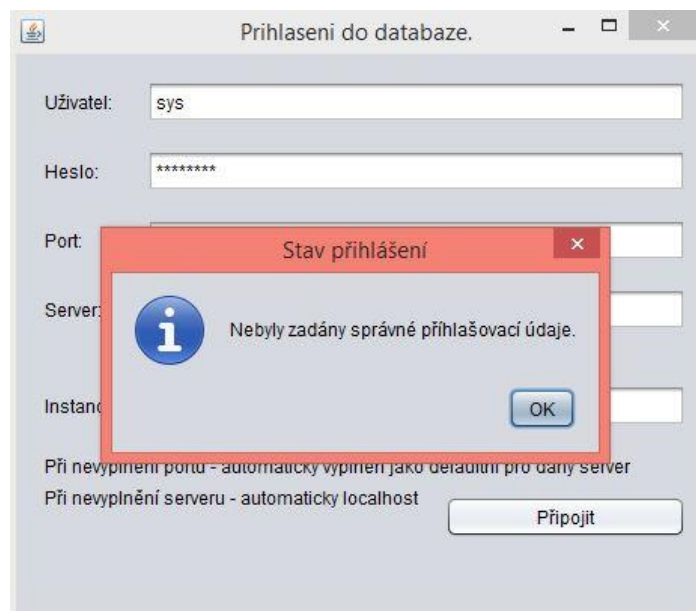
Po volbě daného databázového systému se stisknutím tlačítka Připojit aktivuje dialogové okno z obrázku č. 3, kde uživatel vyplní potřebné údaje k připojení k danému databázovému systému.



The screenshot shows a Windows-style dialog box titled "Přihlasení do databaze." with a red title bar. It contains five text input fields labeled "Uživatel:", "Heslo:", "Port:", "Server:", and "Instance:". Below the fields, there are two lines of text: "Při nevyplnění portu - automaticky vyplněn jako defaultní pro daný server" and "Při nevyplnění serveru - automaticky localhost". At the bottom right is a button labeled "Připojit".

Obrázek 3 - Přihlášení k DS (vlastní zpracování)

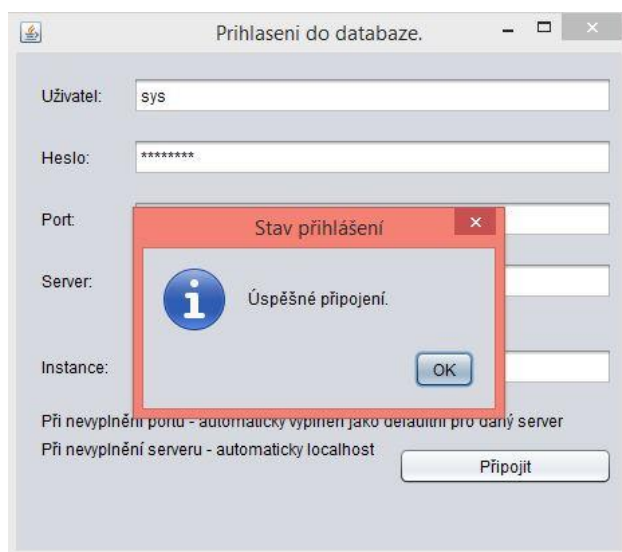
Při nevyplnění správných údajů pro připojení vyskočí dialogové okno oznamující uživateli, že něco není v pořádku (obrázek č. 4).



This screenshot shows the same "Přihlasení do databaze." dialog box, but with the "Uživatel:" field filled with "sys" and the "Heslo:" field filled with "*****". A smaller error dialog box, titled "Stav přihlášení", is overlaid on top. It features a blue information icon and the text "Nebyly zadány správné přihlašovací údaje." with an "OK" button. The "Připojit" button is still visible at the bottom of the main dialog.

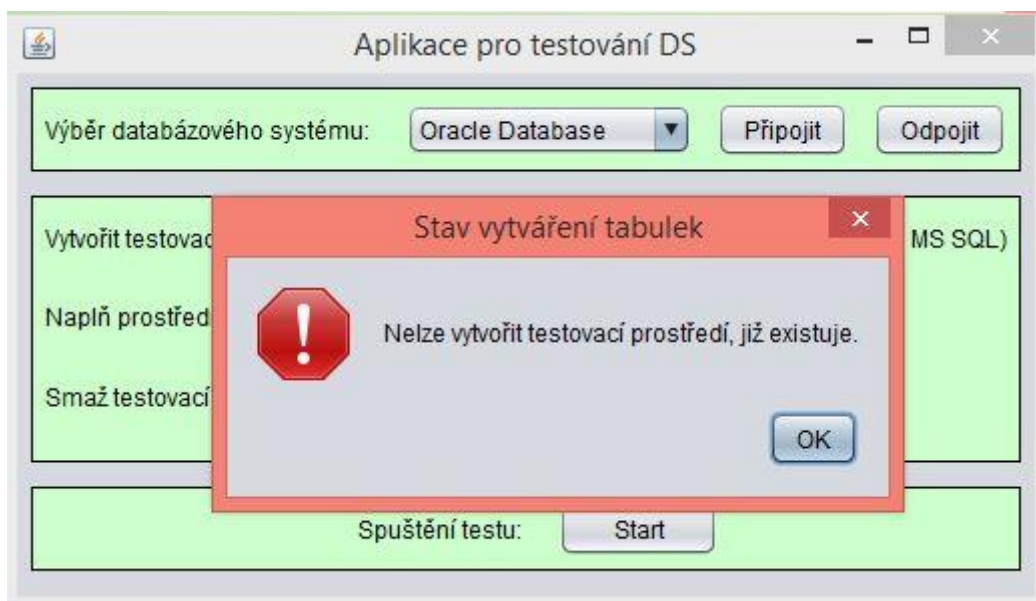
Obrázek 4 - Nesprávné přihlášení (vlastní zpracování)

V případě vyplnění správných přihlašovacích údajů se uživateli zobrazí zpráva o úspěšném připojení (obrázek č. 5) a okno s formulářem pro přihlašovací údaje se samo zavře. Uživateli se opět zobrazí hlavní okno aplikace z obrázku č. 1.

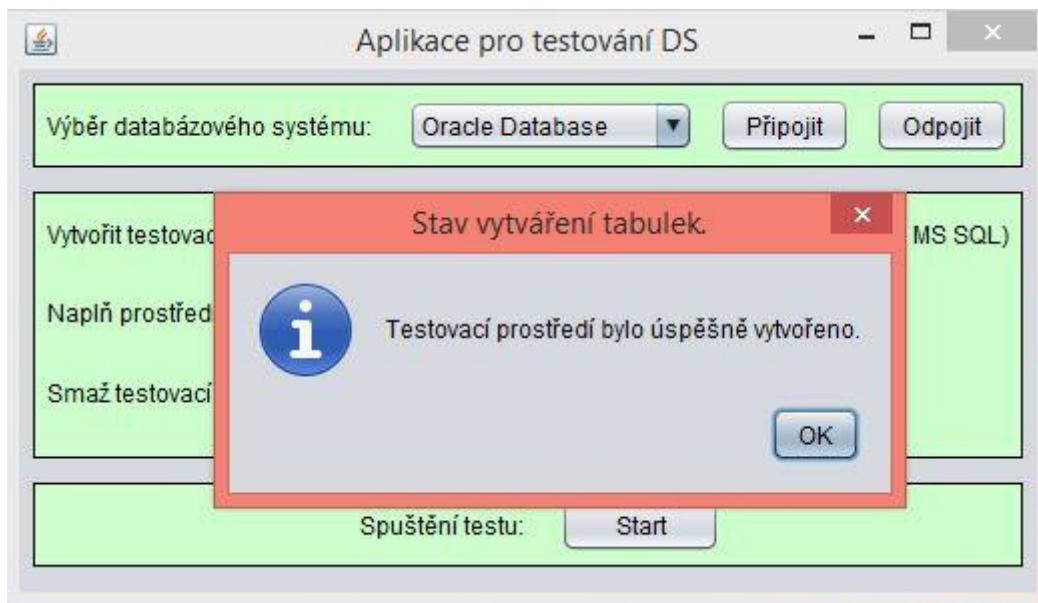


Obrázek 5 - Úspěšné přihlášení

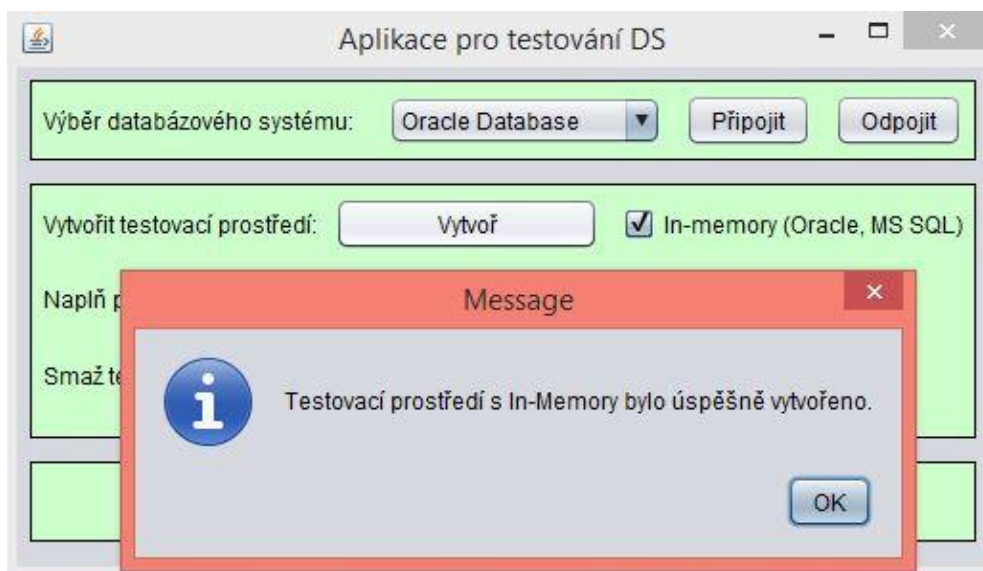
Po úspěšném přihlášení a přepnutí do hlavního okna se již přistupuje ke tvorbě testovacího prostředí tlačítkem Vytvoř. V případě připojení k databázovému systému Oracle Database nebo SQL Server 2016 si uživatel vybírá, zda vytvoří prostředí s využitím In-Memory optimalizovaného prostředí, či bez jeho využití zaškrtnutím/odškrtnutím políčka In-Memory (Oracle, MS SQL). Celá aplikace je ošetřena tak, aby nedocházelo k vytváření testovacího prostředí, které již existuje (duplicitní název tabulek/databáze). Při vzniku tohoto problému je uživateli zobrazeno informační okno upozorňující na tento stav (obrázek č. 6). Při úspěšném vytvoření vyskočí dialogové okno oznamující tento stav (obrázky č. 7 a 8).



Obrázek 6 - Chyba při vytváření existující prostředí (vlastní zpracování)

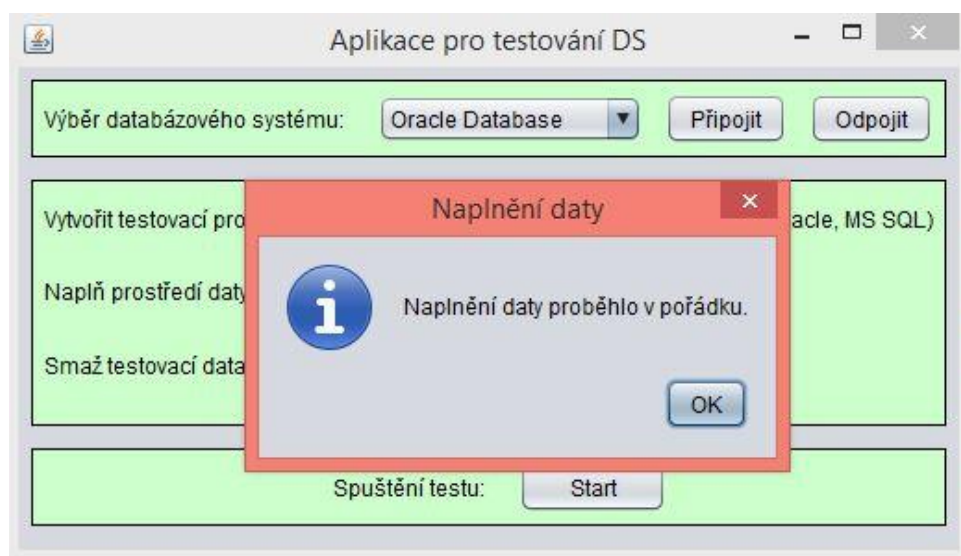


Obrázek 7 - Úspěšné vytvoření testovacího prostředí (vlastní zpracování)



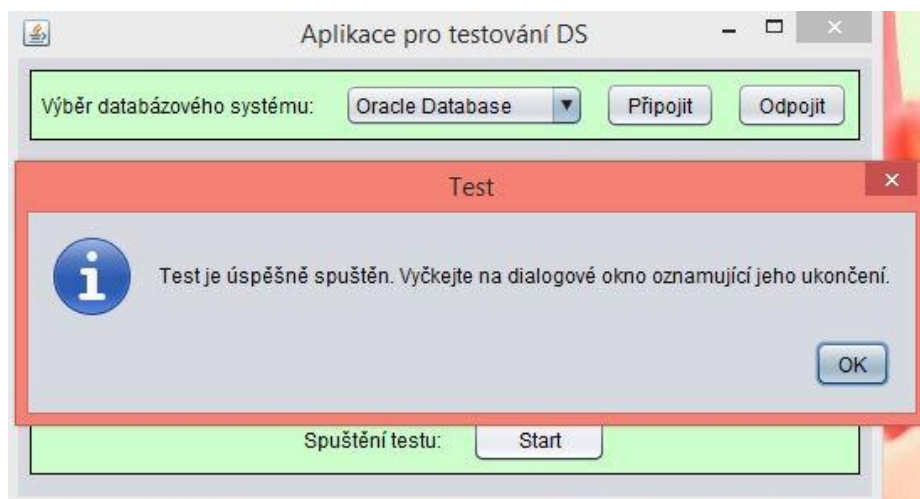
Obrázek 8 - Úspěšné vytvoření testovacího prostředí s In-Memory (vlastní zpracování)

Po vytvoření testovacího prostředí uživatel přestoupí k naplnění testovacího prostředí testovacími daty. Po úspěšném naplnění vyskočí dialogové informační okno (obrázek č. 9). Pro přehled času, který tato činnost zabrala a výsledek se objeví v textovém souboru ve složce, ze které je aplikace spouštěna.



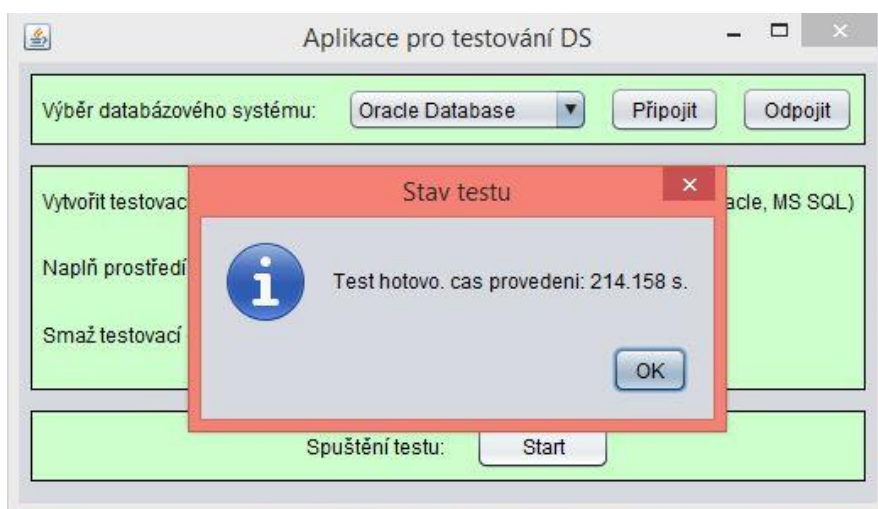
Obrázek 9 - Naplnění daty (vlastní zpracování)

Po vytvoření a naplnění testovacího prostředí už může uživatel přistoupit ke spuštění samotného testu kliknutím na tlačítko Start. Po spuštění testu se uživateli zobrazí informační okno oznamující tuto skutečnost (obrázek č. 10).



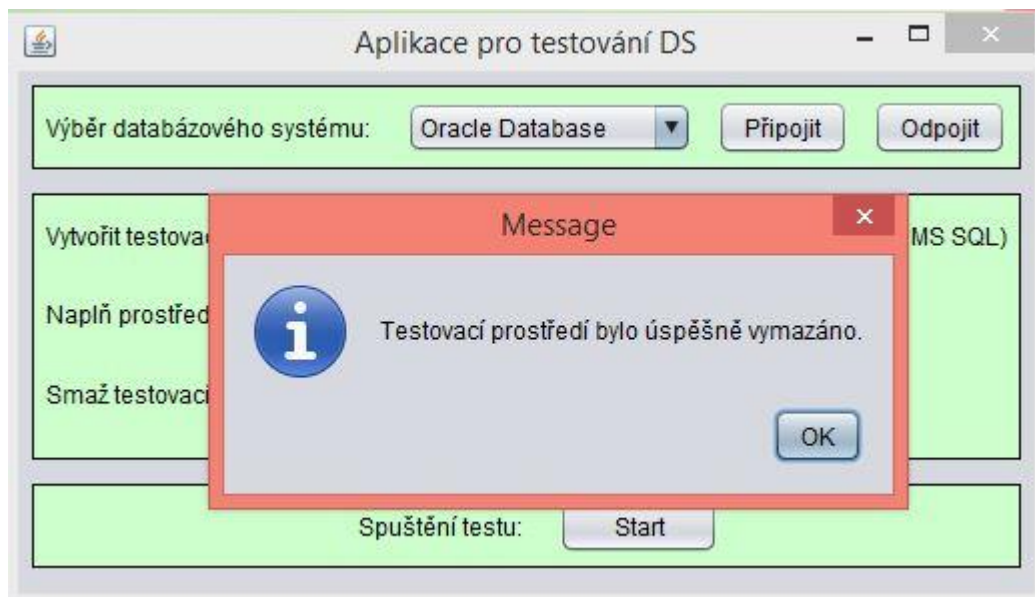
Obrázek 10 - Spuštění testu (vlastní zpracování)

Po skončení testu se zobrazí dialogové okno informující o ukončení testu společně s časem, který byl potřebný k jeho vykonání (obrázek č. 11).



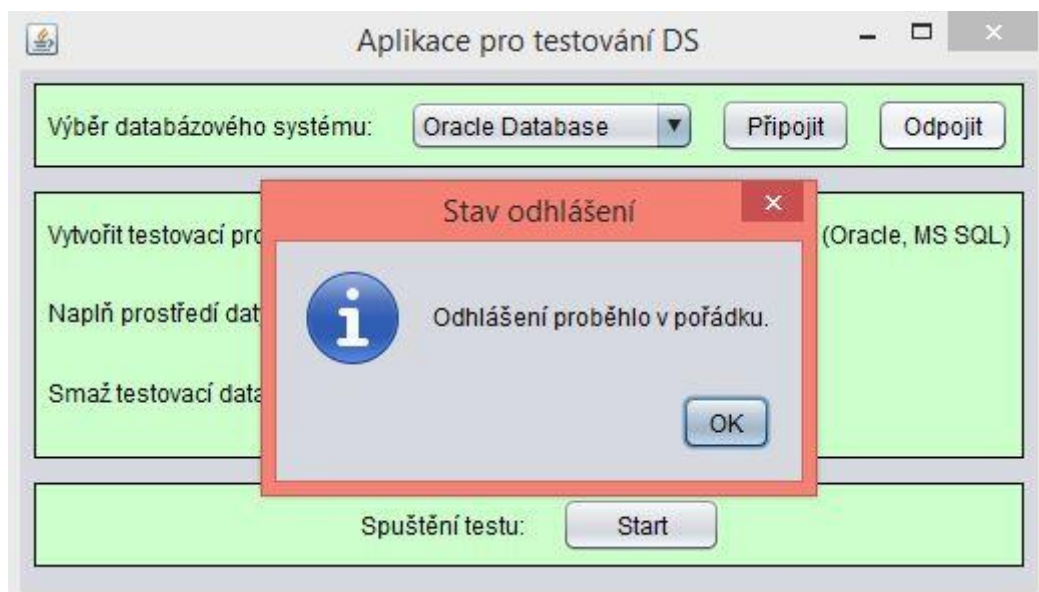
Obrázek 11 - Ukončení testu

Po ukončení testu se uživateli poskytuje nabídka pro smazání testovacího prostředí tlačítkem Smaž data, kterému následuje oznamovací okno o úspěšném provedení této činnosti (obrázek č. 12). Výsledek testu se zobrazí v textovém souboru nacházejícím se ve složce, ze které je daná aplikace spuštěna.



Obrázek 12 - Úspěšné smazání testovacího prostředí (vlastní zpracování)

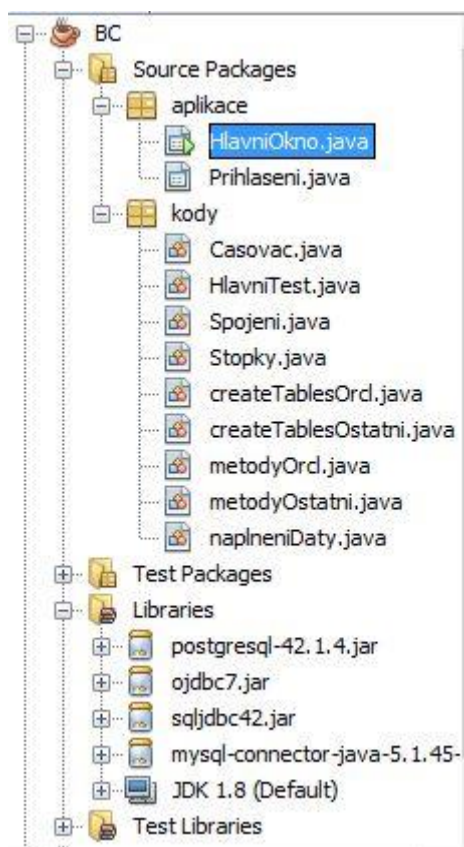
Poslední možností je odpojení od daného databázového systému tlačítkem odpojit. Tato operace je opět provázena dialogovým oknem o úspěšnosti této operace (obrázek č. 13).



Obrázek 13 - Úspěšné odhlášení (vlastní zpracování)

Orientace ve zdrojovém kódu

Celá aplikace je obsažena ve dvou složkách daného projektu. První složka obsahuje 2 balíčky, které oddělují aplikační část (balíček aplikace) od části s pomocnými třídami (balíček kody). Druhá složka – Libraries – obsahuje potřebné knihovny obsluhující připojení k danému databázovému systému. Celé toto rozdělení je vyobrazeno na obrázku č. 13.



Obrázek 13 - Rozdělení projektu (vlastní zpracování)

V balíčku aplikace se nachází dvě třídy – HlavniOkno a Prihlaseni. HlavniOkno slouží ke spuštění celé aplikace a obsahuje příslušné metody pro volání potřebných metod z tříd obsažených v části kódů. K němu je přiložen formulář Prihlaseni, který obstarává zpracování přihlašovacích údajů k danému databázovému systému.

Druhý balíček kody obsahuje třídy pro obsluhu daných operací. Třída Casovac obstarává zpracování naměřených časů, které jsou potřebné pro vykonání dané operace. HlavniTest obsahuje metodu obstarávající přepočet bloků mezi částí samotného testu a jejich spuštění. Třída Spusteni slouží jako pomocná pro předávání existující připojení mezi třídami. Stopky spolupracují s třídou Casovac, které odesílají daný čas zpracování příslušné operace, aby ho mohla přepočítat na potřebné výstupní hodnoty. Třídy CreateTableesOrd a CreateTableesOstatni obsahují příslušné proměnné typu String obsahující již samotné SQL operace nutné k vytvoření daného testovacího prostředí či jeho smazání. Tyto proměnné jsou zpracovány v třídách metodyOrd a metodyOstatni.

Poslední třídou celého projektu je naplneniDaty, která s pomocí přiložených souborů obsahující dotazy s hodnotami pro daný databázový systém vytvoří testovací data v databázi.

Příslušné metody ve zmíněných třídách jsou patřičně okomentovány v kódu, aby bylo jasné, kterou činnost provádějí.

Závěr

Požadovaným cílem bylo vytvoření jednoduché aplikace, která by obsluhovala zpracování časů při zátěžovém testu nad databázovým systémem. Vytvořená aplikace ovládá potřebné prvky nezbytné k zrealizování tohoto cíle. Obsahuje jednoduché uživatelské prostředí, což je velkou výhodou pro budoucího méně zasvěceného uživatele. Chybí ošetření některých chybných uživatelských vstupů, což však nezpůsobilo velké problémy při testování.

V budoucnu by se dále mohla aplikace rozvíjet z pohledu rozšíření nabídky testovaných databázových systémů, popř. vytvoření více spuštěných vláken pro daný test, kdy by se dalo sledovat chování databázového systému při více připojeních. Výsledky by se tedy více přiblížily reálnému zatížení.