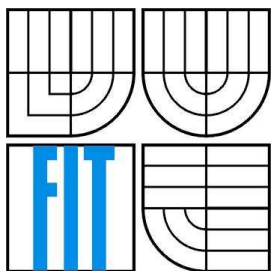




VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
BRNO UNIVERSITY OF TECHNOLOGY



FAKULTA INFORMAČNÍCH TECHNOLOGIÍ
ÚSTAV INTELIGENTNÍCH SYSTÉMŮ

FACULTY OF INFORMATION TECHNOLOGY
DEPARTMENT OF INTELLIGENT SYSTEMS

PODPORA VÝUKY LITEVŠTINY
SUPPORT FOR LEARNING LITHUANIAN

DIPLOMOVÁ PRÁCE
MASTER'S THESIS

AUTOR PRÁCE
AUTHOR

Bc. Martin Kunetka

VEDOUCÍ PRÁCE
SUPERVISOR

Ing. Lukáš Grulich

BRNO 2008

Abstrakt

Tato práce analyzuje požadavky na program pro podporu výuky cizího jazyka. Dále rozebírá jednotlivé aspekty litevštiny a snaží se v návaznosti na poznatky z obou částí navrhnout strukturu a funkcionalitu programu pro podporu výuky litevštiny. Návrh je následně implementován a je zkoumáno vyhovění původním záměrům.

Klíčová slova

Výukový program, cizí jazyk, litevština, učebnice.

Abstract

This thesis analyzes the requirements imposed on applications for teaching foreign languages and proposes a structure and functions of a program fulfilling these requirements in a specific case of Lithuanian language. The proposal is afterwards implemented and evaluated in the respect of meeting the objectives.

Keywords

Learning program, foreign language, Lithuanian, textbook.

Citace

Martin Kunetka: Podpora výuky litevštiny, diplomová práce, Brno, FIT VUT v Brně, 2008.

Podpora výuky litevštiny

Prohlášení

Prohlašuji, že jsem tuto diplomovou práci vypracoval samostatně pod vedením Ing. Lukáše Grulicha.

Uvedl jsem všechny literární prameny a publikace, ze kterých jsem čerpal.

.....
Martin Kunetka
12.5.2008

Poděkování

Chtěl bych poděkovat vedoucímu své diplomové práce za mnoho informací nejen o litevštině.

© Martin Kunetka, 2008.

Tato práce vznikla jako školní dílo na Vysokém učení technickém v Brně, Fakultě informačních technologií. Práce je chráněna autorským zákonem a její užití bez udělení oprávnění autorem je nezákonné, s výjimkou zákonem definovaných případů.

Obsah

| | | |
|-------|---|----|
| 1 | Litevština | 7 |
| 1.1 | Znaky | 7 |
| 1.1.1 | Samohlásky..... | 7 |
| 1.1.2 | Souhlásky..... | 7 |
| 1.1.3 | Speciální znaky..... | 7 |
| 1.2 | Podstatná jména | 8 |
| 1.2.1 | Skloňování | 8 |
| 1.2.2 | Změkčování souhlásek | 8 |
| 1.2.3 | Množné číslo..... | 9 |
| 1.2.4 | Přízvuk..... | 9 |
| 1.2.5 | Výjimky | 11 |
| 1.3 | Přídavná jména..... | 11 |
| 1.3.1 | Skloňování | 11 |
| 1.3.2 | Změkčování souhlásek | 12 |
| 1.3.3 | Množné číslo..... | 12 |
| 1.3.4 | Přízvuk..... | 12 |
| 1.4 | Zájmena..... | 13 |
| 1.5 | Číslovky | 13 |
| 1.5.1 | Výjimky | 14 |
| 1.6 | Slovesa | 14 |
| 1.6.1 | Časování | 14 |
| 1.6.2 | Reflexivní slovesa..... | 15 |
| 1.7 | Ostatní slovní druhy | 15 |
| 2 | Analýza požadavků a možností výukového programu | 16 |
| 2.1 | Slovník | 16 |
| 2.2 | Gramatika..... | 16 |
| 2.3 | Zkoušení..... | 17 |
| 3 | Návrh databáze | 18 |
| 3.1 | Databáze litevštiny..... | 18 |
| 3.1.1 | ER diagram a vysvětlení tabulek | 18 |
| 3.2 | Databáze zkoušení | 20 |
| 3.2.1 | ER diagram a vysvětlení tabulek | 20 |
| 4 | Volba programovacího jazyka | 22 |
| 4.1 | Řešení internetovými stránkami..... | 22 |

| | | |
|-------|--|----|
| 4.1.1 | Výhody | 22 |
| 4.1.2 | Nevýhody..... | 22 |
| 4.1.3 | Programovací jazyk | 23 |
| 4.2 | Řešení aplikací běžící na uživatelském PC..... | 23 |
| 4.2.1 | Výhody | 23 |
| 4.2.2 | Nevýhody..... | 24 |
| 4.2.3 | Programovací jazyk | 24 |
| 4.3 | Porovnání variant | 24 |
| 5 | Jazyk Java..... | 26 |
| 6 | Popis implementace | 28 |
| 6.1 | Logické části programu..... | 28 |
| 6.1.1 | Slovník..... | 28 |
| 6.1.2 | Detaily slov..... | 29 |
| 6.1.3 | Editování slov | 30 |
| 6.1.4 | Přidávání nových slov | 31 |
| 6.1.5 | Zkoušení | 31 |
| 6.1.6 | Uživatelská přívětivost | 34 |
| 7 | Struktura balíčků..... | 35 |
| 7.1 | Spolupráce s databází..... | 35 |
| 7.2 | Panely obrazovek | 35 |
| 7.2.1 | Panely zkoušení | 35 |
| 7.2.2 | Pomocné třídy..... | 35 |
| 7.3 | Importované balíčky | 36 |
| 8 | Možnosti dalšího rozšíření programu | 37 |
| 8.1 | Zkoušení..... | 37 |
| 8.2 | Jazyky | 37 |
| 8.2.1 | Správa zkoušení..... | 38 |
| 9 | Závěr..... | 39 |
| 10 | Literatura | 40 |
| 11 | Přílohy | 41 |

Úvod

Diplomová práce Podpora výuky litevštiny má za cíl vytvořit pomůcku pro vyučující litevského jazyka. Z několika hledisek analyzuje požadavky, které budou na takový program kladeny, a vyhodnocuje, jak by mohly být splněny.

Nedílnou součástí této práce musí být důkladné rozebrání litevštiny z pohledu vybraných gramatických jevů. Některé z nich mohou být implementovány a následně výukovým programem vysvětlovány a zkoušeny.

Diplomová práce se snaží zachytit hlavní rysy litevského jazyka. Větnou stavbu sice nelze zcela vystihnout, ale její dílčí části ano. Podrobně popsáno je zde skloňování podstatných i přídavných jmen a také časování sloves ve všech způsobech i časech.

Nabyté poznatky jsou poté probrány a na jejich základě navrhnuty možnosti, které by mohl program pro podporu výuky litevštiny zahrnovat. Součástí návrhu je rovněž struktura databáze, jež má pojmout nejen slovní zásobu, ale právě také zachytit některé části gramatiky.

1 Litevština

Litevština patří do skupiny baltických jazyků. Je to úřední jazyk Litvy, kde jím mluví přibližně 2,9 milionu lidí. V okolních státech (zejména Bělorusko, Lotyšsko a Polsko) a světě (komunity emigrantů v Argentině, Austrálii, Brazílii, Kanadě, Estonsku, Islandu, Irsku, Rusku, Norsku, Švédsku a Velké Británii) je to pak přibližně 1,1 milionu mluvčích. [6]

Z baltických jazyků se do současnosti dochovaly pouze dva – litevština a lotyšština. V devatenáctém století byla velmi rozšířená tzv. stará pruština, ale ta už vymizela.

1.1 Znaky

1.1.1 Samohlásky

Litevština obsahuje celkem dvanáct samohlásek. Kromě klasických A, E, I, O, U, Y jsou to také Ą, Ę, Ė, Į, Ū a Ū. V malé abecedě vypadají tyto samohlásky takto: ą, ę, ė, į, ū a ū. Samohlásky s obráceným otazníkem se vyslovují jako normální hlásky, akorát dlouze.

1.1.2 Souhlásky

Souhlásek je v litevštině dvacet. Jsou shodné s českými: B, C, Č, D, F, G, H, J, K, L, M, N, P, R, S, Š, T, V, Z a Ž. Kromě toho existuje také spřežka CH. Nejsou zde ovšem některé naše písmena s háčky a také Q, W a X.

Všechny samohlásky s výjimkou J mají palatalizovanou a nepalatalizovanou formu. To znamená, že pokud se za nimi vyskytují samohlásky I nebo E, vyslovují se souhlásky změkčeně.

1.1.3 Speciální znaky

Podle pravopisu se v litevštině používají také znaky jako čárka, obrácená čárka, vlnovka a makron. Určují délku vyslovované hlásky nebo zdůrazňují přízvuk. V drtivé většině případů se ovšem tyto znaky vyskytují pouze ve slovních apod. V běžné psané litevštině se nepíše.

1.2 Podstatná jména

1.2.1 Skloňování

Litevština skloňuje podstatná jména v deseti pádech, ale dva z nich se již nepoužívají. Jeden se vyskytuje už jen v hovorové řeči. Zbývá tedy sedm pádů stejných jako v češtině: nominativ, genitiv, dativ, akuzativ, vokativ, lokál a instrumentál.

Drtivá většina podstatných jmen je buď mužského nebo ženského rodu. Klasický střední rod se zde nevyskytuje, nicméně třetí rod vyjadřuje podstatná jména obojího druhu. Substantiva mohou ještě náležet do čtvrtého nebo pátého rodu, které jsou určitou formou mužského rodu.

Skloňování se provádí podle celkem jedenácti vzorů. Například klasický mužský rod má čtyři vzory: *namas* (dům), *kelias* (cesta), *žodis* (slovo) a *žaltys* (užovka). Podobně je tomu i u ostatních rodů.

Každý pád přidává ke kmeni podstatného jména nějakou koncovku. Týká se to i prvního pádu. Rozpoznání substantiva se tedy provádí právě podle jeho koncovky. Vyskloňované podstatné jméno *žodis* by vypadalo takto: *žodis*, *žodzio*, *žodžiui*, *žodį*, *žodi*, *žodyje*, *žodžiu*. Kmen je tedy „žod“ a k němu v jednotlivých pádech přibýly koncovky *-is*, *-io*, *-iui*, *-į*, *-i*, *-yje*, *-iu*.

Skloňování podstatných jmen v jednotném čísle – přehled koncovek:

| | I | | | | II | | | III | IV | | V |
|---|--------------|---------------|--------------|---------------|-------------|---------------|-------------|---------------|-------------|----------------|--------------|
| | <i>namas</i> | <i>kelias</i> | <i>žodis</i> | <i>žaltys</i> | <i>mama</i> | <i>vyšnia</i> | <i>katė</i> | <i>širdis</i> | <i>alus</i> | <i>vaisius</i> | <i>akmuo</i> |
| 1 | -as | -ias | -is | -ys | -a | -ia | -ė | -is | -us | -ius | -uo |
| 2 | -o | -io | -io* | -io* | -os | -ios | -ės | -ies | -aus | -iaus | -ens |
| 3 | -ui | -iui | -iui* | -iui* | -ai | -iai | -ei | -iai* | -ui | -iui | -eniui |
| 4 | -ą | -ią | -į | -į | -ą | -ią | -ę | -į | -ų | -ių | -enį |
| 5 | -e | -y | -i | -y | -a | -ia | -e | -ie | -au | -iau | -enie |
| 6 | -e | -yje | -yje | -yje | -oje | -ioje | -ėje | -yje | -uje | -iuje | -enyje |
| 7 | -u | -iu | -iu* | -iu* | -a | -ia | -e | -imi | -umi | -iau | -enimi |

1.2.2 Změkčování souhlásek

Některé koncovky (resp. pády) změkčují poslední souhlásku kmenu podstatného jména. Už v předcházejícím příkladu bylo zřejmé, že z písmena „d“ se někdy stalo „dž“. Takovýchto výjimek existuje několik a nejsou nijak systematicky rozpoznatelné. Ve všech pádech každého vzoru musí tedy být uvedeno, zda zde dochází ke změkčení.

V praxi se setkáme s dvěma změnami souhlásek. Písmeno „d“ se měkčí na „dž“ a písmeno „t“ na „č“. Ostatní souhlásky nejsou změkčením ovlivněny, ani kdyby k tomu koncovka vybízela.

Koncovky, které mají změkčování na svědomí, jsou v předcházející tabulce vyznačeny hvězdičkou.

1.2.3 Množné číslo

Všechny rody umožňují použití množného čísla. Plurál se řídí prakticky stejnými pravidly jako jednotné číslo. Ke kmeni, který má totožný tvar jako kmen v singuláru, se přidávají koncovky. Ty jsou ovšem jiné, než v jednotném čísle. I zde může docházet ke změkčení a to dokonce i v jiných pádech než v singuláru. Opět nelze říci, kde se změna souhlásek vyskytuje (obecné pravidlo neexistuje).

Skloňování podstatných jmen v množném čísle – přehled koncovek:

| | I | | | | II | | | III | IV | | V |
|---|-------|--------|---------|---------|-------|--------|-------|--------|-------|---------|---------|
| | namas | kelias | žodis | žaltys | mama | vyšnia | katė | širdis | alus | vaisius | akmuo |
| 1 | -ai | -iai | -iai* | -iai* | -os | -ios | -ės | -ys | -ūs | -ius | -enys |
| 2 | -ų | -ių | -ių* | -ių* | -ų | -ių | -ių* | -ių* | -ų | -iaus | -enu |
| 3 | -ams | -iams | -iams* | -iams* | -oms | -ioms | -ėms | -ims | -ums | -iui | -enims |
| 4 | -us | -ius | -ius* | -ius* | -as | -ias | -es | -is | -us | -ių | -enis |
| 5 | -ai | -iai | -iai* | -iai* | -os | -ios | -ės | -ys | -ūs | -iau | -enys |
| 6 | -uose | -iuose | -iuose* | -iuose* | -ose | -iose | -ėse | -yse | -uose | -iuje | -enyse |
| 7 | -ais | -iais | -iais* | -iais* | -omis | -iomis | -ėmis | -imis | -umis | -iau | -enimis |

1.2.4 Přízvuk

Přízvuk se může v litevštině vyskytovat na různých slabikách. Jsou dva způsoby, kam přízvuk umístit – buď na kmen nebo na koncovku. Rodilí mluvčí toto určují plně automaticky, ale pro cizince se nejedná o snadnou záležitost. Stejně podstatné jméno může mít přízvuk v jednom pádě na kmeni, ale v jiném na koncovce. Dal by se však najít poměrně složitější systém, kterým se pohyb zdůrazněné slabiky řídí.

Zjednodušeně řečeno, přízvuk je na začátku na kmeni, ale kmen i koncovka o něj spolu svedou boj, přičemž jeho konečné umístění závisí na vlastnostech obou částí slova.

Kmen má dvě vlastnosti (Nabízí přízvuk?, Nechá si jej vzít?) a jejich kombinací vznikají čtyři typy podstatných jmen:

- 1: Kmen přízvuk nenabízí a ani se jej nenechá vzít od koncovky.
- 2: Kmen přízvuk nenabízí, ale nechá si jej vzít.
- 3: Kmen přízvuk nabízí, ale nenechá si jej vzít.

4: Kmen přízvuk nabízí a nechá si ho vzít.

Podobně je tomu i u koncovek (Odmítá nabídnutý přízvuk?, Žádá o něj?). Kombinace opět tvoří čtyři různé typy koncovek:

00: Koncovka neodmítá nabídnutý přízvuk, ale sama o něj nežadá.

0b: Koncovka neodmítá nabídnutý přízvuk a sama o něj žádá.

x0: Koncovka odmítá nabízený přízvuk a sama o něj nežadá.

xb: Koncovka odmítá nabízený přízvuk, ale sama o něj žádá.

(znak „x“ znamená odmítnutí přízvuku, znak „b“ vyjadřuje žádost o přízvuk)

Celkový algoritmus by se dal shrnout do tří kroků:

1. Přízvuk je na kmeni.
2. Pokud je podstatné jméno typu 3 nebo 4, nabídne přízvuk koncovce. Ta si jej vezme, pokud je typu 00 nebo 0b.
3. Pokud je koncovka typu 0b nebo xb, tak žádá přízvuk po kmeni. Pokud je ten typu 2 nebo 4, předá přízvuk koncovce.

Skloňování podstatných jmen v jednotném čísle – přehled přízvuků koncovek:

| | I | II | III | IV | V |
|---|----|----|-----|----|----|
| 1 | x0 | 00 | 00 | 0b | 00 |
| 2 | x0 | 00 | 00 | 00 | 00 |
| 3 | x0 | x0 | x0 | x0 | x0 |
| 4 | x0 | x0 | x0 | x0 | x0 |
| 5 | x0 | 00 | 00 | x0 | x0 |
| 6 | 0b | 00 | 00 | 00 | 00 |
| 7 | xb | 00 | 00 | xb | xb |

Skloňování podstatných jmen v množném čísle – přehled přízvuků koncovek:

| | I | II | III | IV | V |
|---|----|----|-----|----|----|
| 1 | 00 | 00 | 00 | 00 | 00 |
| 2 | 00 | 00 | 00 | 00 | 00 |
| 3 | 00 | 00 | 00 | 00 | 00 |
| 4 | xb | xb | xb | xb | xb |
| 5 | 00 | x0 | x0 | x0 | x0 |
| 6 | 00 | 00 | 00 | 00 | 00 |
| 7 | 00 | 00 | 00 | 00 | 00 |

1.2.5 Výjimky

V litevštině existuje několik výjimek, které se neřídí žádným vzorem. Skloňují se nepravidelně a tudíž je nutné znát všech sedm (resp. čtrnáct i s množným číslem) tvarů dotyčného podstatného jména.

Příklad: sesuo (sestra) se skloňuje: sesuo, seser s, se seriai, se serį, seserie, seseryje, seserimi (singulár), se serys, seserų, seseri, se seris, se serys, seseryse, seserimi s. V tabulce vzorů bychom tyto koncovky hledali marně.

Další výjimku tvoří podstatná jména, která se sice skloňují podle pravidel některého vzoru, nicméně gramaticky náleží do jiného rodu než použitý vzor.

1.3 Přídavná jména

1.3.1 Skloňování

Přídavná jména mohou být stejně jako podstatná jména skloňována v sedmi pádech jednotného i množného čísla. Děje se tak podle pěti základních vzorů, přičemž záleží na tom, zda se k adjektivu váže podstatné jméno mužského nebo ženského rodu. Každý vzor je tedy vyskloňován ve dvou rodech.

Skloňování se opět provádí pomocí koncovek, které se připojují ke kmeni přídavného jména. Pět vzorů (geras = dobrý (I), žalias = zelený (II), gerėnsis = lepší (III), medinis = dřevěný (IV), saldus = sladký (V)) se používá podle následujících pravidel: vzory I a II jsou určeny pro klasická přídavná jména. Vzor III předepisuje skloňování adjektiv v komparativu. Podle vzor IV se skloňují přídavná jména vyjadřující původ materiálu. Vzor V obsluhuje speciální adjektiva končící na -us.

Příklad vyskloňovaného přídavného jména geras v mužském rodě: geras, gero, geram, gerą, geras, gerame, geru. V drtivé většině případů bývá pátý pád shodný s prvním pádem. U množného čísla toto platí bez výjimky.

Skloňování přídavných jmen v jednotném čísle – přehled koncovek:

| | I – geras | | II – žalias | | III – gerėnsis | | IV – medinis | | V – saldus | |
|---|-----------|------|-------------|-------|----------------|------|--------------|------|------------|--------|
| | M | F | M | F | M | F | M | F | M | F |
| 1 | -as | -a | -ias | -ia | -is | -ė | -is | -ė | -us | -i |
| 2 | -o | -os | -io | -ios | -io | -ės | -io | -ės | -aus | -ios* |
| 3 | -am | -ai | -iam | -iai | -iam | -ei | -iam | -ei | -iam* | -iai* |
| 4 | -ą | -ą | -iaą | -iaą | -į | -ę | -į | -ę | -ų | -ia* |
| 5 | -as | -a | -ias | -ia | -is | -e | -i | -e | -us | -i |
| 6 | -ame | -oje | -iame | -ioje | -iame | -ėje | -iu | -ėje | -iame* | -ioje* |
| 7 | -u | -a | -iu | -ia | -iu | -e | -iame | -e | -iu* | -ia* |

1.3.2 Změkčování souhlásek

I toto skloňování může mít na svědomí změkčení některých souhlásek. Koncovky, které změkčují, jsou opět vyznačené hvězdičkou. Pokud kmen přídavného jména končí na „t“, změní se poslední znak na „č“. Stejně je tomu v případě „d“ měnícího se na „dž“.

1.3.3 Množné číslo

Také plurál se řídí stejným systémem jako u podstatných jmen. Skloňování tedy znovu probíhá pomocí koncovek, které jsou jiné než v jednotném čísle. I tady dochází v některých pádech pátého vzoru ke změkčování.

Skloňování přídavných jmen v množném čísle – přehled koncovek:

| | I – geras | | II – žalias | | III – gerėnsis | | IV – medinis | | V – saldus | |
|---|-----------|-------|-------------|--------|----------------|--------|--------------|--------|------------|---------|
| | M | F | M | F | M | F | M | F | M | F |
| 1 | -i | -os | -i | -ios | -i | -ė̄s | -iai | -ė̄s | -ūs | -ios* |
| 2 | -ų | -ų | -ių | -ių | -ių | -ių | -ių | -ių | -ių* | -ių* |
| 3 | -iems | -oms | -iems | -ioms | -iems | -ė̄ms | -iams | -ė̄ms | -iems | -ioms* |
| 4 | -us | -as | -ius | -ias | -ius | -es | -ius | -es | -ius* | -ias* |
| 5 | -i | -os | -i | -ios | -i | -ė̄s | -iai | -ė̄s | -ūs | -ios* |
| 6 | -uose | -ose | -iuose | -iose | -iuose | -ė̄se | -iuose | -ė̄se | -iuose* | -iose* |
| 7 | -ais | -omis | -iais | -iomis | -iais | -ė̄mis | -iais | -ė̄mis | -iais* | -iomis* |

1.3.4 Přízvuk

Přízvuk znovu cestuje mezi kmenem přídavného jména a koncovkou. Algoritmus na zjištění jeho konečného umístění je totožný s tím, který byl uveden v kapitole o podstatných jménech. Co se však liší, je typ koncovek v jednotlivých pádech.

Skloňování přídavných jmen v jednotném čísle – přehled přízvuků koncovek:

| | I | II | III | IV | V |
|---|----|----|-----|----|----|
| 1 | x0 | x0 | 00 | 00 | 00 |
| 2 | x0 | x0 | x0 | x0 | 00 |
| 3 | 00 | 00 | 00 | 00 | 00 |
| 4 | x0 | x0 | x0 | x0 | x0 |
| 5 | x0 | x0 | 00 | 00 | 00 |
| 6 | 00 | 00 | 00 | 00 | 00 |
| 7 | xb | xb | xb | xb | xb |

Skloňování přídavných jmen v množném čísle – přehled přízvuků koncovek:

| | I | II | III | IV | V |
|---|----|----|-----|----|----|
| 1 | 0b | 0b | 0b | 00 | x0 |
| 2 | 00 | 00 | 00 | 00 | 00 |
| 3 | 00 | 00 | 00 | 00 | 00 |
| 4 | xb | xb | xb | xb | xb |
| 5 | 0b | 0b | 0b | 00 | x0 |
| 6 | 00 | 00 | 00 | 00 | 00 |
| 7 | 00 | 00 | 00 | 00 | 00 |

Význam značek je stejný jako u koncovek podstatných jmen. Raději jej zde ovšem uvedu znovu:

00: Koncovka neodmítá nabídnutý přízvuk, ale sama o něj nežádá.

0b: Koncovka neodmítá nabídnutý přízvuk a sama o něj žádá.

x0: Koncovka odmítá nabízený přízvuk a sama o něj nežádá.

xb: Koncovka odmítá nabízený přízvuk, ale sama o něj žádá.

(znak „x“ znamená odmítnutí přízvuku, znak „b“ vyjadřuje žádost o přízvuk)

1.4 Zájmena

Zájmena nemají při skloňování žádné předepsané vzory. U každého slova tohoto druhu je tudíž nutné uvést všech sedm tvarů. Ve skutečnosti však stačí pouze šest tvarů, jelikož pátý pád (vokativ) zájmena netvoří.

Příklad skloňování zájmena aš (já): aš, manęs, man, mane, -, manyje, manimi.

1.5 Číslovky

Stejně jako zájmena, ani číslovky se neskloňují podle žádných vzorů. U všech je tedy potřebné uvést všech sedm (resp. šest) tvarů. U malých číslovek se ovšem jinak skloňuje v mužském a jinak v ženském rodě.

Příklad: vienas (jeden) se v mužském rodě skloňuje: vienas, vieno, vienám, viena, -, vienamè, víenu. V ženském je to ovšem: vienà, vienos, víenai, viena, -, vienojè, víena.

1.5.1 Výjimky

Výjimku tvoří číslovka dešimt (deset). Narozdíl od ostatních číslovek se totiž neskloňuje a zůstává pořád ve stejném tvaru. Pojí se s druhým pádem (např. dešimt namuṣ – deset domů).

1.6 Slovesa

V litevštině existují čtyři časy: minulý (préteritum), přítomný (présens), budoucí (futurum) a minulý opakovaný (frekventativ). První tři jsou stejné jako v českém jazyce, poslední jmenovaný slouží k vyjádření opakované činnosti (např. imdavau = brával jsem).

Kromě oznamovacího způsobu (indikativ) se v litevštině vyskytuje také podmiňovací způsob (kondicionál) a rozkazovací způsob (imperativ).

Všechna slovesa tvoří všechny čtyři časy i všechny tři způsoby.

1.6.1 Časování

V přítomném čase se časuje podle čtyř vzorů (něšti – nést, kėlti – zvednout, mylėti – milovat, bijoti – bát), přičemž první z nich jsou si velmi podobné. Liší se pouze vloženou samohláskou „i“.

Časování v přítomném čase - koncovky

| | nėšti | kėlti | mylėti | bijoti |
|----------|-------|-------|--------|--------|
| 1. j.č. | -u | -iu | -iu | -au |
| 2. j.č. | -i | -i | -i | -ai |
| 3. j.č. | -a | -ia | -i | -o |
| 1. mn.č. | -ame | -iame | -ime | -ome |
| 2. mn.č. | -ate | -iate | -ite | -ote |
| 3. mn.č. | -a | -ia | -i | -o |

V minulém čase se slovesa časují dvěma způsoby se stejnými vzory. Nėšti a kėlti tvoří jednu skupinu, mylėti a bijoti druhou. Jestliže se v přítomném čase sloveso řídí například vzorem něšti, neznamená to, že se stejným vzorem řídí i v minulém čase. Může se klidně stát, že v minulém čase spadá do druhé skupiny a bude se časovat například podle vzoru bijoti. Pro jednoznačné určení časování je tedy nutné, aby bylo sloveso uvedeno ve třech tvarech – infinitivu, třetí osobě přítomného času a třetí osobě minulého času. Rozlišení na jednotné a množné číslo se v litevštině u třetí osoby nevyskytuje.

Časování v minulém čase - koncovky

| | něšti | kešti | mylěti | bijoti |
|----------|-------|-------|--------|--------|
| 1. j.č. | -iau | -iau | -au | -au |
| 2. j.č. | -ei | -ei | -ai | -ai |
| 3. j.č. | -è | -è | -o | -o |
| 1. mn.č. | -ème | -ème | -ome | -ome |
| 2. mn.č. | -ète | -ète | -oite | -oite |
| 3. mn.č. | -è | -è | -o | -o |

Časování v budoucím čase se provádí se stejnými koncovkami pro všechny čtyři vzory. Obdobně tomu je i u minulého opakovaného času a také u podmiňovacího a rozkazovacího způsobu. Narozdíl od češtiny existuje v litevštině imperativ ve třetí osobě. Jediný případ, kdy nelze rozkazovací způsob vytvořit, je první osoba jednotného čísla.

Časování v ostatních časech a způsobech - koncovky

| | frekventativ | futurum | kondicionál | imperativ |
|----------|--------------|---------|-------------|----------------|
| 1. j.č. | -dav-au | -s-iu | -č-iau | |
| 2. j.č. | -dav-ai | -s-i | -t-um(ei) | -k |
| 3. j.č. | -dav-o | -s | -t-ų | tegu + présens |
| 1. mn.č. | -dav-ome | -s-ime | -t-u(mè)me | -k-ime |
| 2. mn.č. | -dav-ote | -s-ite | -t-u(mè)te | -k-ite |
| 3. mn.č. | -dav-o | -s | -t-ų | tegu + présens |

1.6.2 Reflexivní slovesa

V češtině se vyskytuje gramatický jev sloveso se zvrtným (reflexivním) zájmenem – například „nést se“. V litevštině se k vyjádření tohoto tvaru nepoužívá zájmeno, ale speciální koncovky.

Reflexivní slovesa se opět skloňují podle čtyř výše vyjmenovaných vzorů. Systém je ve všech osobách, časech i způsobech stejný, akorát se ke kmenům přidávají jiné (zpravidla delší) koncovky a v některých případech i přípony.

1.7 Ostatní slovní druhy

Príslovce, předložky, spojky, částice a citoslovce se stejně jako v češtině neskloňují. Za všech okolností tedy mají stejný tvar.

2 Analýza požadavků a možností výukového programu

Tématem této diplomové je program pro podporu výuky litevštiny. Program si tedy nedává za cíl naučit uživatele cizí jazyk, protože slouží pouze jako pomůcka pro učitele. Ti během výuky postupují podle předem daných osnov.

U málo používaných jazyků je ovšem velkým problémem sehnat jakoukoliv učebnici. Kompletní učebnice i s pracovním sešitem není prakticky nikdy k dispozici. Pokud si vyučující přeje dát svým žákům nějaká cvičení jako domácí úkoly, musí si vše připravit sám.

Jako řešení se nabízí víceúčelový program, který by sloužil nejen jako prostředek k domácím úkolům, ale také jako slovník nebo pomocník s gramatikou.

2.1 Slovník

K dispozici je volně šiřitelný česko-litevský slovník obsahující poměrně rozsáhlou slovní zásobu. Toho můžeme využít a tento slovník do programu zakomponovat. Uživatelé v něm budou moci vyhledávat překlady jak z českého jazyka do litevského, tak naopak.

2.2 Gramatika

Popsat gramatiku jazyka není možné, a tak by bylo celkem zbytečné pokoušet se implementovat překladač celých vět. Některé důležité gramatické jevy ovšem lze zachytit a výukový program by s nimi měl být obeznámen.

V kapitole o litevštině byl popsán způsob, jakým se skloňují podstatná jména. Program by mohl být seznámen se všemi vzory a pravidly pro skloňování. Při zobrazení detailu určitého slova by tedy mohl uvést jeho tvary ve všech pádech.

To se samozřejmě netýká pouze podstatných jmen. Na stejném principu by mohlo fungovat i skloňování přídavných jmen, zájmen a číslovek.

Rovněž časování sloves má přesně daná pravidla, a tak je možné podle určitých zákonitostí vypsát všechny tvary.

Dalším gramatickým jevem je například stěhování přízvuku mezi kmenem podstatného jména a koncovkou slova. Jak už bylo uvedeno, i toto se dá přesně určit. Program tedy může být seznámen s různými částmi gramatiky, které budou probírány v jednotlivých lekcích.

2.3 Zkoušení

Program může být využíván jako pomocník – vyhledá-li uživatel nějaké slovo, zobrazí se mu všechny správné tvary. Kromě toho však může plnit také úlohu domácího zkoušení. Ke každé lekci v učebnici / osnovách bude vytvořeno několik cvičení různého typu.

Jako první se samozřejmě nabízí zkoušení slovíček. Učitel může určit, která slovíčka náleží do které lekce, a program z nich potom bude uživatele (žáky) zkoušet. Studenti budou psát jak české ekvivalenty k litevským slovům, tak litevské k českým.

V úvodních lekcích může být rovněž cvičení, které bude požadovat vypsání všech sedmi tvarů podstatného jména (přídavného jména, zájmena,...). Vzhledem k tomu, že rozpoznání, podle kterého vzoru se substantivum skloňuje, se provádí podle koncovky, není nutné zkoušet toto dlouhodobě.

Jiná situace je však u sloves. Jejich časování se sice také řídí určitými vzory, ale rozpoznání vzorů není tak jednoduché jako u podstatných jmen. Ke správnému časování je nutné znát tři tvary slovesa. Proto by měl program zkoušet také znalost zbývajících dvou tvarů (ne jen infinitivu).

Mnohem více gramatických jevů je možno najít v celých větách nebo souvětích. Toto už nelze naimplementovat tak, aby program „rozuměl“ učivu, nicméně to ve zkoušení nebrání. Mohou být připravena cvičení (např. na správné použití časů nebo správné spojování do souvětí), kde bude mít žák na výběr ze tří předem připravených vět, z nichž jen jedna bude správná.

Tímto způsobem se dá zkoušet gramatika postupně ve všech lekcích. Spoustu dalších pravidel – jako třeba umístění přízvuku ve vyskloňovaném podstatném jménu – lze rovněž zachytit.

Program samozřejmě nemůže být náhražkou pracovního sešitu žáků základních či středních škol. Učitel nemá možnost kontrolovat na hodině, zda a jak studenti úkol splnili. Zkoušení na počítači může žák opakovat tak dlouho, až bude mít všechny odpovědi správně. Výukový program spíše bude sloužit jako doplněk k hodinám s učitelem. Pomůže zopakovat probranou látku a vhodným způsobem také studenty přezkouší. Jeho využití se tedy spíše posouvá k jazykovým školám či k vyučování na vysoké škole. Lektoři nemají v hodinách moc času, aby kontrolovali domácí úkoly. Mnohdy nemají ani prostor udělat dostatek cvičení – stává se, že předvedou dvě nebo tři věty z celého cvičení a zbytek nechají na studentech.

Ti jsou naštěstí ve většině případů přítomni dobrovolně, protože se chtějí cizí jazyk z vlastní vůle naučit. Výukový program se tedy nebudou snažit obelstít „klikáním“ odpovědí tak dlouho, až budou správné, ale využijí jej tak, jak mají. Zejména tedy pro kontrolu, že probíranému učivu skutečně porozuměli.

3 Návrh databáze

Databáze je rozdělena do dvou hlavních částí. Jedna má za úkol „umět“ litevsky a druhá obstarává zkoušení.

Nejprve tedy musí vzniknout část, která v sobě uchovává kompletní slovní zásobu a která se také pokouší postihnout vybrané gramatické jevy litevštiny. Až na tomto základě vznikne část databáze, jež bude provádět zkoušení z jednotlivých lekcí.

3.1 Databáze litevštiny

Nejprve bych nastínil hlavní myšlenku databáze. Centrálním bodem je tabulka Slovo, kde budou uchována litevská slova a jejich český ekvivalent. K většině slov potřebujeme znát spoustu dalších vlastností, ale ty se mezi jednotlivými slovními druhy natolik liší, že ukládat je do hlavní tabulky nemá smysl.

Proto zavedeme pouze jeden další atribut – slovní druh. Na tomto základě se pak v databázi nasměrujeme dál. Například u podstatných nebo přídavných jmen si zjistíme, podle jakého vzoru se slovo skloňuje. Podle vlastností kmene i koncovky poté můžeme určit umístění přízvuku.

Podobně tomu je i u sloves. Z tabulky sloves si zjistíme kmen a k němu budeme při časování přidávat koncovky podle vzorů, dle kterých se dané sloveso časuje.

Nyní podrobněji vysvětlím jednotlivé tabulky a informace, které v nich budou uloženy.

3.1.1 ER diagram a vysvětlení tabulek

ER diagram této části databáze je z důvodu nadměrné velikosti uveden v příloze (1).

3.1.1.1 Podstatná

V této tabulce má každé podstatné jméno uchováno svůj kmen a jeho vlastnosti. Skloňování totiž probíhá přidáváním koncovek ke kmeni a proto je nutné kmen separovat. Důležité vlastnosti kmene jsou dvě: zda přízvuk nabízí koncovce a zda si jej nechá od koncovky vzít. Tyto dva atributy jsou uloženy jako typ boolean. Vzor, podle kterého se podstatné jméno skloňuje, je cizím klíčem do tabulky „PodstatnaVzory“. Poslední boolovský atribut určuje, zda je dané substantivum výjimkou či nikoliv. Funkci primárního klíče obstarává ID totožné s ID v tabulce „Slova“.

3.1.1.2 Přídavná

Tato tabulka má prakticky stejnou strukturu jako tabulka podstatných jmen. U přídavných jmen však ukládáme dva vzory – jeden pro skloňování s mužským rodem a jeden pro ženský rod. Oba tyto atributy jsou cizími klíči do tabulky „PridavnaVzory“.

3.1.1.3 Podstatná – vzory

V této tabulce jsou uchovány všechny vzory, podle kterých se podstatná jména skloňují. Základní znění vzorového slova je ve sloupci Vzor. Pro všechny pády pak musí být uložena koncovka a její vlastnosti. Boolovské atributy uvádějící, jestli koncovka přijímá přízvuk, jestli o něj žádá a jestli změkčuje poslední písmeno kmene. To samé se týká i množného čísla.

3.1.1.4 Přídavná – vzory

Tabulka funguje na stejném principu jako její ekvivalent k podstatným jménům. Vzory pro mužské a ženské skloňování jsou zde uloženy zvlášť. Kmen je sice stejný, ale oba rody mají různé koncovky.

3.1.1.5 Podstatná – výjimky

Vymyká-li skloňování podstatného jména pravidlům a neřídí-li se žádnými vzory, je třeba uložit všechny tvary, kterých může substantivum nabývat. První pád jednotného čísla se nachází v tabulce „Slovo“, zbylé pády jednotného čísla a všechny pády množného čísla zde.

3.1.1.6 Přídavná – výjimky

Tato tabulka je úplně stejná jako tabulka výjimek podstatných jmen. Musí zde však být uloženy tvary pro skloňování jak v mužském, tak v ženském rodě. Celkový počet sloupců je tedy oproti předchozímu případu dvojnásobný.

3.1.1.7 Zájmena

Skloňování zájmen nemá žádné regule, tudíž je ve všech případech nutné uložit všechny tvary. První pád se opět nachází v tabulce „Slovo“, zbylé v této.

3.1.1.8 Číslovky

Také číslovky nejsou skloňovány podle vzorů a i tady se nevyhneme uložení všech tvarů.

3.1.1.9 Slovesa

Časování sloves probíhá v minulém a přítomném čase podle různých vzorů. Odkazy na vzory jsou tedy brány jako cizí klíče do tabulky „SlovesaVzory“. Atribut Kmen uchovává samotný kmen slovesa, ke kterému se koncovky následně připojují. Posledním sloupcem tabulky je boolovský atribut Výjimka plnící stejnou funkci jako u podstatných a přídavných jmen.

3.1.1.10 Slovesa – vzory

Podle vzorů se časuje pouze v přítomném a minulém čase. Jednotlivé vzory tedy mají v tabulce uvedeny koncovky pro všechny osoby jen v těchto časech. Třetí osobu stačí uložit pouze jednou, jelikož se koncovky v singuláru a plurálu neliší.

3.1.1.11 Slovesa – pravidelná

Časování v budoucím a opakovaném minulém čase je stejné pro všechna slovesa. Stejně tak i rozkazovací a podmiňovací způsob má totožná pravidla. Do této tabulky tedy uložíme koncovky první až třetí osoby obou čísel. Jednotlivé řádky jsou tvořeny různými časy případně způsoby.

3.1.1.12 Slovesa – výjimky

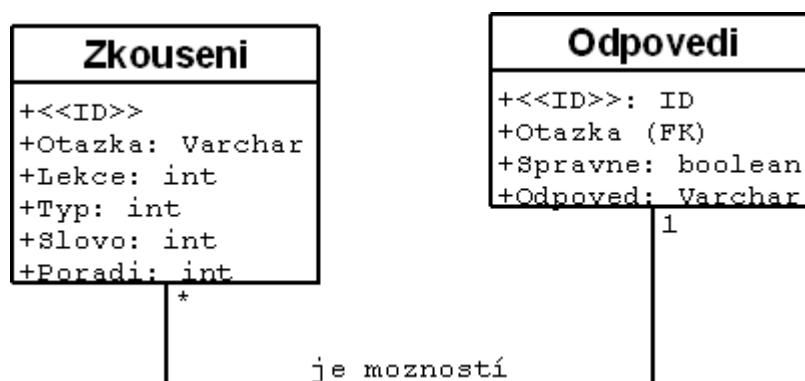
Pokud se stane, že časování slovesa nelze provést podle běžných pravidel, musíme uložit všechny jeho tvary do zvláštní tabulky. Je nutné uvést všechny osoby obou čísel všech časů i způsobů.

3.2 Databáze zkoušení

Druhá část databáze uchovává informace o tom, co a jak je ve které lekci zkoušeno. V této fázi analýzy existuje zkoušení dvojího „typu“. Jedno se zaměřuje na slovíčka ze všech hledisek – jedná se tedy jak o překlad z jednoho či druhého jazyka, tak zkoušení různých tvarů slova vzniklých skloňováním nebo časováním.

Gramatické jevy, které nelze aplikační logikou postihnout, mohou být též zkoušeny. Například správný slovosled může být otestován třemi možnostmi, z nichž musí uživatel vybrat správnou. Obecně se však tímto způsobem dá ptát na cokoliv.

3.2.1 ER diagram a vysvětlení tabulek



Obrázek 1 – ER diagram části databáze věnované zkoušení

3.2.1.1 Slovíčka

V této tabulce jsou shromažďována cvičení zkoušená v jednotlivých lekcích. Příslušnost k lekci je dána atributem typu integer. Atribut Slovo je cizím klíčem do tabulky „Slova“ z předcházející části databáze. Udává, které slovo bude zkoušeno.

Typ (integer) pak určuje, o jaký typ zkoušení se jedná. Například 1 udává překlad z litevštiny do češtiny, číslo 2 překlad z češtiny do litevštiny, číslo 3 vyskoňování podstatného jména, číslo 4 vyskoňování přídavného jména v mužském rodě, atd.

3.2.1.2 Odpovědi

Tabulka odpovědi eviduje jednotlivé odpovědi na otázky z přecházející tabulky. Kromě samotného textu odpovědi jsou zde další dva atributy. Cizí klíč Otázka vyjadřuje, ke které otázce odpověď náleží, a boolovský atribut Správně určuje, zda je daná odpověď správná či nikoliv.

4 Volba programovacího jazyka

Základní cíle programu pro podporu výuky litevštiny byly stanoveny v první části práce. Volba konkrétního programovacího jazyka, ve kterém bude projekt realizován, však vyžaduje důkladnější analýzu možností funkcí, které by mohl program poskytovat.

Od začátku bylo jasné, že v úvahu připadají dva základní typy provedení. Mohlo se jednat buď o internetové stránky, nebo o aplikaci běžící na počítači uživatele. Obě možnosti mají nesporné výhody i nevýhody, a tak jsem musel nejprve pečlivě vyhodnotit, kterým funkcím dát přednost a které naopak nejsou pro splnění cíle programu tolik důležité.

4.1 Řešení internetovými stránkami

4.1.1 Výhody

Narozdíl od aplikace distribuované na CD případně jiným způsobem má internetové řešení velkou výhodu v podobě snadnější administrace a zejména aktualizace. V databázi slovíček a podrobných informací o nich se mohou snadno vyskytnout chyby. Ty by se daleko lépe opravovaly v případě, že by data byla uložena pouze jednou na serveru a nikoliv u všech uživatelů zvlášť. Postupně by navíc mohla přibývat další slovíčka, popřípadě nové doplňující údaje vztahující se k nim.

Hlavní součástí programu je zkoušení, které studentům dá představu, jak moc probírané učivo pochopili. Zpětnou vazbu by ovšem ocenil i učitel a pokud by jeho žáci vyplňovali testy přes Internet, mohla by aplikace automaticky ukládat počty správných či špatných odpovědí. Samozřejmě by nemusela evidovat odpovědi individuálně ke každému uživateli – to by byla spíše slabá stránka programu. Studenti by byli raději, kdyby odpovídali anonymně a lektor by přitom nepřišel o hodnotné údaje.

I v oblasti zkoušení by se mohly vytvářet nové typy cvičení, které by se mnohem snadněji zaváděly do provozu. Pokud by se učitel nebránil pravidelné aktualizaci testů, mohl by cvičení vytvářet na základě průběhu vyučovacích hodin. Kdyby se stalo, že nějaká původně plánovaná cvičení nestihne v hodině se studenty udělat, mohl by je později přidat do databáze zkoušení. Stejně tak by mohl posílit testy zaměřené na určitou oblast, která například studentům v hodině nešla.

4.1.2 Nevýhody

Podstatným problémem řešení internetovými stránkami je, jak už z principu vyplývá, nutné připojení k Internetu. Je pravda, že v dnešní době to představuje potíže pro jen velmi malou skupinu mladších lidí. Jenže jazykové školy nejsou uzavřeny ani starším zájemcům o studium a koneckonců i mezi mladými se najdou tací, pro něž by nutnost připojení k síti byla problémem. Jedná se však také o

pohodlí – aplikaci běžící na vlastním počítači mohou studenti spustit kdekoliv, třeba ve vlaku na notebooku.

Společně s tímto řešením by nastala také povinnost zavést identifikaci a autentizaci uživatelů. K internetové aplikaci by samozřejmě neměli mít přístup všichni, a tak by se studenti museli přihlašovat. Znamenalo by to také rozdělení oprávněných uživatelů minimálně do dvou rolí – právo vytvářet a editovat cvičení by měl pochopitelně jen učitel.

Přihlašování by ale mohlo mít negativní dopad na používání samotného programu. Ačkoliv by mohl lektor výslovně zdůraznit, že sbírané statistiky odpovědí jsou anonymní, všichni by mu rozhodně nemuseli věřit. Proto by raději cvičení nedělali, než aby riskovali, že se učitel dozví, co všechno neumí. Takovýto přístup se samozřejmě týká jen části potenciálních studentů, nicméně riziko je natolik velké, že je třeba se tímto psychologickým faktorem zabývat. Druhou stranou mince by zase mohla být obava žáků, že kantor uvidí čas i datum jejich přihlášení. Proto by se raději do systému přihlašovali, aby působili dojmem, že se výuce věnují.

Ať už by chování studenta bylo ovlivněno tak či tak, pokaždé lze říct, že to je spíše stinná stránka tohoto typu řešení. Program má sloužit jako pomůcka ke zkvalitnění výuky a nemá vytvářet na studenty tlak k používání nebo naopak strach software využívat.

4.1.3 Programovací jazyk

Pokud bych zvolil řešení internetovou aplikací, nabízela by se celá řada vhodných jazyků v čele s PHP, ASP a dalšími. S největší pravděpodobností bych vybral první zmíněný, jelikož díky stejné „síle“ různých variant by byl nejlepší volbou jazyk, se kterým mám nejvíce zkušeností.

4.2 Řešení aplikací běžící na uživatelském PC

4.2.1 Výhody

Značná část výhod i nevýhod tohoto řešení kopíruje své protějšky z předcházející části. Je vcelku logické, že velkým kladem takto provedené aplikace by byla nezávislost na připojení k Internetu. Problémy spojené s touto oblastí byly vyjmenovány a určitě nejsou zanedbatelné. Nainstalovat aplikaci na osobní počítač dnes již zvládne prakticky každý, a tak převažuje výhoda možnosti pracovat offline. Další výhodou reprezentuje také již zmíněná anonymita při zkoušení, která by sice byla u konkurenčního řešení garantovaná, ale to ještě neznamená, že by se tak uživatelé také chovali.

Odpadla by rovněž starost se správou přihlašovacích údajů. Studenti se do podobných kurzů většinou přihlašují na půl roku nebo na rok a každý z nich by musel obdržet své přihlašovací jméno a heslo. Ačkoliv by se nejednalo o nijak důsledně chráněnou internetovou aplikaci, jazykové školy by se přesto určitě snažily zabránit studentům minulého období v přístupu k programu. To by v praxi

znamenal nutnost spravovat uživatele – každému novému žákovi přidělit údaje a naopak po každém školním roce blokovat přístup studentům, kteří školu již opustili.

4.2.2 Nevýhody

Vyskytne-li se v databázi litevských slov nějaká chyba či nepřesnost, bylo by poměrně nesnadné tento nedostatek napravit. Jakmile si studenti aplikaci i s databází nainstalují, už se k opravám jen tak nedostanou. Škola by sice mohla rozdávat (popřípadě nabízet ke stáhnutí) různé „záplaty“, jenže procento žáků, kteří by tuto službu využili by asi bylo nevelké.

Toto řešení projevuje jen malou flexibilitu v oblasti zkoušení. Škola sice může rozdávat každým rokem jiná instalační CD s opravenými databázemi, nicméně někteří učitelé by určitě dali přednost mnoha výhodám, které byly zmíněny v části 4.1.1.

Další potíže se mohou vyskytnout i v samotné aplikaci. S obrovským množstvím slovíček je daleko pravděpodobnější výskyt chyby v databázi, ale také v programu se může časem projevit nesprávné fungování. Stejně jako v případě DB, i toto lze rozumně řešit pouze jednou za určitý logický interval (semestr, školní rok).

Nevýhodou také je, že tento způsob vyžaduje vlastnictví počítače. Veřejné knihovny a podobné instituce nedovolí instalovat vlastní software, a tak je osobní PC nutností. Vzhledem k rozšíření informačních technologií v dnešním světě, kdy svůj počítač má prakticky každý (a když ne vlastní, tak aspoň rodinný), jsem se rozhodl tuto nevýhodu zanedbat.

4.2.3 Programovací jazyk

Programovacích jazyků vhodných pro vytvoření aplikace běžící na běžném počítači je opravdu široká škála. Volba by se zúžila na takové prostředky, které umožňují snadné vytváření grafických uživatelských rozhraní.

Hojně rozšířeným a na platformách značně nezávislým jazykem je Java. Pokud bych tedy z obou možností celkového provedení programu zvolil tuto, jako implementační jazyk bych si vybral právě Javu.

4.3 Porovnání variant

Když ve stručnosti shrnu výhody a nevýhody obou variant, tak co do flexibility programu i jednoduchosti provedení aktualizací potřebných dat je jednoznačným vítězem řešení pomocí internetové aplikace. K tomuto se ovšem váže otázka, zda by tato výhoda opodstatnila volbu zmíněné varianty. Aktualizace databáze slovíček (nejen kvůli přidávání nových slovíček, ale také k opravě starých chyb) například jednou za semestr se mi jeví jako přijatelná. Pokud by se takto rozdávaly nové instalační disky CD (případně by byla nová instalace dostupná na Internetu), nebyl by problém rovnou

přidat také novou verzi samotného programu. Tím by byl zajištěn i vývoj aplikace. Vezmeme-li v úvahu, kolik uživatelů by reálně mohlo program používat, jsou novinky uvolňované jednou za půl roku, nebo dokonce i jednou za rok, stále ucházející.

Naopak řešení pomocí aplikace instalované na uživatelský počítač slibuje značný komfort. Program lze využívat nezávisle na připojení k Internetu a navíc odpadají všechny strasti v souvislosti s autentizovaným přihlášením. A tyto problémy by se dotkly jak samotných studentů, tak škol, které by výuku litevštiny zajišťovaly.

Z výše vyjmenovaných důvodů jsem se rozhodnul realizovat program pro podporu výuky litevštiny jako aplikací instalovanou a běžící na uživatelském počítači. Výhody internetového řešení podle mého názoru nepřevýšily klady druhé možnosti.

Samotnou implementaci jsem pak provedl ve v současné době populárním jazyce Java.

5 Jazyk Java

Historie jazyka Java spadá do 90. let minulého století. Návrh a vývoj obstarává společnost Sun Microsystems. Hlavním požadavkem při návrhu byla nezávislost na cílové architektuře. Té vývojáři dosáhli kompilováním zdrojových kódů nikoliv do strojového kódu procesoru, ale do bajtového mezikódu. Až ten je následně překládán pomocí tzv. virtuálních strojů pro účely konečné platformy.

Podle prohlášení tvůrců je Java „jednoduchý, objektově orientovaný, distribuovaný, interpretovaný, robustní, bezpečný, nezávislý na architektuře, přenositelný, vysoce výkonný, víceprocesní a dynamický jazyk“ [4].

Jednoduchost – spočívá ve snadném přechodu programátorů na tento jazyk. Java obsahuje minimální množství jazykových konstrukcí, které jsou navíc odvozeny z již existujících jazyků. Odstraněny jsou také nejčastější problémové oblasti – příkazy goto, přetížení operátorů, vícenásobná dědičnost nebo ukazatelová aritmetika.

Java byla od počátku vyvíjena jako **objektově orientovaný** jazyk. S výjimkou osmi primitivních datových typů jsou všechny ostatní objektové.

Distribuovanost – Java je navržena pro podporu aplikací na síti. Podporuje různé úrovně síťového připojení, snadno vytváří komunikaci pomocí socketů, umí otevírat vzdálené objekty, atd.

Interpretace – jak už bylo zmíněno, zdrojové kódy se nejprve překládají do bajtového mezikódu. Až poté jsou v závislosti na cílové platformě překládány konkrétním virtuálním strojem.

Robustnost – Java je určena pro psaní široké škály softwaru. Programy se vyznačují velkou spolehlivostí právě díky tomu, že některé „zatemňující“ konstrukce nejsou přípustné. O správu paměti se navíc stará tzv. garbage collector, který automaticky odstraňuje nepoužité objekty a slučuje volné části paměti. Velké plus představuje rovněž strukturované ošetřování výjimek.

Bezpečnost – Java vychází z předpokladu, že se nedá nikomu a ničemu věřit. Díky tomu, že nepodporuje práci s ukazateli, zabrání přesměrování ukazatelů do jiné části paměti, kde se mohou nacházet chráněná data. Také načítaný kód ze sítě je kontrolován „runtime systémem“, jenž zajistí, že nedojde k porušení jazykových omezení Javy.

Nezávislost na architektuře – díky finálnímu zpracování konkrétních virtuálních strojů jsou aplikace psané v Javě použitelné bez větších rozdílů na všech rozšířených operačních systémech.

Přenositelnost – „nezávislost na architektuře není to samé jako přenositelnost, ale pouze její velkou částí. Přenositelnost musí navíc zamezit existenci jakýchkoliv implementačně závislých aspektů jazyka.“ Návrh se opírá o normu POSIX (Portable Operating System Interface).

Výkonnost – i když je Java interpretovaný jazyk, je její rychlost dostačující pro vykonávání běžných programů. Další zrychlení přináší interpret typu just-in-time překládající dynamicky na základě potřeby části kódu přímo do strojového kódu procesoru.

Podpora vláken – Java pochopitelně, stejně jako všechny moderní jazyky, podporuje vlákna. Součástí je taktéž vyloučení souběžného přístupu.

Dynamičnost – třídy lze do programu zavádět podle potřeby a dokonce je možné tyto třídy importovat ze sítě. Java umožňuje zjistit, instancí jaké třídy je daný objekt a s touto informací dále pracovat.

Popis jednotlivých rysů Javy použil pan Lubomír Brůha ve své knize Java, hotová řešení [4]. Pasáže v uvozovkách citují autora, zbylé jsou pak interpretací jeho výkladu.

6 Popis implementace

Aplikaci tvoří základní grafické rozhraní odvozené od třídy JFrame z knihovny Swing. Do tohoto hlavního rámce je přidána nástrojová lišta a také panel, do něž se vykresluje. Panelů odvozených od třídy JPanel je více – pro každou obrazovku programu jeden. V závislosti na tom, kterou obrazovku je zrovna nutné vykreslit, se do hlavního rámce přidávají jednotlivé panely.

Každý panel poskytuje metody důležité pro běh programu. Jedná se zejména o funkce mající na starost správné vykreslení všech prvků a údajů v nich. U obrazovek používaných ke zkoušení pak panely obsahují také metody zjišťující kontrolu vyplněných textových (případně jiných) polí.

Panely jsou ovšem zcela odfiltrovány od spolupráce s databází. Pokud potřebují načíst data, volají metody tříd starajících se o komunikaci s databází. Panely získaná data pouze vykreslí na příslušné místo.

Spojení s databází zajišťují třídy jednoho balíčku. Každá tabulka databáze je reprezentovaná jednou třídou. Samotné třídy pak poskytují metody na výběr dat dle různých kritérií, vkládání nových záznamů a upravování nebo mazání stávajících záznamů.

6.1 Logické části programu

6.1.1 Slovník

Slovník je základem každého programu zabývajícího se cizími jazyky. Přístup k němu tedy musí být rychlý a jeho ovládání intuitivní. Na obrazovku slovníku se vždy dá dostat jedinou klávesovou zkratkou.



Obrázek 2 – obrazovka slovníku

Uživatel má možnost zvolit ze dvou přepínačů podle toho, zda chce vyhledávat v české či litevské části slovníku. Do tabulky pod formulářem se vypíše odpovídající slova v obou jazycích. Jako první se do výpisu zařadí slova, která jsou zcela shodná s údajem, jenž uživatel vyplnil do textového pole. Poté se doplní nalezená slova začínající na zadaný řetězec seřazená podle abecedy.

6.1.2 Detaily slov

Vzhledem k tomu, že litevština skloňuje podstatná jména, přídavná jména, zájmena i číslovky a také časuje slovesa, vytváří se velké množství tvarů každého ohebného slova. Stejně jako v češtině se skloňování i časování děje podle určitých vzorů. To sice může být snadné pro rodilého mluvčího, ale ne už pro někoho, kdo se jazyk učí. Vzor lze ve většině případů poznat pouhým pohledem na koncovku prvního pádu, ale někdy není identifikace tak jednoduchá. Uživatelé každopádně uvítají možnost vypsání všech tvarů slova – buď ve všech pádech a rodech, nebo ve všech časech.

dviratis - kolo (jízdní)

Slovní druh: Podstatné jméno Editovat slovo

Vlastnosti slova

| Vlastnost slova | Hodnota |
|-----------------|--|
| Výjimka | slovo se řídí klasickými pravidly |
| Kmen | dvirat |
| Vzor | žodis |
| Rod | mužský |
| Nabízí přízvuk | ne |
| Předá přízvuk | ne |
| Plurál | slovo tvoří jak jednotné, tak množné číslo |

Tvary slova

| Pád / čas | Tvar slova | Přízvuk |
|--------------|-------------|---------|
| 1. pád j.č. | dviratis | kmen |
| 2. pád j.č. | dviračio | kmen |
| 3. pád j.č. | dviračiui | kmen |
| 4. pád j.č. | dviratį | kmen |
| 5. pád j.č. | dvirati | kmen |
| 6. pád j.č. | dviratyje | kmen |
| 7. pád j.č. | dviračiu | kmen |
| 1. pád mn.č. | dviračiai | kmen |
| 2. pád mn.č. | dviračių | kmen |
| 3. pád mn.č. | dviračiams | kmen |
| 4. pád mn.č. | dviračius | kmen |
| 5. pád mn.č. | dviračiai | kmen |
| 6. pád mn.č. | dviračiuose | kmen |
| 7. pád mn.č. | dviračiais | kmen |

Obrázek 3 – obrazovka detailu slova

Údaje vyplněné v obou tabulkách závisí na slovním druhu konkrétního slova. Na obrázku je jako příklad uvedeno podstatné jméno. To má několik základních vlastností: kmen, vzor, rod, informaci o tom, zda kmen nabízí a předá přízvuk, zda podstatné jméno tvoří obě čísla a také jestli se jedná o výjimku.

Všechny tyto údaje se zjistí z databáze a poté vyplní do první, informační tabulky. Druhá tabulka pak obsahuje tvary, kterých slovo může v jednotlivých pádech, případně rodech nebo časech nabývat. U podstatného a přídavného jména může být důležitý také údaj o tom, na které části slova se nachází přízvuk. Předávání přízvuku mezi kmenem a koncovkou se řídí pravidly popsány v části 1.2.4 a výsledek je zapsán taktéž do tabulky.

6.1.3 Editování slov

Volně poskytovaná data česko-litevského slovníku nezaručují stoprocentní správnost všech údajů. První problém se vyskytnul při převádění dostupného slovníku do databáze. U jednotlivých podstatných jmen totiž nebyl uveden vzor, podle kterého se skloňují. Ve většině případů se vzor dal automaticky rozpoznat, ale například vzory „širdis“ a „žodis“ mají stejnou koncovku.

Editace slova: bodlák - usnis (Podstatné jméno)

| | | | |
|------------------------|---|----------------------|--------------------------------------|
| Česky | <input type="text" value="bodlák"/> | 1. pád j.č.: | <input type="text" value="usnis"/> |
| Litevsky | <input type="text" value="usnis"/> | 2. pád j.č.: | <input type="text" value="usnies"/> |
| Vzor: | <input type="text" value="širdis"/> ▼ | 3. pád j.č.: | <input type="text" value="usni"/> |
| Kmen: | <input type="text" value="usn"/> | 4. pád j.č.: | <input type="text" value="usni"/> |
| Nabízí přízvuk: | <input checked="" type="checkbox"/> ano | 5. pád j.č.: | <input type="text" value="usnie"/> |
| Předá přízvuk: | <input checked="" type="checkbox"/> ano | 6. pád j.č.: | <input type="text" value="usnyje"/> |
| Plurál: | <input type="checkbox"/> pouze plurál | 7. pád j.č.: | <input type="text" value="usnimi"/> |
| Výjimka: | <input checked="" type="checkbox"/> ano | 1. pád mn.č.: | <input type="text" value="usnys"/> |
| | <input type="button" value="Upravit"/> | 2. pád mn.č.: | <input type="text" value="usnių"/> |
| | <input type="button" value="Smazāt"/> | 3. pád mn.č.: | <input type="text" value="usnims"/> |
| | | 4. pád mn.č.: | <input type="text" value="usnis"/> |
| | | 5. pád mn.č.: | <input type="text" value="usnys"/> |
| | | 6. pád mn.č.: | <input type="text" value="usnyse"/> |
| | | 7. pád mn.č.: | <input type="text" value="usnimis"/> |

Obrázek 4 – obrazovka editace slova

Při naplňování databáze jsem zjistil spoustu dalších nejasností, a tak je výskyt chyby téměř jistý. Uživatelům by asi velmi vadilo, kdyby jim aplikace vypisovala špatné tvary slova a oni neměli možnost opravy.

Proto má každý slovní druh příslušnou obrazovku, do které lze vyplnit všechny údaje jak o slově, tak o jeho tvarech. Z roletkové nabídky vzorů může uživatel vybrat jeden z jedenácti litevských vzorů a také může prostřednictvím zaškrtačacích políček opravit obecná pravidla týkající se konkrétního slova.

Pokud uživatel změní vzor nebo kmen, automaticky se v pravé části upraví všechny tvary slova. Za normálních okolností pravou část obrazovky uživatel nemůže editovat. Jestli však zadá, že slovo je výjimkou a neřídí se tudíž klasickými pravidly, zpřístupní se mu rovněž textová pole jednotlivých tvarů.

Je-li slovo uživatelem vložené – do databáze slov si jej tedy přidal sám – může ho i smazat. V takovém případě se příslušné tlačítko neobjeví neaktivní, nýbrž plně funkční.

6.1.4 Přidávání nových slov

Databáze litevských slovíček není všeobsáhlá a uživatelé by mohli ocenit možnost přidávání vlastních slov. Pokud si nějaké nové slovo získají z jiných zdrojů, například od učitele na hodině, mohou si jej posléze do svého slovníku doplnit.



Obrázek 5 – obrazovka přidání nového slova

V první fázi uživatel zadá základní tvar v českém i litevském jazyce a vybere slovní druh. Poté jej program právě v závislosti na vybraném slovním druhu přesměruje na obrazovku editace konkrétního druhu.

6.1.5 Zkoušení

Zkoušení je rozděleno do několika lekcí. V každé z nich mohou být obsaženy všechny typy cvičení. O jednotný vzhled a funkčnost celého zkoušení se opět stará speciální panel odvozený od třídy JPanel.

V něm je umístěno několik komponent včetně vnořeného panelu, který se liší mezi jednotlivými typy cvičení.

6.1.5.1 Překladové cvičení

Uživatel musí přeložit slovíčko v základním tvaru (tedy v prvním pádě jednotného čísla, infinitivu atd.). Obrazovka vypadá stejně jak pro překládání z češtiny do litevštiny, tak z litevštiny do češtiny.

Pokud uživatel slovo špatně přeloží, informuje ho o tom hlášením vpravo od tlačítka. Program také vypíše správnou odpověď.



Obrázek 6 – obrazovka zkoušení: překládání

6.1.5.2 Cvičení na skloňování

Cvičení na skloňování je společné pro podstatná jména i pro přídavná jména. Uživateli je předán základní tvar slova a on musí vyplnit druhý až sedmý pád jednotného čísla a také první až sedmý pád množného čísla. U přídavných jmen otázka specifikuje, zda má uživatel skloňovat v mužském nebo ženském rodě.

Všechny tvary, které nebyly zadány správně, aplikace zvýrazní a nechá uživatele prostudovat chyby.



Obrázek 7 – obrazovka zkoušení: skloňování podstatného jména

6.1.5.3 Cvičení na skloňování sloves

U sloves se nezkouší všechny tvary všech časů a způsobů naráz. Předpokládá se, že studenti se stejně nebudou učit více časů či způsobů najednou, ale spíše postupně. Zkoušení je tedy oddělené a program sdělí uživateli požadovaný úkol v otázce.

The screenshot shows a window titled "Vyčasujte v přítomném čase." (Conjugate in the present tense). It contains a list of verb forms and their corresponding conjugated forms in input fields:

| | |
|----------------------------|---------|
| jíst | valgyti |
| Přítomný čas - 1.os. j.č. | valgau |
| Přítomný čas - 2.os. j.č. | valgai |
| valgo | valgao |
| valgome | valgame |
| Přítomný čas - 2.os. mn.č. | valgote |

At the bottom, there is a "Další" (Next) button and a message: "Vyznačené odpovědi nebyly správné." (Marked answers were not correct).

Obrázek 8 – obrazovka zkoušení: časování slovesa v přítomném čase

Stejně jako v předchozím případě, i tentokrát program barevně odliší tvary, které uživatel neuvedl správně.

6.1.5.4 Výběr z několika možností

Někdy je potřeba zkoušet také jiné gramatické jevy, než jaké je program schopen pojmut. Může se jednat o spojování vedlejších vět, různé ustálené větné vazby nebo něco zcela jiného. Řešením je výběr z několika nabídnutých možností.

Pokud byla uživatelem zvolena špatná odpověď, aplikace upozorní na chybu a správnou odpověď opět barevně odliší.

The screenshot shows a window titled "Změkčování kmene podst. jm. se v mn. č. vždy provede:" (Softening of the stem of nouns in the plural is always done:). It contains four radio button options:

- a) u vzoru *namas*
- b) u vzoru *akmuo*
- c) u vzoru *širdis*
- d) u vzoru *žaltys*

At the bottom, there is a "Další" (Next) button and a message: "Špatná odpověď. Správná byla vyznačena." (Wrong answer. The correct one was marked).

Obrázek 9 – obrazovka zkoušení: výběr z několika možností

6.1.6 Uživatelská přívětivost

Při návrhu vzhledu aplikace jsem kladl důraz na co nejjednodušší ovládání. Snadné přechody mezi stránkami pomocí přesně popsaných tlačítek umožní ovládat program i méně zdatným uživatelům. Ti, kteří budou aplikaci používat dlouhodobě, možná ocení klávesové zkratky, pomocí nichž lze rychleji přepínat mezi jednotlivými obrazovkami (ukázka na obr. 10).

Mnoho uživatelů by mohlo mít problém se speciálními litevskými znaky. Na běžné klávesnici takové znaky nenajdeme. V panelu jazyků sice můžeme doinstalovat nový jazyk „litevština“, který změní rozložení kláves, i tak by ale mohly být s psaním znaků *ą*, *ė*, *ę*, *į*, *ų* a *ū* potíže. Uživatel si však může v nabídce speciálních znaků vložit potřebné písmeno do schránky a pak ho do příslušného textového pole vložit. Aby nemusel pokaždé myš najíždět na nabídku a pak se znovu vracet do textového pole, přidal jsem ke každému znaku klávesovou zkratku. Pomocí CTRL a konkrétní samohlásky se do schránky vloží požadovaný znak. Speciálních písmen „e“ a „u“ je více, proto je u jedné možnosti použita modifikační klávesa CTRL a u druhé ALT. Potřebuje-li uživatel vložit velké písmeno, stačí k dané kombinaci přidat ještě klávesu SHIFT.



Obrázek 10 – ukázka hlavní nabídky a nabídky speciálních znaků

7 Struktura balíčků

7.1 Spolupráce s databází

O spolupráci s databází se starají třídy balíčku „DB“. Každá tabulka databáze je zde zastoupená jednou třídou. Metody zajišťující výběr dat z databáze, jejich aktualizaci, přidávání a mazání obsahují konkrétní SQL dotaz, který je vykonán. Metody mohou vrátit konkrétní instanci třídy (například pokud se slovo hledá podle jeho ID), pravdivostní hodnotu (true / false) úspěchu aktualizace databáze nebo tzv. ResultSet (java.sql.ResultSet). ResultSet obsahuje množinu instancí dané třídy, které vyhovují konkrétní podmínce. Například při hledání ve slovníku se vrací nejprve množina slov odpovídající vloženému řetězci a poté množina slov začínající na uživatelův řetězec.

Databáze obsahuje 17 tabulek. Tříd v balíčku DB je 18 – poslední třída se stará o samotné spojení s konkrétní databází.

7.2 Panely obrazovek

Panely obrazovek se nacházejí v balíčku „litevstina“. Všech třináct základních obrazovek je reprezentováno třídou s příslušným názvem. Jedná-li se například o obrazovku na editování podstatného jména (UpravaPodstatne.java), je konstruktoru předáno také ID konkrétního substantiva.

Zkoušení se provádí v panelu Lekce. Do něj je v závislosti na typu cvičení vložen panel s textovými poli (případně přepínači), do nichž uživatel vyplní své odpovědi.

7.2.1 Panely zkoušení

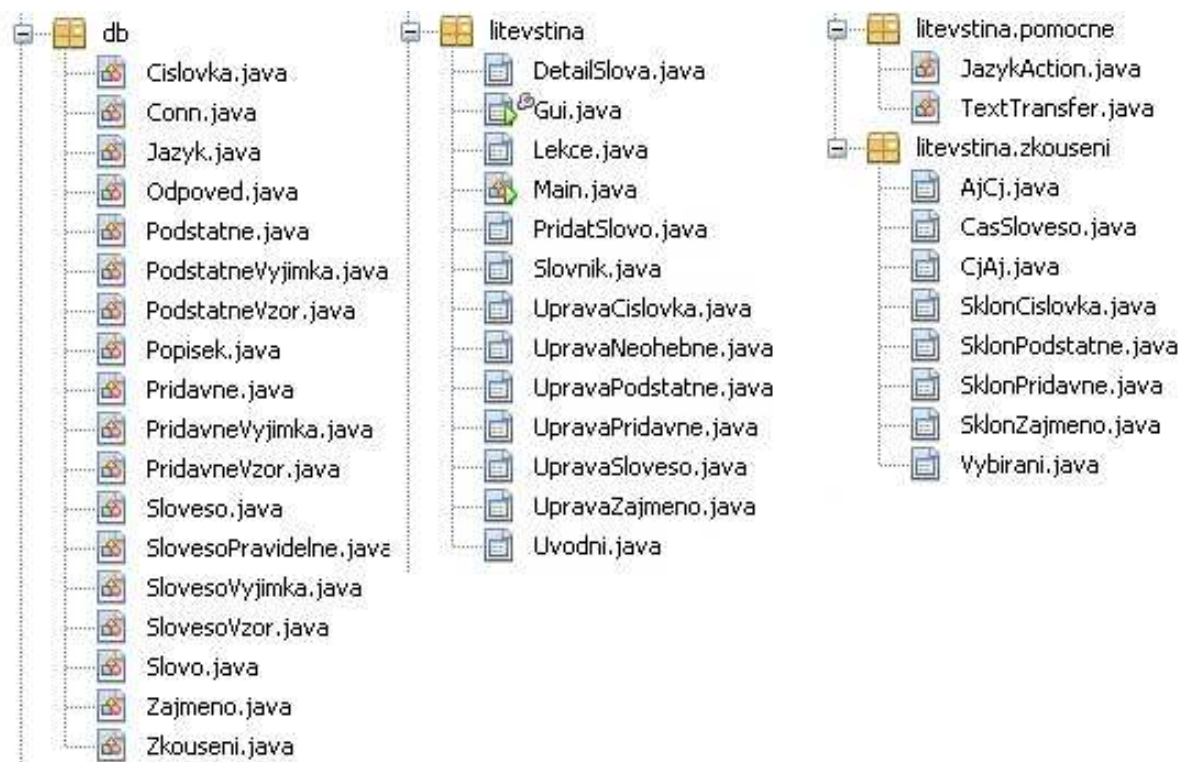
Panely zkoušení jsou součástí vnořeného balíčku „litevstina.zkouseni“. Tento balíček obsahuje celkem osm tříd. Různých typů cvičení je sice v této chvíli čtrnáct, nicméně mnohé z nich jsou natolik podobné, že se dají realizovat ve stejné obrazovce.

7.2.2 Pomocné třídy

Druhý vnořený balíček „pomocne“ obsahuje dvě třídy zajišťující pomocné funkce programu. Jedna z nich se stará o spolupráci se schránkou (kopírování speciálních znaků do schránky) a druhá je vytvořena pro účely snadnějšího přepínání mezi jazyky.

7.3 Importované balíčky

Pro grafické prvky na stránce jsou použity základní třídy balíčku javax.swing. Dále jsem využil některé třídy z balíčku java.sql – jako například ResultSet, Statement a jiné. Z balíčku java.awt jsem použil obecné třídy typu Dimension, Cursor apod.



Obrázek 11 – přehled tříd v jednotlivých balíčcích

8 Možnosti dalšího rozšíření programu

Aplikace byla vytvořena tak, aby splnila všechny požadavky, které na ni byly v začátku kladeny. Jejím používáním ovšem uživatelé jistě odhalí, jaké další funkce jim chybí, případně jaká rozšíření by si představovali.

Samostatnou kapitolou je pochopitelně obohacování slovní zásoby v databázi, případně přidávání dalších specifických vlastností jednotlivých slov. To však spadá do trochu jiné kategorie úkolů, které se netýkají přímo implementační části. Snad pouze zohlednění nových vlastností slov by si vyžádalo zásah do kódu.

8.1 Zkoušení

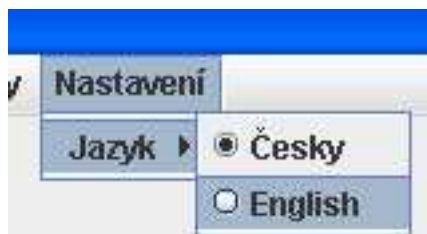
V této chvíli je v aplikaci implementováno čtrnáct základních typů cvičení. Některé se sice liší pouze minimálně – například časování sloves je rozděleno na šest samostatných úkolů pro všechny čtyři časy (přítomný, minulý, opakovací, budoucí) a dva způsoby (rozkazovací a podmiňovací). Separátní řešení má v tomto případě své opodstatnění. Studenti budou jednotlivé časy probírat postupně během roku, a tak by bylo nevhodné, aby po nich aplikace požadovala vyplnění i zbylých údajů.

Již zmíněných 14 typů cvičení pokrývá určitou část gramatických jevů, ale zdaleka ne všechny. Zde tím pádem vzniká prostor pro přidávání dalších úkolů.

8.2 Jazyky

Na jazykových školách ještě nemusí být úplně běžná přítomnost zahraničních studentů, avšak v případě vysokých škol to není nic neobvyklého. I tito žáci by si mohli zapsat kurzy litevštiny, jenže nedostatečná znalost češtiny by jim víceméně zabránila v používání programu.

Řešením by byla možnost přepínání mezi jazyky – například do angličtiny. Po přepnutí by se všechny popisky objevovaly v angličtině, zatímco aplikace by fungovala stále stejně. Toto budoucí rozšíření jsem při návrhu a implementaci vzal v úvahu a do jiného jazyka lze přepnout už nyní.



Obrázek 12 – nabídka jazyků

Přepínání do jiného jazyka by ovšem nemělo smysl, pokud by nebyla k dispozici rovněž slovní zásoba v jiné řeči. Jelikož jsem měl přístup pouze k volně šiřitelnému česko-litevskému slovníku,

nejsou anglické překlady v databázi vyplněny. Z tohoto důvodu tedy nemůže fungovat slovník – místo anglických významů se objeví pouze prázdná pole. Na zbylé funkce nemá přepnutí do jiného jazyka vliv – skloňování litevských jmen případně časování sloves probíhá samozřejmě stejně.

Vzhledem k tomu, že jsem neměl možnost zabudovat přepínání jazyků jako plnohodnotnou funkci, zůstává stále v oblasti budoucího rozšíření. Tudíž jsem ani nepřekládal všechny textové popisky, nýbrž pro ilustraci jen některé z úvodních obrazovek.

8.2.1 Správa zkoušení

Zkoušení je zatím v návaznosti na učebnici rozděleno do desíti lekcí. V kapitole 4 jsem rozebíral téma úpravy jednotlivých cvičení. Kdyby zvítězilo řešení internetovými stránkami, bylo by editování zkoušení mnohem snazší.

To ovšem neznamená, že by tato funkce nepřicházela v úvahu. Pokud by měli učitelé zájem o to, aby si mohli jednotlivá cvičení uzpůsobit, mohla by se doimplementovat jednoduchá správa zkoušení. Při každém přijmutí nových studentů by jazyková škola mohla rozdávat instalační CD s jinou databází.

9 Závěr

Z dostupných zdrojů jsem prozkoumal hlavní rysy litevského jazyka. Některé z nich jsou zachytitelné do databáze, případně do aplikační logiky. Zejména složitější gramatické konstrukce však není možné úplně popsat.

Analyzoval jsem požadavky a možnosti programu sloužícího pro podporu výuky cizího jazyka. Společně s poznatky o litevštině jsem získané informace použil na návrh funkcí programu a struktury databáze.

Databázi jsem rozdělil na dvě části. První z nich se snaží uchovat co nejvíce dat o gramatice litevštiny. Druhá část poté slouží k jedné ze stěžejních funkcí programu – zkoušení.

Na základě návrhu jsem v rámci diplomové práce vytvořil databázi a naplním ji daty. Slovní zásoba je volně přístupná, ale musela být převedena pro mé účely. Zkoušení a otázky byly vyplněny na základě učebnice litevštiny, ke které je tento program tvořen.

Aplikace byla implementována v programovacím jazyce Java. Splnila všechny požadavky kladené v původním zadání. Nabízí rovněž možnosti dalšího rozšíření.

10 Literatura

- [1] Erhart, A.: Litevština. SNP Praha, 1956.
- [2] Hoskovec, T.: Litevská gramatika. Studijní materiály Filozofické fakulty Masarykovy univerzity, 2006.
- [3] On-line výkladový slovník litevštiny. Dostupný na URL <http://www.lkz.lt>
- [4] Brůha, L.: Java, hotová řešení. Computer Press, Brno, 2003.
- [5] Chapman, S. J.: Začínáme programovat v jazyce Java. Computer Press, Praha, 2001.
- [6] Lithuanian language, Wikipedia. Dostupné na URL http://en.wikipedia.org/wiki/Lithuanian_language
- [7] On-line dokumentace programovacího jazyka Java. Dostupná na URL <http://java-sun.com>

11 Přílohy

Příloha č. 1: ER-diagram databáze litevštiny:

