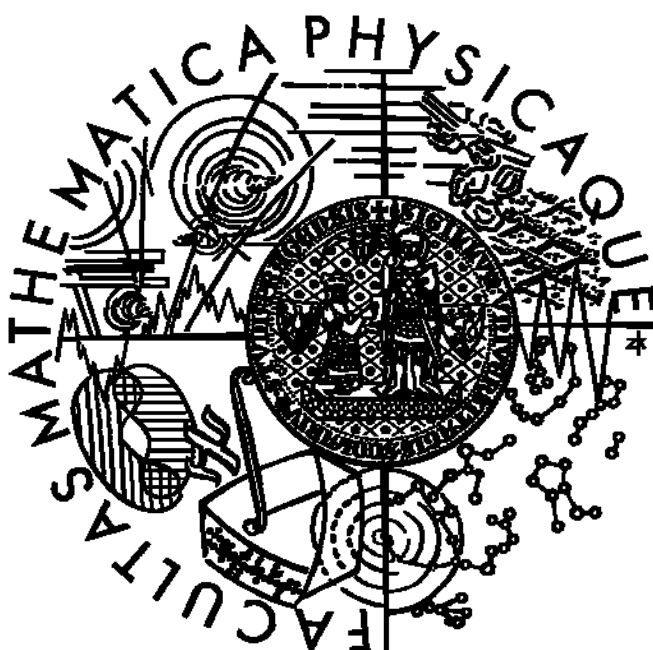


Univerzita Karlova v Praze

Matematicko-fyzikální fakulta

BAKALÁŘSKÁ PRÁCE



Vladislav Kozák

Informační systém pro školy

Katedra softwarového inženýrství

Vedoucí bakalářské práce: **Mgr. Mikuláš Patočka**

Studijní program: **Správa počítačových systémů**

2007

Ďakujem pánovi Mgr. Mikulášovi Patočkovi za ochotu, vynaložený čas, energiu a za odborné vedenie bakalárskej práce.

Prehlasujem, že som bakalársku prácu napísal samostatne a výhradne s použitím citovaných prameňov. Súhlasím zo zapožičaním práce.

V Prahe dňa 26. 7. 2007

Vladislav Kozák

Obsah

1. ÚVOD.....	6
1.1. ZÁKLADNÉ INFORMÁCIE O PROJEKTE	6
2. ANALÝZA NÁVRHU	7
2.1. UPRESNENIE ZADANIA	7
2.2. UŽIVATEĽSKÉ ROZHRANIE	8
2.3. DATABÁZOVÝ SERVER	9
2.4. ER MODEL	9
3. PREHĽAD TABULIEK.....	13
3.1. TABUĽKA SK_UCITELIA	13
3.2. TABUĽKA SK_PREDMETY	14
3.3. TABUĽKA SK_SADY	15
3.4. TABUĽKA SK_PRED_SADY	15
3.5. TABUĽKA SK_TRIEDY	16
3.6. TABUĽKA SK_OKRESY	18
3.7. TABUĽKA SK_ZIACI	18
3.8. TABUĽKA SK_ZNAMKY	20
3.9. TABUĽKA SK_LOGY	21
4. VEDĽAJŠIE OKNÁ APLIKÁCIE	23
4.1. OKNO PRE PRIHLÁSENIE	23
4.2. NASTAVENIA	24
4.3. SPRÁVA OKRESOV	26
4.4. ZADÁVANIE ZNÁMOK	27
4.5. TLAČENIE VYSVEDČENÍ.....	29
4.6. FORMÁT VYSVEDČENIA.....	30
4.7. ZNÁMKOVANIE	32
5. HLAVNÉ OKNO APLIKÁCIE	33
5.1. POPIS.....	33
5.2. ZÁLOŽKA PREDMETY	33
5.3. ZÁLOŽKA ŽIACI	35
5.4. ZÁLOŽKA TRIEDY	36
5.5. ZÁLOŽKA SADY	38
5.6. ZÁLOŽKA ZNÁMKY	39
5.7. ZÁLOŽKA LOGY	40
6. UŽIVATEĽSKÁ PRÍRUČKA	43
6.1. POŽIADAVKY	43
6.2. SPUSTENIE	43
6.3. ZMENY NASTAVENÍ A PROSTREDIA.....	44
6.4. SPRÁVA UŽIVATEĽOV	45
6.5. HLAVNÉ FUNKCIE APLIKÁCIE.....	46
6.6. TLAČ VYSVEDČENÍ	49

7. ZÁVER	52
7.1. ZMENY PRI VÝVOJI	52
7.2. ZHRNUTIE	52
8. BIBLIOGRAFIA	53

Názov práce: Informační systém pro školy

Autor: Vladislav Kozák

Katedra: Katedra softwarového inženýrství

Vedúci bakalárskej práce: Mgr. Mikuláš Patočka

e-mail vedúceho: mikulas@artax.karlin.mff.cuni.cz

Abstrakt: Predmetom práce je vytvorenie informačného systému pre stredné a základné školy. Aplikácia umožňuje ukladať a archivovať dáta súvisiace so školou a hromadne tlačiť vysvedčenia. Systém má intuitívne ovládanie a obsahuje prehľadné grafické rozhranie pre nastavenie tlače. Aplikácia pracuje s lokálnym alebo vzdialeným databázovým serverom a podporuje viac užívateľský prístup.

Klíčové slová: informačný systém, tlač vysvedčení, škola

Title: Information system for schools

Author: Vladislav Kozák

Departement: Department of Software Engineering

Supervisor: Mgr. Mikuláš Patočka

Supervisor's e-mail address: mikulas@artax.karlin.mff.cuni.cz

Abstract: The task of the work is to develop information system for primary and secondary schools. The application allows to insert and store data related with school and performs mass printing of school reports. The user interface is easy to use and includes graphic interface for setting the printer output. The application communicates with database server and supports parallel access.

Keywords: information system, report printing, school

1. Úvod

1.1. Základné informácie o projekte

Inšpirácia

Informatizácia školstva je v súčasnej dobe často diskutovanou témou. Hlavným cieľom je uľahčiť a zefektívniť časovo náročné, ťažkopádne ale aj rutinné úkony. Miesto rozsiahlych archívov naplnených tonami papierov by sme radšej chceli vidieť „nejaký ten server s diskovým políčkom“. Niekedy sa ale stáva, že nám aplikácie prácu skôr komplikujú. Častejší je prípad, kedy aplikácia prácu uľahčuje, ale zďaleka nie do takej miery, aká by sa v tejto problematike očakávala.

Niekoľko rokov skúseností s používaním aplikácií slúžiacich na uchovávanie informácií o študentoch, ich študijných výsledkoch, ale najmä na hromadnú tlač vysvedčení ma inšpirovalo k tomu, aby som navrhol a naprogramoval podobnú vlastnú aplikáciu. Zameram sa na predovšetkým na praktickosť a flexibilitu, ale i robustnosť prevedenia. Spomínané aplikácie mi prišli nešikovné práve v týchto smeroch a často som narážal na neschopnosť spomínaných aplikácií nastaviť užívateľské vlastnosti konkrétnym potrebám.

Zadanie problému a cieľ

Cieľom je naprogramovať robustnú aplikáciu, ktorá bude slúžiť na obsluhu a uschovávanie všetkých potrebných informácií o učiteľoch, predmetoch, študentoch a ich záverečných hodnoteniach. Hlavnou funkciou aplikácie bude vyhotovenie a hromadná tlač vysvedčení na konci každého polroku resp. roku. Riešenie by preto malo obsahovať prehľadné rozhranie na nastavenie výstupu na tlačiareň a samotnú tlač.

Zameriam sa na praktickosť riešenia, aby práca s aplikáciou z pohľadu užívateľa bola efektívna a jednoduchá. Podstatou je, aby užívateľské rozhranie bolo značne flexibilné – najmä rozhranie pre zadávanie výsledkov a pre nastavenie výstupu na tlačiareň. Ovládacie prvky a funkcie aplikácie musia pôsobiť intuitívne. Riešenie bude obsahovať ešte množstvo ďalších vylepšení a nápadov, ktoré budú popísané v ďalších kapitolách. Samozrejmosťou je viac užívateľský režim (paralelný prístup).

2. Analýza návrhu

2.1. Upresnenie zadania

Cieľová skupina

Cieľovou skupinou budú všetky stredné a základné školy, ktoré vystavujú žiakom vysvedčenia a chcú ušetriť učiteľov od otrockého vypisovania a následného prepisovania vysvedčení pri každej drobnej chybe. Súčasťou požiadaviek tejto skupiny je aj možnosť archivovať výsledky a nahliadnuť do výsledkov z pred niekoľkých rokov. Aplikácia predpokladá, že budúci užívatelia budú používať platformu Microsoft Windows.

Výber programovacieho jazyka

Vyberaný programovací jazyk a vývojové prostredie muselo spĺňať určité základné predpoklady. Prvým z nich bol, že vybraný jazyk bude patriť medzi vyššie programovacie jazyky. Táto potreba vyplýva z predpokladu, že užívateľské rozhranie má byť prehľadné a ovládanie intuitívne. Je tiež nutné, aby programovací jazyk obsahoval rozumné nástroje na komunikáciu s databázovým serverom. Výsledkom náročného výberu bolo nakoniec rozhodnutie programovať aplikáciu v Delphi verzii 6.

Rozšírenie zadania

Z programátorského hľadiska ide o databázovú aplikáciu, ktorá komunikuje s databázovým serverom pomocou dotazovacieho jazyka SQL. Bude sa skladať z dvoch hlavných vrstiev – z databázovej a aplikačnej. Z dôvodu väčšej efektivity sa budem snažiť naprogramovať na databázovej úrovni čo najviac užitočných procedúr a funkcií a tým ušetriť časové náklady na prenos dát medzi serverom a aplikáciou.

Súčasťou riešenia bude tiež implementácia viacerých známych algoritmov, napr. hashovací algoritmus pre ukladanie hesiel. Pri tlačení vysvedčení je nevyhnutné dať užívateľovi právo tlač kedykoľvek prerušiť. Aplikácia bude posilať dáta na tlačiareň a súčasne musí striehnuť či nebola poslaná požiadavka na ukončenie tlače. To si vynúti použitie súbežných vlákien (Multi-Thread).

Paralelný prístup budem zabezpečovať uzavieraním kritických sekcií do transakcií¹ a nastavením príslušného stupňa izolácie transakcií, čím zaručím ich atomické prevedenie a tým zamedzím aj pri paralelnom behu tvorbe dátových anomálií ako napr. inkonzistencii dát. Aplikácia bude komunikovať s databázovým serverom prostredníctvom SQL dotazov, ktorých obsah sa

¹ Databázová transakcia je skupina príkazov, ktoré prevedú databázu z jedného konzistentého stavu do druhého, pričom musia spĺňať 4 základné vlastnosti tzv. ACID (atomicitu, konzistenciu, izolovanosť, trvalosť). Ak je transakcia atomická, musí prebehnúť celá, alebo neprebehnúť vôbec.

nachádza buď priamo v zdrojovom kóde, alebo sa načítavajú zo skriptových súborov uložených na disku.

Dôležité je zabezpečiť, aby dáta aplikácie boli chránené pred neoprávneným prístupom. Každý užívateľ aplikácie by mal byť autentifikovaný užívateľským menom a heslom, ktoré sú nutnou podmienkou pre prístup k dátam a funkciám aplikácie. Potrebná je tiež schopnosť rozlišovať medzi týmito užívateľmi a nastavovať im práva. Na základe ich práv im potom umožňovať prístup iba k vybranej podmnožine ovládacích prvkov.

2.2. Užívateľské rozhranie

Vzhľad

Ak má aplikácia umožňovať rýchle a intuitívne ovládanie, je návrh grafického rozhrania (GUI) veľmi dôležitá súčasť projektu. Nepredpokladá sa, že úroveň infromatických znalostí potencionálneho užívateľa bude vysoká. Práve naopak, prácu s programom by mal zvládnuť aj učiteľ minimálnymi znalosťami práce na PC. Pri každej operácii sa musíme postarať o zamedzenie prístupu k ovládacím prvkom, ktoré by mohli narušiť jej správny priebeh a výsledok.

Aplikácia sa bude skladať z hlavného okna, ktoré má niekoľko záložiek a z vedľajších okien. Každá záložka hlavného okna bude obsahovať skupinu ovládacích prvkov na obsluhu istých logicky súvisiacich podmnožín dát. Časom sa ukázalo, že bude praktické, aby záložiek bolo 7, pričom jedna z nich bude mať čisto infromatívny charakter.

Vedľajšie okná budú slúžiť na sprístupnenie ovládacích prvkov, ktoré sa esteticky alebo funkčne nehodili na žiadnu zo záložiek hlavného okna, napr. okno pre nastavenia, prihlasovacie okno, okno na tlačenie a iné. Detailný popis okien a ovládacích prvkov sa nachádza v kapitole 6.5.

Práva

Dôležitá je požiadavka na zabezpečenie aplikácie. Užívatelia sa preto budú do aplikácie prihlasovať užívateľským menom a heslom a podľa ich práv sa im sprístupnia tie ovládacie prvky, ktoré im náležia. Pri vymýšľaní spôsobu, akým uchovávať a využívať informácie o užívateľských právach ma napadli 2 varianty.

Pri prvom variante som počítal s tým, že práva sa dajú usporiadať hierarchicky a akonáhle bude mať užívateľ právo na istej úrovni, automaticky mu patria práva na nižších úrovniach. V tom prípade by sa zdalo rozumné pamätať si iba najvyššiu úroveň práv pre užívateľa. V druhom variante som sa rozhodol práva rozdeliť do 6-tich skupín, a práva užívateľa by jednoznačne určovalo nezáporné celé číslo, ktorého binárny zápis by určoval jednotlivé povolenia na skupiny práv.

Nakoniec vyhral variant druhý, pretože sa mi javil viac flexibilný a šikovný. Práva užívateľa bude možné šikovne zistiť pomocou binárneho operátora AND. Ovládacie prvky hlavného okna boli rozmiestnené na 6

záložiek tak, aby odpovedali šiestim skupinám práv užívateľa. Užívateľovi budú po identifikácii prístupné iba záložky, na ktoré má právo.

2.3. Databázový server

Alternatívy

Už v návrhu bolo rozhodnuté, že na komunikáciu s databázou bude použitý dotazovací jazyk SQL. Pri vyberaní databázového servera som zvažoval viacero alternatív. Medzi tie najlepšie varianty patrili Microsoft SQL Server 2000 alebo 2005 a viacero verzií Oracle. Databázové systémy Oracle sú považované za veľmi stabilné, bezpečné a s množstvom vstavaných funkcií a procedúr. Na druhej strane SQL stroje od Microsoftu sú príznačné tým, že väčšinu funkcií a procedúr požadujú od užívateľa a len málo z nich je implementovaných od výrobcu. Nezanedbateľná je tiež finančná dostupnosť, ale aj samotná veľkosť tohto databázového systému.

Voľba

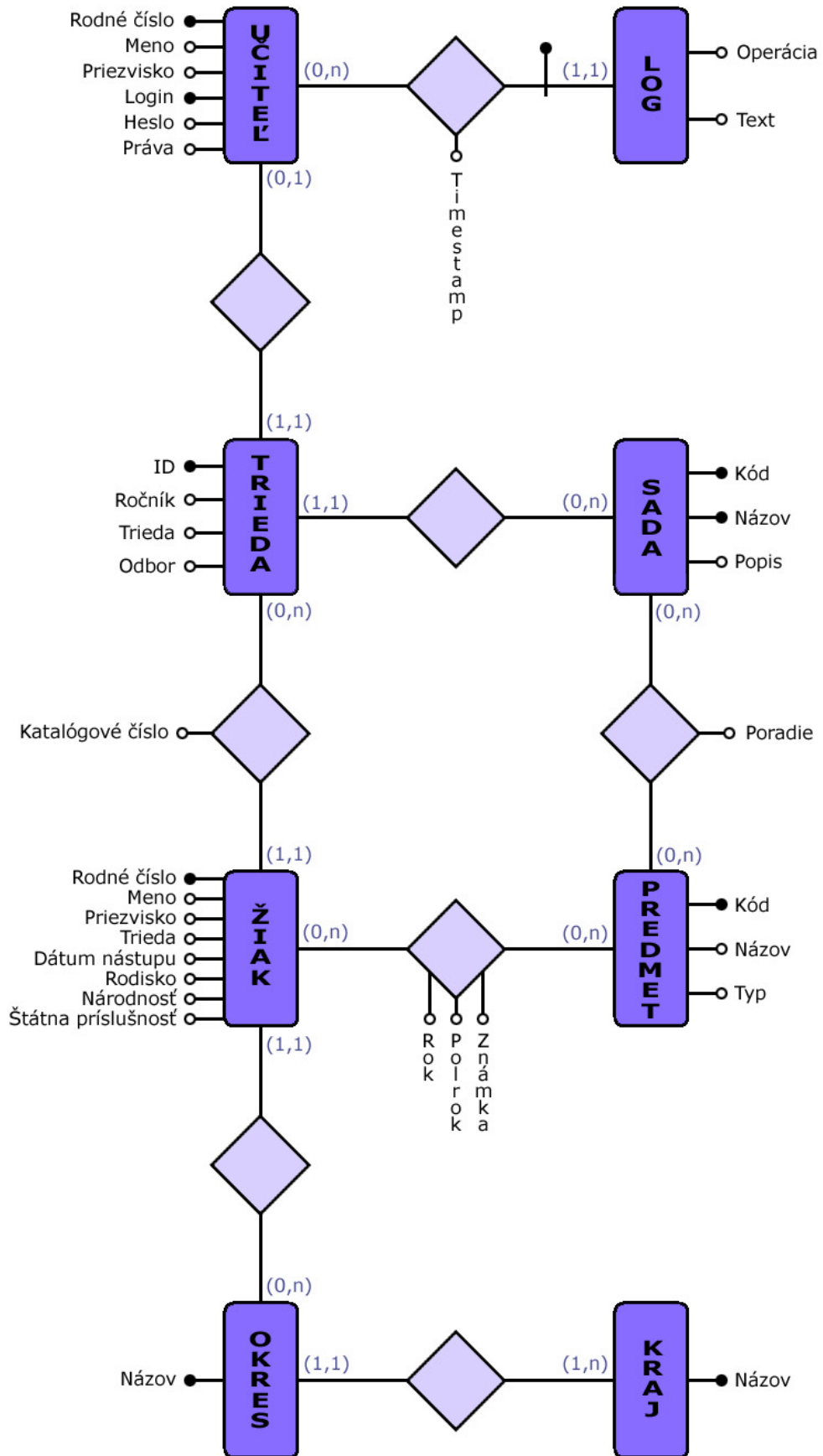
Pri rozhodovaní som si uvedomil, že moje požiadavky na funkcie a procedúry sú príliš špecifické, preto sa mi väčšina z vstavaných procedúr a funkcií Oracle javila pre moje riešenie nepoužiteľná. Ďalším faktom, ktorý zavážil, je ťažká finančná dostupnosť databázového servera Oracle. Pričom Microsoft dovoľuje voľné stiahnutie a používanie MS SQL Server 2005 na nekomerčné účely s obmedzenou funkčnosťou. V neposlednom rade hovorili proti Oracle aj jeho diskové a procesorové nároky. Rozhodol som sa preto pre MS SQL Server 2005 Express, ktorý lepšie vyhovuje našim požiadavkám.

2.4. ER Model

E-R diagram

Prvým krokom pri návrhu databázovej aplikácie je zhotovenie E-R diagramu². E-R diagram pre moju databázovú aplikáciu si môžete pozrieť na obr.č. 1. Popis jednotlivých vzťahov ako aj vysvetlenie, prečo bol model a databáza navrhnutá práve takýmto spôsobom sú podané v ďalšom odseku.

² Entitno-relačný diagram je základnou komponentou modelu a používa sa na vizuálnu reprezentáciu dát. E-R diagram graficky znázorňuje jednotlivé entity a relácie medzi nimi a hrá podstatnú úlohu pri návrhu tabuliek, optimalizácii a odstraňovaní redundancií v databáze.



obr.č. 1 – Entitno-Relačný Diagram

Vysvetlenie vzťahov

Dôvodov, prečo navrhnúť databázu spomínaným spôsobom je hneď niekoľko. Je nutné vysvetliť niektoré pojmy. Sada je pre nás postupnosť predmetov, ktorá odpovedá poradiu predmetov tak, ako sú zapísané v triednej knihe. Dôvod, prečo je pre nás daná informácia zaujímavá je prostý a vyplýva z cieľa spraviť aplikáciu ľahko a rýchlo ovládateľnú. Na vlastnej skúsenosti som pocítil, že je prakticky nemožné efektívne a rýchlo zadávať žiakom známky, pokiaľ ich poradie v triednej knihe a v aplikácii je rozdielne.

Dôležitý postreh je, že každá trieda má práve jednu sadu (pretože má práve jednu triednu knihu). Takže pre vzťah trieda-sada nie je samostatná tabuľka, ale informácie o sade triedy je priamo v tabuľke „Triedy“.

Normalizácia

Normalizácia modelu je proces, pri ktorom dochádza k úprave databázového modelu, aby sa eliminovala možnosť vzniku istých dátových anomálií, ako napríklad redundancia³. Normalizácia je proces, ktorý je presne popísaný štandardizovaným algoritmom. Pri databázovom modeli našich rozmerov však nebolo potrebné algoritmus implementovať.

Pre väčšinu databáz stačí prevod do 3NF⁴, dokonca pri vyšších stupňoch normalizácie môže dochádzať k poklesu výkonnosti. Z týchto dôvodov som sa rozhodol môj relačný model normalizovať do 3NF.

Indexy

Podľa [3] je index štruktúra, ktorú poskytujú databázové systémy, aby zlepšili spracovanie dát a konkrétny databázový systém stanoví, ako má index pracovať a ako ho bude využívať. V praxi slúžia indexy na zrýchlenie prístupu k dátam na základe podmienky vo WHERE klauzule. Nemenia syntax ani sémantiku príkazov. Dôležité je si uvedomiť, že index treba používať iba tam, kde výrazne pomôže často kladeným dotazom, inak iba zbytočne spomaľuje aktualizáciu dát. Indexy treba tiež vytvárať pre všetky cudzie kľúče, pretože sa zrýchľujú operácie s nadriadenou tabuľkou.

Predošlý teoretický odstavec ukázal, ako treba pri návrhu a vytváraní indexov postupovať. Podstatným krokom bolo rozmyslieť si, ako často bude prípadný index využívaný a aká veľká bude selektivita daného indexovaného atribútu. Na základe toho som musel rozhodnúť, či je vhodné index použiť a ak áno, tak aký typ indexu použiť. Pre primárne kľúče nebolo potrebné indexy explicitne vytvárať, pretože sa o vytvorenie indexu postará databázový systém.

³ Redundancia v databáze je nadbytočné uchovávanie informácií na viacerých miestach, čím dochádza k zvyšovaniu časových nárokov pri aktualizácii a úprave dát.

⁴ 3.normálna forma je množina pravidiel, ktorých dodržanie vedie k optimálnejšiemu prístupu k dátam v databáze. Je nadmnožinou 1. a 2. normálnej formy.

Výsledok

Výsledkom tvorby ER modelu, jeho normalizácie, kreslenia diagramu a analýzy všetkých požiadaviek je zoznam tabuliek, ktorého detailnejší prehľad vrátane zoznamu atribútov, kľúčov a indexov nájdete v nasledujúcej kapitole.

3. Prehľad tabuliek

3.1. Tabuľka Sk_Ucitelia

Popis

Tabuľka obsahuje zoznam učiteľov na škole a informácie o nich. Každý učiteľ je zároveň aj užívateľ aplikácie, preto budem tieto pojmy považovať za ekvivalentné a využívať ten, ktorý sa v danej spojitosti viac hodí. Správu užívateľov má na starosti unikátny užívateľ „admin“, ktorý jediný má prístup k záložke pre správu užívateľov v okne *Nastavenia*. V definíciách tabuliek budem zámerne kvôli prehľadnosti vynechávať obsah niektorých obmedzení, prípadne budem kľúčové slovo „CONSTRAINT“ skracovať na „CON“.

Zoznam atribútov

- **RC** – rodné číslo učiteľa, správnosť tvaru rodného čísla kontroluje obmedzenie *Valid_U_RC*, je to primárny kľúč tabuľky
- **Meno** – meno učiteľa, dĺžka maximálne 15 znakov
- **Priezvisko** – priezvisko učiteľa, maximálne 15 znakov
- **ULogin** – login učiteľa, ktorým sa prihlasuje do aplikácie, v prípade, že nie je pri vytváraní explicitne určený, je vygenerovaný automaticky; je to kandidátny kľúč tabuľky
- **PW** – heslo užívateľa, je to 32 znakový hash hesla (MD5), ktorým sa užívateľ identifikuje pri prihlasovaní
- **Prava** – je nezáporné celé číslo, ktoré odpovedá právam užívateľa pristupovať k jednotlivým záložkám s dátami a upravovať ich. Záložky sú usporiadané a jednotlivé bity bitovej reprezentácie čísla *Prava* určujú, na ktoré z nich má právo. Práva sa overujú bitovým operátorom *AND*.

Definícia tabuľky

```
CREATE TABLE Sk_Ucitelia (  
  RC CHAR(11)  
    CONSTRAINT Valid_U_RC CHECK (...)  
    CONSTRAINT PK_U_RC PRIMARY KEY  
  Meno VARCHAR(15) NOT NULL,  
  Priezvisko VARCHAR(15) NOT NULL,  
  ULogin VARCHAR(10) NOT NULL  
    CONSTRAINT SK_U_Login UNIQUE,  
  PW CHAR(32) NOT NULL,  
  Prava INT NOT NULL );
```

3.2. Tabuľka Sk_Predmety

Popis

Tabuľka uchováva údaje o predmetoch, ktoré sa vyučujú na škole. Každý predmet je jednoznačne určený svojim kódom. Predmety sú potom zapisované do sád, jednotlivým študentom sa zapisujú niektoré predmety zo sady ich triedy.

Zoznam atribútov

- **Kod** – kód predmetu, ktorý tvorí práve 5 znakov. Korektnosť tvaru kódu overuje obmedzenie *Valid_Kod*. Je to primárny kľúč (*PK_Kod*).
- **Nazov** – názov predmetu. Tento názov bude použitý aj pri tlačení vysvedčení, preto aplikácia pri vkladaní používa na hodnotu tohto atribútu funkciu *UpCaseFirst*.
- **Typ** – určuje typ predmetu. Jedná sa o celé číslo, ktoré leží v intervale <1,4> a určuje ako je predmet známkovaný. Keď má predmet Typ=1 znamená to, že sa jedná o voliteľný predmet. Ak má predmet Typ=2, je to predmet s povinnou dochádzkou a je hodnotený známami A/N (absolvoval/neabsolvoval) a ak je Typ=3 ide o povinný známkovaný predmet, pričom známka leží na stupnici od 1 do 5. Typ=4 určuje, že sa jedná o hodnotenie správania. Dôvod, prečo sme pre správanie vytvorili samotný typ je rozdielna stupnica ako aj slovné hodnotenie známok pre bežné predmety a správanie. Každý predmet ešte môže byť ospravedlnený, vtedy mu je pridelená „známka“ O. Na kontrolu správneho typu predmetu slúži tabuľkové obmedzenie *Valid_Typ*.

Definícia tabuľky

```
CREATE TABLE Sk_Predmety (  
    Kod CHAR(5)  
        CONSTRAINT PK_Kod PRIMARY KEY  
        CONSTRAINT Valid_Kod CHECK (LEN(Kod) = 5),  
    Nazov VARCHAR(20) NOT NULL,  
    Typ INT NOT NULL  
        CONSTRAINT Valid_Typ CHECK (Typ>0 AND Typ<5));
```

Indexy

Tabuľka má 2 indexy. Prvý z nich je *Sk_Predmety(Typ)* a jeho potreba súvisí s vysokou selektivitou tohto stĺpca, keďže nadobúda iba jednu zo štyroch hodnôt. Druhý index sa viaže na atribút *Názov*, podľa ktorého budú riadky často utriedované.

3.3. Tabuľka Sk_Sady

Popis

Tabuľka obsahuje zoznam sád. Sadu triedy sme definovali ako postupnosť predmetov, ktorá kopíruje hlavičku triednej knihy. Hlavný dôvod, prečo som si sady definoval a implementoval do návrhu je uľahčenie zadávania známok žiakovi na konci roku. Rozsiahlejšiemu popisu dôvodov a funkcii sád sa venuje odsek Vysvetlenie vzťahov kapitoly 2.4. Sada je identifikovaná kódom alebo názvom.

Zoznam atribútov

- **Kod** – kód sady. Jedná sa o kladné celé číslo. Kód sady tvorí primárny kľúč tabuľky PK_Sady.
- **Nazov** – názov sady. Tento atribút by mal slúžiť na ľahšiu identifikáciu sady. Názov je neprázdny reťazec dĺžky menšej ako 15, ktorý by mal byť unikátny. Typickým príkladom hodnoty atribútu je „Prvaci AJ-FR“, ktorý by napr. identifikoval sadu pre triedy prvého ročníka, ktorej žiaci majú za cudzie jazyky zvolený Anglický a Francúzsky jazyk.
- **Popis** – atribút slúži na rozsiahlejší popis sady. Je to reťazec dĺžky až do 30 znakov, ktorého text uchováva pomocnú informáciu o sade.

Definícia tabuľky

```
CREATE TABLE Sk_Sady (  
  Kod INT  
  CONSTRAINT PK_Sady PRIMARY KEY,  
  Nazov VARCHAR(15) NOT NULL  
  CONSTRAINT SK_S_Nazov UNIQUE,  
  Popis VARCHAR(30) );
```

3.4. Tabuľka Sk_Pred_Sady

Popis

Tabuľka odpovedá relácii medzi sadou a predmetom. Určuje ktoré predmety sa nachádzajú v ktorých sadách a v akom poradí. Primárny kľúč tvorí kód sady spolu s poradím.

Zoznam atribútov

- **Kod_S** – kód sady. Obmedzenie *FK_PS_Sady* určuje, že tento atribút je cudzí kľúč odkazujúci sa na atribút *Kod* tabuľky *Sk_Sady*. Pri mazaní alebo úprave sa zmeny prejavajú kaskádovito.

- **Kod_P** – kód predmetu. Jedná sa o cudzí kľúč odkazujúci sa na atribút *Kod* tabuľky *Sk_Predmety*. Správnosť kontroluje obmedzenie *FK_PS_Predmety*. Prípadná zmena hodnoty tohoto kľúča v materskej tabuľke sa prejaví opäť kaskádovito.
- **Por** – poradie predmetu. Tento atribút určuje poradie predmetu s kódom *Kod_P* v sade s kódom *Kod_S*. Keďže v každej sade sa každý predmet musí nachádzať na inom mieste, tvorí dvojica (*Kod_S, Por*) primárny kľúč tabuľky.

Definícia tabuľky

```
CREATE TABLE Sk_Pred_Sady (  
    Kod_S INT NOT NULL  
        CONSTRAINT FK_PS_Sady FOREIGN KEY REFERENCES  
    Sk_Sady(Kod)  
        ON UPDATE CASCADE  
        ON DELETE CASCADE,  
    Kod_P CHAR(5) NOT NULL  
        CON FK_PS_Predmety FOREIGN KEY REFERENCES  
    Sk_Predmety(Kod)  
        ON UPDATE CASCADE,  
    Por INT NOT NULL  
        CONSTRAINT PK_Psady PRIMARY KEY (Kod_S, Por));
```

Indexy

Tabuľka obsahuje iba indexy pre cudzie kľúče, ktoré sú tu 2. Jeden sa odkazuje na kód predmetu do tabuľky predmetov a druhý na kód sady do tabuľky sád.

3.5. Tabuľka Sk_Triedy

Popis

Tabuľka obsahuje zoznam tried, ktoré sú na škole. Trieda je jednoznačne identifikovaná svojím ID, alebo dvojicou atribútov (ročník, trieda). Každá trieda má práve jednu sadu a práve jedného triedneho učiteľa. Každý učiteľ môže byť triednym učiteľom maximálne jednej triede.

Zoznam atribútov

- **ID** – dvojnakový identifikátor triedy. Je to primárny kľúč (*PK_Triedy*).
- **Rocnik** – ročník triedy. Reprezentuje ho celé číslo z intervalu <1,9>. Správnosť ročníku kontroluje obmedzenie *Valid_Roc*
- **Trieda** – jednoznaková hodnota rozlišuje triedy v rovnakom ročníku. Hodnota tohto atribútu by mala odpovedať jednému písmenu, napríklad „A“.

- **KodSady** – kód sady triedy. Atribút je typu integer. Je to cudzí kľúč odkazujúci sa na atribút *Kod* tabuľky *Sk_Sady*. Pri zmene hodnoty tohto atribútu v materskej tabuľke sa zmeny prejavajú kaskádovito.
- **Ucitel** – rodné číslo triedneho učiteľa. Opäť sa jedná o cudzí kľúč, tento krát sa odkazuje na atribút *RC* tabuľky *Sk_Ucitelia*. Je to kandidátny kľúč⁵.
- **Odbor** – slúži na uchovanie informácie o zameraní triedy. Jedná sa o reťazec premenlivej dĺžky s obmedzením na 30 znakov. Typickou hodnotou tohto atribútu by mohlo byť „Informatika“.

Definícia tabuľky

```
CREATE TABLE Sk_Triedy (  
  ID CHAR(2)  
  CONSTRAINT PK_Triedy PRIMARY KEY,  
  Rocnik INT NOT NULL  
  CONSTRAINT Valid_Roc CHECK (Rocnik<10 AND Rocnik>0),  
  Trieda CHAR(1) NOT NULL  
  CON Valid_Tri CHECK (Trieda >= 'A' AND Trieda <=  
  'Z'),  
  KodSady INT NOT NULL  
  CONSTRAINT FK_T_Sady FOREIGN KEY REFERENCES  
  Sk_Sady(Kod)  
  ON UPDATE CASCADE,  
  Ucitel CHAR(11) NOT NULL UNIQUE  
  CON FK_T_Ucitel FOREIGN KEY REFERENCES  
  Sk_Ucitelia(RC)  
  ON UPDATE CASCADE,  
  Odbor VARCHAR(30));
```

Indexy

Tabuľka má 2 cudzie kľúče. Atribút *Ucitel* je odvodený z primárneho kľúča tabuľky *Sk_Ucitelia* – rodného čísla učiteľa. *KodSady* sa odkazuje do tabuľky so sadami na atribút *Kod*. Pre oba tieto atribúty je teda vytvorený index. Indexovaná je tiež dvojica (*Ročník*, *Trieda*), ktorá je často používaná vo *WHERE* klauzule.

⁵ Kandidátny kľúč je atribút, ktorý má vlastnosti NOT NULL a UNIQUE, teda nadobúda neprázdnych jednoznačne identifikovateľných hodnôt.

3.6. Tabuľka Sk_Okresy

Popis

Tabuľka bola navrhnutá kvôli potrebe uchovávať informácie o okrese a kraji, v ktorom boli žiaci narodený. Keďže každý okres má práve jeden kraj, postačí nám jediná tabuľka.

Zoznam atribútov

- **Okres** – reťazec dĺžky maximálne 15 znakov, ktorý odpovedá názvu okresu. Je to primárny kľúč tabuľky
- **Kraj** – kraj okresu. Neprázdny názov kraja, do ktorého patrí daný okres.
- **CFG** – je hodnota typu bit. Rozlišuje, či sa jedná o kraj, ktorý bol pridaný defaultne, alebo bol pridaný nejakým užívateľom aplikácie. Prvú skupinu okresov dokáže obsluhovať iba administrátor aplikácie.

Definícia tabuľky

```
CREATE TABLE Sk_Okresy (  
    Okres VARCHAR(15)  
    CONSTRAINT PK_Okres PRIMARY KEY,  
    Kraj VARCHAR(15) NOT NULL,  
    CFG BIT );
```

3.7. Tabuľka Sk_Ziaci

Popis

Tabuľka uchováva informácie o žiakoch, ktorý sa nachádzajú na škole. Jednoznačná identifikácia žiaka je jeho rodné číslo. Každý žiak má práve jednu triedu.

Zoznam atribútov

- **RC** – rodné číslo žiaka. Rodné číslo by malo byť v tvare s lomítkom, takže typ atribútu je reťazec dĺžky 11, ktorého znaky sú číslice od 0 do 9 s výnimkou siedmeho znaku – lomítka. Správny tvar rodného čísla kontroluje obmedzenie *Valid_RC*. Rodné číslo je primárny kľúč tabuľky.
- **Meno** – meno žiaka. Typ atribútu je reťazec maximálnej dĺžky 15.
- **Priezvisko** – priezvisko žiaka. Typ atribútu je zhodný s typom atribútu *Meno*.
- **Trieda** – je identifikátor triedy žiaka. Je to neprázdna hodnota, ktorá sa odkazuje do tabuľky Sk_Triedy na atribút *ID*.

- **Datum_nastupu** – dátum nástupu žiaka na školu. Korektný tvar je „Rok/Polrok“, napr. 2008/1.
- **KC** – číslo katalógu. Je to poradové číslo žiaka v katalógu, ktoré sa môže líšiť od poradového čísla pri utriedení podľa celého mena. Je to celé číslo, ktoré musí byť v rámci každej triedy unikátne.
- **Rodisko** – miesto narodenia žiaka. Typ tohto atribútu je reťazec maximálnej dĺžky 15.
- **Okres** – okres v ktorom leží miesto narodenia. Keďže rodisko neurčuje jednoznačne okres, musel byť do tejto tabuľky pridaný aj tento atribút. Okres je reťazec premenlivej dĺžky, maximálne však 15 znakov. Je to cudzí kľúč odkazujúci do tabuľky Sk_Okresy na atribút Okres, upravovanie sa prejaví kaskádovito.
- **Narodnost** – národnosť žiaka. Je to neprázdny reťazec maximálnej dĺžky 15.
- **Stat_prisl** – štátna príslušnosť. Je to dvojznakový reťazec, napr. „SR“ alebo „ČR“.

Definícia tabuľky

```
CREATE TABLE Sk_Ziaci (  
    RC CHAR(11)  
        CONSTRAINT PK_RC PRIMARY KEY  
    Meno VARCHAR(15) NOT NULL,  
    Priezvisko VARCHAR(15) NOT NULL,  
    Trieda CHAR(2) NOT NULL  
        CON FK_Z_Trieda FOREIGN KEY REFERENCES Sk_Triedy(ID)  
        ON UPDATE CASCADE,  
    Datum_Nastupu CHAR(6) NOT NULL  
        CONSTRAINT Valid_Nastup CHECK(...),  
    KC INT NOT NULL,  
    Rodisko VARCHAR(15) NOT NULL,  
    Okres VARCHAR(15) NOT NULL  
        CON FK_Z_Okres FOREIGN KEY REFERENCES  
        Sk_Okresy(Okres)  
        ON UPDATE CASCADE,  
    Narodnost VARCHAR(15) NOT NULL,  
    Stat_prisl CHAR(2) NOT NULL );
```

Indexy

Tabuľka obsahuje 2 atribúty, ktoré sú cudzími kľúčmi. Pre oba aplikácia vytvára indexy. Sú to atribúty *Trieda* a *Okres*. Index sa vytvára aj pre dvojicu (*Priezvisko*, *Meno*), ktorá je často použitá pre utriedenie výsledkov dotazu.

3.8. Tabuľka Sk_Znamky

Popis

Tabuľka obsahuje informácie o predmetoch, ktoré majú žiaci zvolené a známkach, ktoré im týchto predmetov boli uzavreté. Primárnym kľúčom tabuľky je štvorica (rodné číslo žiaka, kód predmetu, rok, polrok). Keď sa žiakovi zvolia predmety, ktoré bude navštevovať, zapíšu sa do tabuľky Sk_Znamky s „prázdnu“ známku. Na konci roku sa už len doplnia príslušné známky.

Zoznam atribútov

- **RC** – rodné číslo žiaka. Je to cudzí kľúč odkazujúci sa na atribút RC tabuľky Sk_Ziaci.
- **Kod** – kód predmetu. Atribút je cudzí kľúč odkazujúci na atribút Kod tabuľky Sk_Predmety
- **Polrok** – polrok, v ktorom žiak predmet navštevuje. Je to hodnota typu integer, ktorá nadobúda hodnoty 1 pre prvý polrok a 2 pre polrok druhý.
- **Rok** – rok, v ktorom žiak predmet navštevuje. Rok je nezáporné celé číslo, ktoré musí ležať v intervale <2000,3000>.
- **Znamka** – známka, ktorá bola žiakovi z predmetu uzavretá a ktorá sa objaví na vysvedčení. Má tvar jedného znaku. Povolené hodnoty závisia na type predmetu (atribút Sk_Predmety(Typ)). Keď má predmet Typ=1 znamená to, že sa jedná o voliteľný predmet. Ak má predmet Typ=2, je to predmet s povinnou dochádzkou a je hodnotený známkami 'A' alebo 'N' (absolvoval/neabsolvoval) a ak je Typ=3 ide o povinný známkovaný predmet, pričom známka je od '1' do '5'. Pre Typ=4 je to interval <'1',..., '4'>. Každý predmet ešte môže byť ospravedlnený, vtedy mu je pridelená známka 'O'. Korektnosť hodnoty sa overuje na aplikačnej úrovni.

Definícia tabuľky

```
CREATE TABLE Sk_Znamky (
  RC CHAR(11)
  CONSTRAINT FK_Z_RC FOREIGN KEY REFERENCES
  Sk_Ziaci(RC)
  ON UPDATE CASCADE,
  Kod CHAR(5)
  CON FK_P_Kod FOREIGN KEY REFERENCES Sk_Predmety(Kod)
  ON UPDATE CASCADE,
  Polrok INT
  CONSTRAINT Valid_PR CHECK (Polrok LIKE '[1-2]'),
  Rok INT
  CONSTRAINT Valid_Rok CHECK (Rok>=2000 AND Rok<=3000)
  CONSTRAINT PK_Znamky PRIMARY KEY
```

```
(RC, Kod, Polrok, Rok),  
Znamka CHAR(1));
```

Indexy

Tabuľka má 2 indexy pre cudzie kľúče: *RC* a *Kod*. V dôsledku malého počtu možných hodnôt atribútu *Znamka* sa predpokladá vysoká selektivita pri dotazoch na tento atribút. Napriek tomu index vytváraní nebude, keďže *Znamka* nie je používaná v podmienkovej klauzule.

3.9. Tabuľka Sk_Logy

Popis

Tabuľka slúži na evidenciu niektorých akcií vykonávaných užívateľmi aplikácie a slúži ako informačný zdroj o „pohybe“ v aplikácii. Medzi inými sa eviduje prihlasovanie a odhlasovanie užívateľov, čím sa zabráni súčasnému prístupu z dvoch rozličných miest pomocou jedného užívateľského účtu. Primárny kľúč tabuľky tvorí dvojica (login užívateľa, presný čas operácie).

Zoznam atribútov

- **ULogin** – užívateľské meno učiteľa. Je to cudzí kľúč odkazujúci na atribút *ULogin* tabuľky *Sk_Ucitelia*.
- **LTime** – presná časová známka operácie. Tento atribút je typu *DATETIME* a informuje o čase, v ktorom bola príslušná operácia vykonaná.
- **Oper** – popisuje typ operácie. Hodnota atribútu je 3-znakový reťazec, ktorý jasne popisuje typ operácie, ktorá bola vykonaná. Definovaných je niekoľko typov operácií. „LOC“ odpovedá operácii zamykania alebo odomykania triedy pri zadávaní známok. „LIN“ a „LOU“ sú príznaky pre logy súvisiace prihlasovaním a odhlasovaním. Hodnoty „USE“ a „SET“ odpovedajú úprave užívateľov a nastavení. Správnosť hodnoty kontroluje obmedzenie *Valid_Op*.
- **LText** – text logu. Stručný text, ktorý popisuje akciu a údaje, ktorých sa akcia týka. Veľkosť reťazca je obmedzená na 150 znakov.

Definícia tabuľky

```
CREATE TABLE Sk_Logy (  
    ULogin VARCHAR(10) NOT NULL  
        CONSTRAINT FK_L_Login REFERENCES  
        Sk_Ucitelia(ULogin),  
    LTime DATETIME NOT NULL  
        CONSTRAINT PK_Logy PRIMARY KEY (ULogin,LTime),  
    Oper CHAR(3) NOT NULL  
        CONSTRAINT Valid_Op CHECK (Oper IN
```

```
( 'LOC', 'LIN', 'LOU' .. ) );  
LText VARCHAR(150));
```

Indexy

Index sa vytvára pre atribút *Oper* z dôvodu vysokej selektivity tohto stĺpca. Druhým indexom je index pre cudzí kľuč *ULogin* odkazujúci sa na *Sk_Ucitelia(Login)*.

4. Vedľajšie okná aplikácie

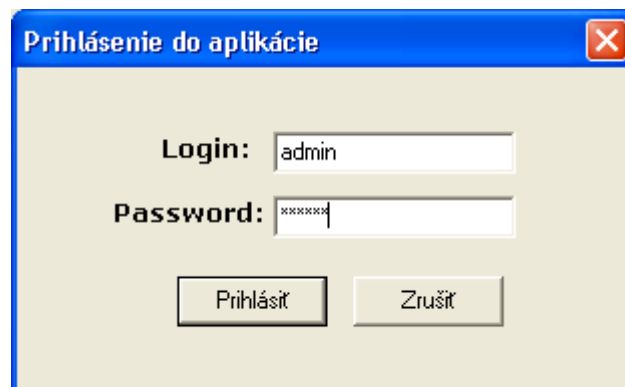
4.1. Okno pre prihlásenie

Funkcia

Toto dialógové okno slúži na prihlásenie užívateľa do aplikácie. Objavuje sa pri spustení aplikácie a bez zadania správneho užívateľského mena a hesla nebudú povolené skoro žiadne funkčnosti aplikácie. Po jeho zavretí je možné ho opätovne vyvolať z položky *Súbor* menu hlavného okna kliknutím na *Login*. Ak iný užívateľ stále prihlásený, treba ho najskôr odhlásiť použitím voľby *Logout*. Stav prihlásenia ako aj prihlásenú osobu ľahko zistíme z hlavičky hlavného okna.

Vlastnosti

Pri vytváraní okna sa súčasne overuje funkčnosť pripojenia k databázovému serveru a taktiež existencia potrebných databázových objektov. Oboje sú nutné podmienky pre používanie aplikácie. Na testovanie funkčnosti pripojenia som vytvoril funkciu *CheckConnect*, ktorá overí existenciu servera, platnosť nastaveného účtu a vracia hodnotu typu boolean o úspešnosti tejto akcie. Potom sa zisťuje existencia kostry našej databázovej aplikácie. Kostru som definoval ako množinu objektov potrebných k fungovaniu aplikácie, teda tabuliek, funkcií, procedúr a triggerov. K tomuto účelu som vytvoril procedúru *TestDatabase*, ktorá za pomoci informácií z tabuliek *INFORMATION_SCHEMA.tables* a *sys.objects*⁶ a overuje prítomnosť požadovaných databázových objektov.

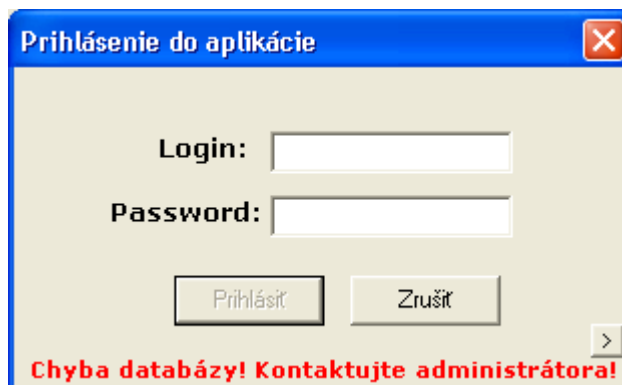


obr.č. 2 – Prihlasovacie okno pri funkčnom spojení

Ak všetko funguje, aplikácia vytvorí spojenie s databázovým serverom a sprístupní tlačidlo *Login*. Po zadaní prihlasovacieho mena (*Login*) a hesla (*Password*) aplikácia spustí skript, ktorým overí, či užívateľ existuje a porovná zadané zahashované heslo s tým v databáze. Na hashovanie sa používa

⁶ Jedná sa o systémové tabuľky, ktoré uchovávajú informácie o existujúcich tabuľkách, funkciách, procedúrach atď.

hashovací algoritmus MD5⁷. Ak je heslo správne, spustí sa hlavné okno aplikácie a sprístupnia sa ovládacie prvky náležiacie užívateľovi v závislosti na jeho právach.



obr.č. 3 – Prihlasovacie okno pri poruche spojenia

Ak spojenie nefunguje, alebo je databáza nekonzistentná, bude tlačidlo *Login* nefunkčné a súčasne bude viditeľné upozornenie o porušení funkčnosti spojenia. Správne nastavenie pripojenia k databázovému serveru ako aj obnovenie konzistencie databázy je možné vykonať v okne *Nastavenia*.

4.2. Nastavenia

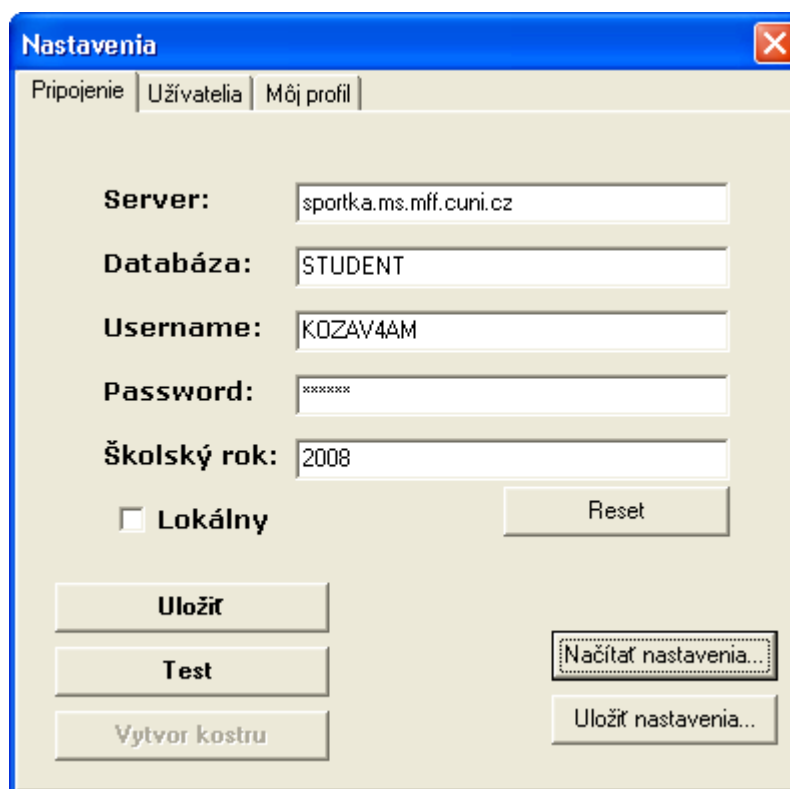
Funkcia

Dialógové okno, ktoré slúži na nastavenie vlastností aplikácie a správu užívateľských účtov. Okno obsahuje viacero záložiek, ktoré sú sprístupnené v závislosti na užívateľovi. Administrátor aplikácie má prístupné všetky záložky nastavení, kým bežný užívateľ má povolenú len záložku *Môj profil*. Okno je možné vyvolať z položky *Súbor* v menu hlavného okna vybratím voľby *Nastavenia*. Pri nefunkčnosti spojenia ho možno spustiť priamo tlačidlom pri upozornení o narušení spojenia v prihlasovacom okne. V tom prípade bude prístupná iba záložka na nastavenie spojenia.

Záložka Pripojenie

Záložka *Pripojenie* umožňuje nastaviť spojenie s databázovým serverom. Zaškrŕavacie políčko *Local* určuje, či sa aplikácia pripája k lokálnemu serveru, alebo vzdialenému serveru. Pole *Server* odpovedá názvu servera, pole *Databáza* určuje názov databázy, nad ktorou má aplikácia pracovať. V kolónkach *Username* a *Password* sa nastaví existujúci užívateľ na tomto serveri. Textové pole *Školský rok* slúži na nastavenie aktuálneho školského roku. Tlačidlo *Uložiť* uloží súčasné nastavenia a pokúsi sa o opätovné prihlásenie.

⁷ V kryptografii je MD5 (Message-Digest algorithm 5) široko používaná kryptografická hash funkcia s 128-bitovou hodnotou hashu. MD5 bola navrhnutá Ronaldom Rivestom v roku 1991.



obr.č. 4 – Nastavenie pripojenia

Na testovanie funkčnosti pripojenia a kontrolu existencie kostry databázy slúži tlačidlo *Test*. Testovanie pripojenia sa vykonáva rovnako ako pri štarte aplikácie, ktoré je popísané vo vlastnostiach dialógového okna v odseku 4.1.

Ak je pripojenie k databáze funkčné, ale kostra databázy nie je úplná, sprístupní sa tlačidlo *Vytvor kostru*, ktoré kompletne obnoví kostru databázy. Obnovia sa definície všetkých funkcií a procedúr potrebných na bezchybný chod aplikácie. Pri tom však dôjde aj k opätovnému vytvoreniu všetkých tabuliek, takže prípadné existujúce dáta budú stratené. Z programátorského hľadiska je táto akcia riešená tak, že je volaná procedúra *LoadMyScript*, ktorá má ako parameter skriptový súbor a jednotlivé skripty odosiela na databázový server. Skriptové súbory pre obnovenie kostry sú *schema_create.sql* a *schema_initfunc.sql* a nachádzajú sa spolu s ostatnými SQL skriptami v adresári *scripts*.

Záložka Užívatelia

Táto záložka slúži na správu užívateľov aplikácie. Záložka obsahuje prehľad všetkých existujúcich užívateľov, umožňuje pridávanie nových užívateľov, ich upravovanie a mazanie. Záložka je prístupná iba správcovi aplikácie, teda užívateľovi „admin“.

Po kliknutí na tlačidlo *Pridaj*, ktoré slúži na pridávanie užívateľov, sa objavia prázdne textové polia, do ktorých administrátor vyplní príslušné údaje a nastaví práva nového užívateľa. Práva na prístup k jednotlivým záložkám hlavného okna aplikácie sa nastavujú zaškrtnutím príslušných

*CheckBoxov*⁸. Objavia sa tiež tlačidlá *OK* a *Zrušiť* na potvrdenie prípadne zrušenie zmien. Pri stlačení tlačidla *Uprav*, slúžiaceho na úpravu aktuálneho užívateľa, je situácia obdobná – sprístupnia sa textové polia a umožní sa úprava súčasných dát. Po potvrdení operácie aplikácia skontroluje korektnosť zadaných/upravených dát. Následne vygeneruje a spustí príslušný SQL dotaz. Pri tom používa databázové procedúry *Sk#_UcitelAdd* a *Sk#_UcitelEdit*.

Na mazanie slúži tlačidlo *Zmaž*. Po jeho stlačení aplikácia overí, či chcete označeného užívateľa skutočne zmazať. Ak áno, skontroluje či užívateľ nie je triednym učiteľom v niektorej z tried, a potom vygeneruje príslušný príkaz na jeho odstránenie z databázy.

Záložka *Môj profil*

Záložka zobrazuje užívateľovi informácie o jeho profile a umožňuje mu zmeniť si heslo. Po stlačení tlačidla *Zmena hesla* sa užívateľovi objaví kolónka pre zadanie pôvodného hesla a 2 kolónky pre zadanie a overenie nového hesla. Ak je pôvodné heslo zadané správne a nové heslá sa zhodujú, dôjde po stlačení *OK* k vygenerovaniu SQL skriptu na zmenu súčasného hesla na nové.

4.3. Správa okresov

Funkcia

Okno na obsluhu okresov a krajov. Spustíme ho vybratím voľby *Správa okresov* z položky *Akcie* menu hlavného okna. Každému študentovi sa pri pridávaní alebo úprave nastavuje rodisko a okres, v ktorom bol narodený. Každý okres má práve jeden kraj. Problémom je, že miesto narodenia neurčuje jednoznačne okres, preto je v aplikácii definovaný iba vzťah okres - kraj.

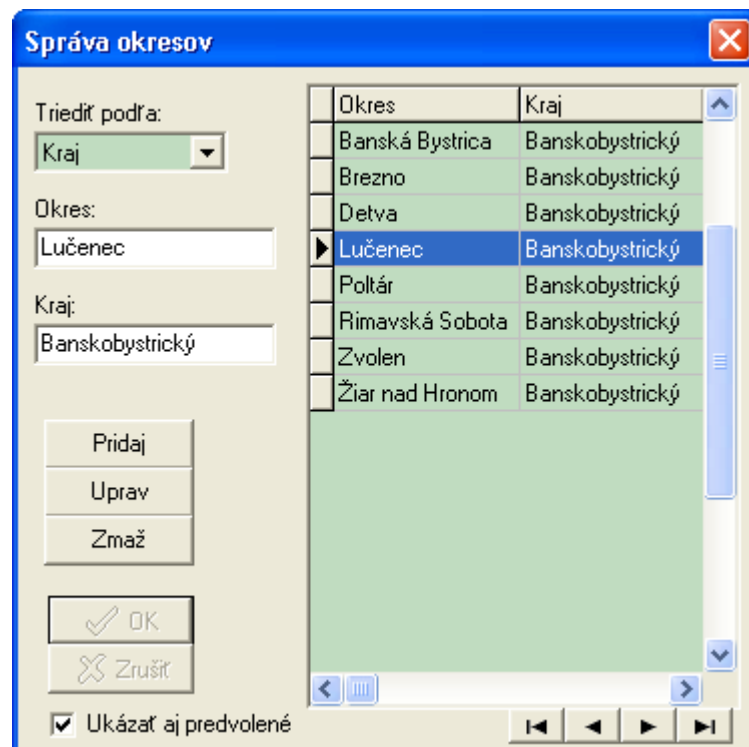
Každý riadok tabuľky s okresmi má okrem atribútov *okres* a *kraj* aj atribút *CFG*. To rozdeľuje okresy na 2 typy. Prvým typom sú okresy, ktoré sú pridané do aplikácie pri vytváraní kostry databázy pomocou konfiguračného skriptu. Druhý typ tvoria okresy pridané užívateľmi. Bežný užívateľ má prístup iba k druhej skupine, kým administrátor môže upravovať aj tie štandardné.

Vlastnosti

V pravej časti sa nachádza náhľad na okresy a ich kraje, ktorý zabezpečuje instancie komponenty *TDBGrid*. Vľavo sa nachádza roletové menu pre výber sa atribútu, podľa ktorého majú byť dáta utriedené. Pri zmene hodnoty tohto zoznamu sa zavolá jeho metóda *OnChange*, ktorej telo zabezpečí nové utriedenie. Pod týmto zoznamom sú textové polia, ktoré zobrazujú informácie o aktuálne označenom okrese. Nižšie sa nachádzajú tlačidlá na pridávanie, úpravu a mazanie okresov a tlačidlá pre potvrdenie

⁸ *CheckBox* je anglický výraz pre zaškrávané políčko, v Delphi sa trieda odpovedajúca zaškrávaciemu políčku volá *TCheckBox*.

a zrušenie týchto operácií. Administrátorovi sa navyše v ľavom dolnom rohu zobrazí checkbox, ktorého zaškrtnutie umožní upravovať okresy, ktoré boli do aplikácie pridané štandardne.



obr.č. 5 – Správa okresov

Pri použití tlačidla *Pridaj* alebo *Uprav* aplikácia zavolá procedúru *O_Lock* s vhodnými parametrami. Tá zobrazí a príslušne inicializuje textové polia pre pridanie nového alebo úpravu označeného okresu a zakáže používanie tých ovládacích prvkov, ktoré by túto operáciu narušovali. Súčasne obnoví funkciu tlačidiel *OK* a *Zrušiť* pre potvrdenie a zrušenie operácie. Spomínaná funkcia má 2 parametre typu boolean, pričom prvý určuje, či má prvky zakázať alebo obnoviť a druhý či má textové polia inicializovať pre pridávanie nového alebo úpravu existujúceho okresu. Po potvrdení operácie sa aplikácia pokúsi príslušné dáta pridať do databázy, pričom používa uložené databázové procedúry *Sk#_OkresyAdd* a *Sk#_OkresyEdit*. Ak pridanie alebo úprava zlyhá, chybové hlásenie DB servera aplikácia odchyti a zobrazí užívateľovi. Nakoniec sa zavolá opäť procedúra *O_Lock*, ktorej prvý parameter bude mať tento krát hodnotu „False“, čo má za následok inverzný efekt – potrebné ovládacie prvky sa sprístupnia. Mazanie možno vyvolať tlačidlom *Zmaž*, ktorého metóda *OnClick* skontroluje správnosť mazania a vyzve k potvrdeniu tejto operácie.

4.4. Zadávanie známok

Funkcia

Okno sa objaví pri požiadavke na zadávanie známok študentom istej triedy. Túto akciu možno vyvolať zo záložky *Známky* vybraním potrebnej

triedy a kliknutím na tlačidlo *Zadávať známky*. V návrhu bola stanovená požiadavka na ľahké, rýchle a flexibilné užívateľské rozhranie pre zadávanie známok. Z tohto dôvodu sa definoval pojem sady ako postupnosti predmetov v poradí, v akom sú uvedené v triednej knihe.

Vlastnosti

Okno má v hornej časti roletový zoznam žiakov triedy. V spodnej časti sa nachádzajú tlačidlá slúžiaci na posun medzi žiakmi. V pravom dolnom rohu je tlačidlo na uloženie zmien a tlačidlo na zatvorenie okna. V strednej časti sa nachádzajú kódy predmetov a k nim príslušné textové polia pre zadanie a upravovanie známky z predmetu. Tieto názvy predmetov a textové polia sú vytvárané dynamicky v závislosti na predmetoch v sade.

Subject Code	Grade	Subject Code	Grade
SLJ01	2	INFCV	2
ANJ01	3	GEOGR	1
NEJ01	2	BIOLG	2
FRJ01		DEJEP	2
MATEM	3	CHEMO	
INFOR	3	FYZIK	

obr.č. 6 – Zadávanie známok

Pri zvolení žiaka sa sprístupnia textové polia iba pre tie predmety, ktoré má žiak zapísané a vyplnia sa známky, ktoré už žiak má uzavreté. Označí sa prvé z týchto polí. Po zadaní znaku sa spustí procedúra *MyChange*, ktorá za pomoci funkcie *CheckTyp* zistí, či zadaný znak odpovedá korektnému hodnoteniu pre tento typ predmetu. Ak nie, znak je zamietnutý a aplikácia čaká kým nie je zadaná správna známka. Po zadaní korektnej hodnoty ukazovateľ preskočí automaticky na ďalšie textové pole pre zadanie známky. Po zadaní poslednej známky preskočí ukazovateľ na tlačidlo *Ďalší*.

Užívateľ teda iba stláča príslušné znaky a po zadaní poslednej známky prejde na nasledujúceho študenta stlačením klávesy Enter. Okrem toho predmety má usporiadané zhodne s triednou knihou a pre každého študenta sú prístupné iba predmety, ktoré skutočne navštevuje, takže zadávanie uzavretých známok je rýchle a praktické.

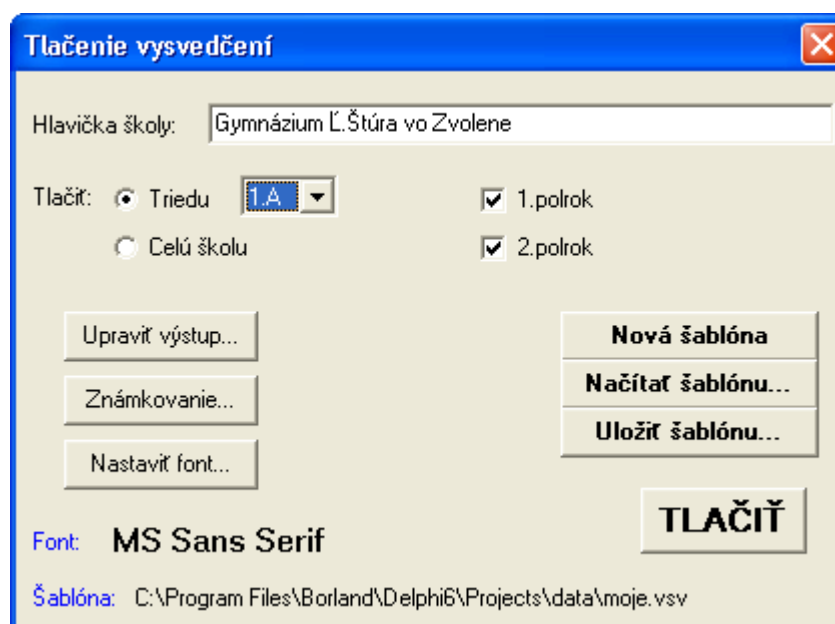
4.5. Tlačenie vysvedčení

Funkcia

Toto okno umožňuje užívateľom nastavovať vlastnosti tlače a formát výstupu na tlačiareň ako aj obsluhovať samotnú tlač vysvedčení. Pri tlačení sa využívajú šablóny. Šablónu definujeme ako kompletnú sadu nastavení potrebnú pre vytlačenie vysvedčenia. Uložená je v súbore a obsahuje nastavenie výstupu, nastavenie slovného pomenovania známok, nastavenie typu písma a hlavičky školy.

Vlastnosti

V hornej časti okna sa nachádza textové pole pre nastavenie hlavičky školy. Pod ňou je skupina ovládacích prvkov pre výber triedy, ktorú chce užívateľ tlačiť a pre nastavenie polrokov, z ktorých majú byť tlačené známky. Aspoň jeden z nich musí byť zaškrtnutý. Pod nimi sa na ľavej strane nachádzajú 3 tlačidlá pre nastavenie výstupu, nastavenie slovného popisu hodnotenia a typu písma. Na spodku okna je ukážka zvoleného písma a názov aktuálne zvolenej šablóny. V pravej spodnej časti sa nachádza trojica tlačidiel pre obsluhu šablón.



obr.č. 7 – Tlač vysvedčení

Pri nastavovaní tlače si užívateľ vytvorí vlastnú šablónu, prípadne otvorí existujúcu šablónu a vykoná potrebné zmeny. Ak zvolí možnosť *Nová šablóna*, aplikácia vytvorí novú šablónu, ktorá si vyžaduje prejsť a nastaviť výstup, typ písma aj slovné ekvivalenty známok. Ak stlačí *Načítať šablónu*, zavolá sa funkcia *Execute* dialógového okna pre výber otváraného súboru *OpenDialogSab*, ktorá vracia hodnotu typu boolean. Tá určuje, či užívateľ zvolil nejaký súbor. Dialógové okno umožní vybrať súbor iba so správnou príponou. V kladnom prípade aplikácia zavolá funkciu *LoadSabFromFile*, ktorá má ako parameter zvolený súbor a načíta všetky nastavenia šablóny.

Ak chce užívateľ zmeniť typ písma, použije tlačidlo *Nastaviť font*. Toto tlačidlo spustí dialógové okno pre výber typu písma, teda instanciu komponenty *TFontDialog*. Ak chce zmeniť formát výstupu na tlačiareň, musí použiť tlačidlo *Upraviť výstup*, ktoré zabezpečí otvorenie okna pre nastavenie formátu vysvedčenia. Viac o úprave výstupu sa nachádza v popise okna *Formát vysvedčenia*. Pre nastavenie slovného popisu známk stlačte tlačidlo *Známkovanie*. Opäť sa otvorí samostatné okno, ktorého funkcie a popis nájdete v kapitole 4.7. Po ukončení úprav môžete šablónu uložiť použitím príslušného tlačidla a vybratím názvu súboru, kam majú byť nastavenia uložené.

Samotnú tlač možno spustiť kliknutím na *Tlačiť*. Otvorí sa dialógové okno pre nastavenie tlače. Po výbere tlačiarne a zahájení tlače sa spustí samostatné vlákno, ktoré slúži na prerušenie tlače. Kedykoľvek počas tlače možno tlačenie prerušiť a aplikácia prestane odosielať dáta na tlačiareň.

4.6. Formát vysvedčenia

Funkcia

Okno možno vyvolať kliknutím na tlačidlo *Upraviť výstup*, ktoré sa nachádza v okne pre nastavenie tlače. Slúži ako grafické rozhranie pre nastavenie výstupu na tlačiareň. Nastavujú sa polohy, rozmiestnenie a obsah jednotlivých kolóniek v závislosti na tom, aký tvar má tlačivo pre vysvedčenie. Hlavnú časť okna zaberá biely panel, ktorý znázorňuje tlačivo pre vysvedčenie. Na tejto ploche sa nachádzajú 2 typy objektov⁹: údajové a predmetové.

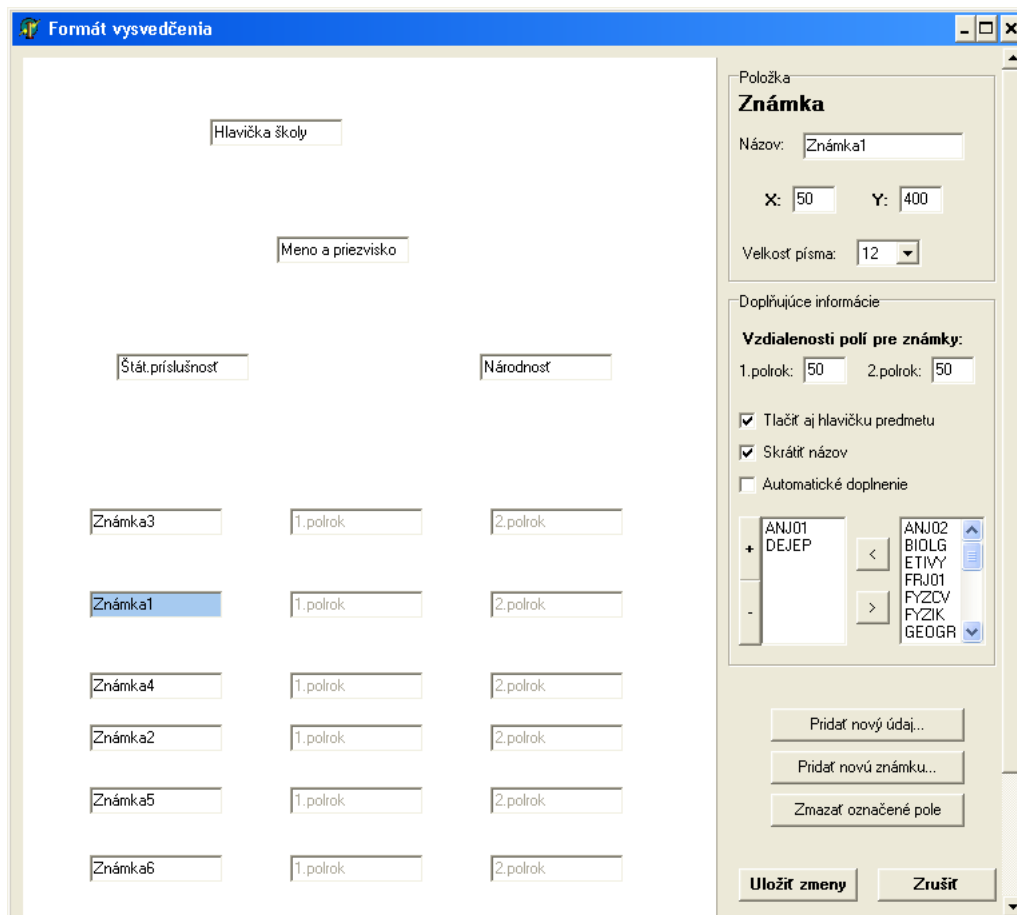
Vlastnosti

Údajový objekt určuje vlastnosti a umiestnenie kolónky, do ktorej bude pri tlačení doplnený údaj, zvolený z roletového zoznamu na ovládacom paneli. Ďalšie parametre údajového objektu sú veľkosť písma, ktorou bude vytlačený a súradnice na ploche.

Predmetový objekt určuje umiestnenie 3 kolóniek. Prvá kolónka určuje, kde bude vytlačený názov predmetu, druhá a tretia určujú kde bude vytlačená známka z prvého a druhého polroku. Parametre sa nastavujú na ovládacom paneli. *Názov* je dobrovoľný parameter a slúži pre pomocné účely. Nižšie sa nachádzajú polia pre zadanie súradníc a veľkosti písma. Ďalej nájdeme polia pre nastavenie vzdialenosti medzi jednotlivými kolónkami a 3 zaškrtnuté políčka. Prvé z nich určuje, či sa má tlačiť aj názov predmetu, čo nie je vhodné ak je názov predmetu už na tlačive. Druhé políčko určuje, či má byť hlavička predmetu vytlačená celá, alebo iba jeho prvá časť. To sa využije v prípade, že tlačivo obsahuje príponu a treba doplniť iba prvú časť názvu. Posledné určuje, či sa v prípade neuspokojenia požiadavky na vytlačenie niektorého z predmetov má doplniť najbližší voľný predmet. Na spodnej časti

⁹ V tomto kontexte používame slovo „objekt“ na popis prvkov umiestnených na plochu znázorňujúcu tlačivo pre vysvedčenie.

ovládacieho panelu sa nachádzajú 2 zoznamové polia pre nastavenie predmetov a ich priority pri tlači. Ľavý zoznam určuje, ktoré predmety a v akom poradí má program pri vyhodnocovaní doplniť. Z pravého zoznamu možno za pomoci tlačidiel so symbolmi šípok presúvať predmety a použitím tlačidiel so symbolmi plus a mínus predmety zoradiť.



obr.č. 8 – Nastavenie formátu vysvedčenia

Objektu odpovedá v aplikácii instancia vlastného typu *Kolónka*, ktorý obsahuje všetky potrebné položky. Aplikácia si udržuje pole týchto objektov a pri úprave pracuje z jeho kópiou. Pri uložení zmien sa zmeny z kópie prenású do pôvodného poľa. V prípade načítania šablóny sa nastavenia objektov načítajú zo súboru do poľa, pri ukladaní šablóny sa zase naopak ukladajú z pamäti do súboru.

Kliknutím myšou na kolónku objektu ho označíme, čo aplikácia znázorní tým, že zmení jeho farbu na modro. Vpravo sa nachádza ovládací panel, ktorý zobrazuje a umožňuje meniť údaje aktuálne označeného objektu. Klikaním na panel posúvame označený objekt na zvolené súradnice, čo zabezpečuje telo metódy *OnMouseDown* tohto panelu. Pozíciu možno tiež meniť zadávaním súradníc. Textové polia pre súradnice pri zmene kontrolujú správnosť hodnoty a posúvajú objektom.

Pri stlačení tlačidla *Pridať nový údaj* alebo *Pridať nový predmet* sa v ľavom hornom rohu objaví nový objekt príslušného typu. Na ovládacom paneli sa je možné nastaviť vlastnosti. Objekt treba posunúť na správne miesto.

4.7. Známkovanie

Funkcia

Toto modálne okno slúži na nastavenie slovného popisu hodnotenia. Slovný popis známky je text, ktorý sa zobrazí na vysvedčení v kolónke pre hodnotenie, ak žiak obdržal túto známku. Napríklad pre známku „1“ je tento text „výborný“. Na školách sa používa odlišná stupnica a slovný popis pre známky zo správania a z bežných predmetov.

Vlastnosti

Pri vytváraní okna sa vytvoria textové polia pre zadanie slovných popisov a inicializujú sa hodnotami aktuálnej šablóny. Užívateľ môže tieto popisy upravovať, stlačením tlačidla *Uložiť* sa nastavia zadané popisy do aktuálnej šablóny.

5. Hlavné okno aplikácie

5.1. Popis

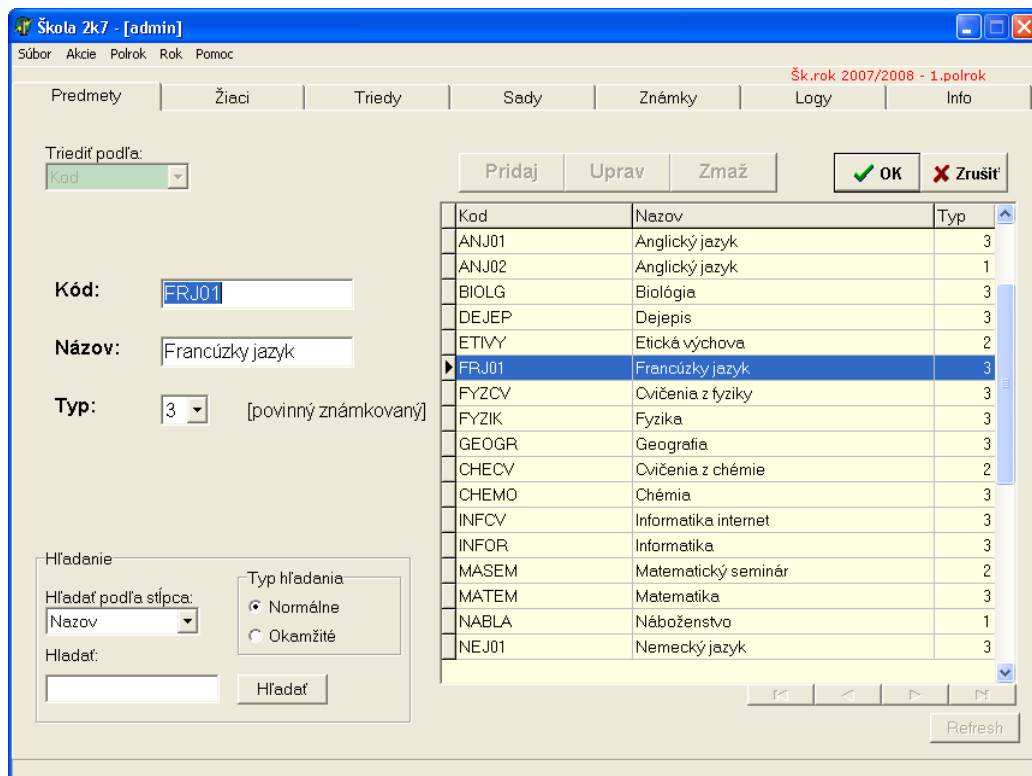
Hlavné okno aplikácie obsahuje 7 záložiek: Predmety, Žiaci, Triedy, Sady, Znamky, Logy a Info. Jednotlivé záložky sú prístupné na základe práv aktuálne prihláseného užívateľa. Záložka *Info* obsahuje základné informácie o programe, a je prístupná aj bez prihlásenia do aplikácie. Ostatné záložky slúžia na obsluhu dát, pričom rozloženie ovládacích prvkov a ich vlastností je pre každú z nich takmer identické. Každá táto záložka obsahuje náhľad na príslušné dáta (komponenta *DBGrid*), výber položky, podľa ktorej má byť náhľad utriedený (komponenta *ComboBox*) a textové polia pre zobrazenie údajov aktuálne označeného záznamu. Ovládacie prvky na vyhľadávanie sú umiestnené na každej záložke, kde to má nejaký význam, pričom je možné použiť vyhľadávanie na úplnú zhodu (Normálne) alebo priebežné vyhľadávanie na aktuálnu zhodu (Okamžité). Jednotlivé záložky presnejšie popisujú nasledujúce kapitoly.

5.2. Záložka Predmety

Popis

Záložka slúži na pridávanie, úpravu a mazanie predmetov. Pracuje s tabuľkou `Sk_Predmety`.

Obrázok



obr.č. 9 – Záložka pre správu predmetov

Ovládacie prvky a funkcie

V ľavej hornej časti záložky sa nachádza výberový zoznam *P_ComboBoxOrder*, ktorý slúži na výber atribútu, podľa ktorého sa dáta triedia. Triediť možno podľa atribútov *Kod* a *Nazov*. Pri zmene položky sa spustí metóda *OnChange*, ktorá zoradí dáta podľa zvolenej hodnoty.

V pravej časti sa nachádzajú tlačidlá *Pridaj*, *Uprav* a *Zmaž*. Po kliknutí na tlačidlo *Pridaj* alebo *Uprav* sa spustí procedúra *P_Lock*, ktorá na základe parametrov sprístupní alebo zakáže ovládacie prvky, textové polia pre zadanie/upravenie údajov a tlačidlo na potvrdenie prípadne zrušenie zmien. Pri potvrdení zmien sa zavolá databázová funkcia *Sk#_PredmetAdd*, resp. *Sk#_PredmetEdit*, ktoré overia korektnosť dát, a pridajú hodnoty do databázy. V prípade chyby zachytí aplikácia chybové hlásenie SQL servera a vypíše ho pomocou funkcie *MessageDlg*, slúžiacej na zobrazenie užívateľských hlásení. Po kliknutí na *Zmaž* aplikácia overí, že užívateľ s operáciou súhlasí, a korektné odstráni príslušné dáta vrátane závislostí.

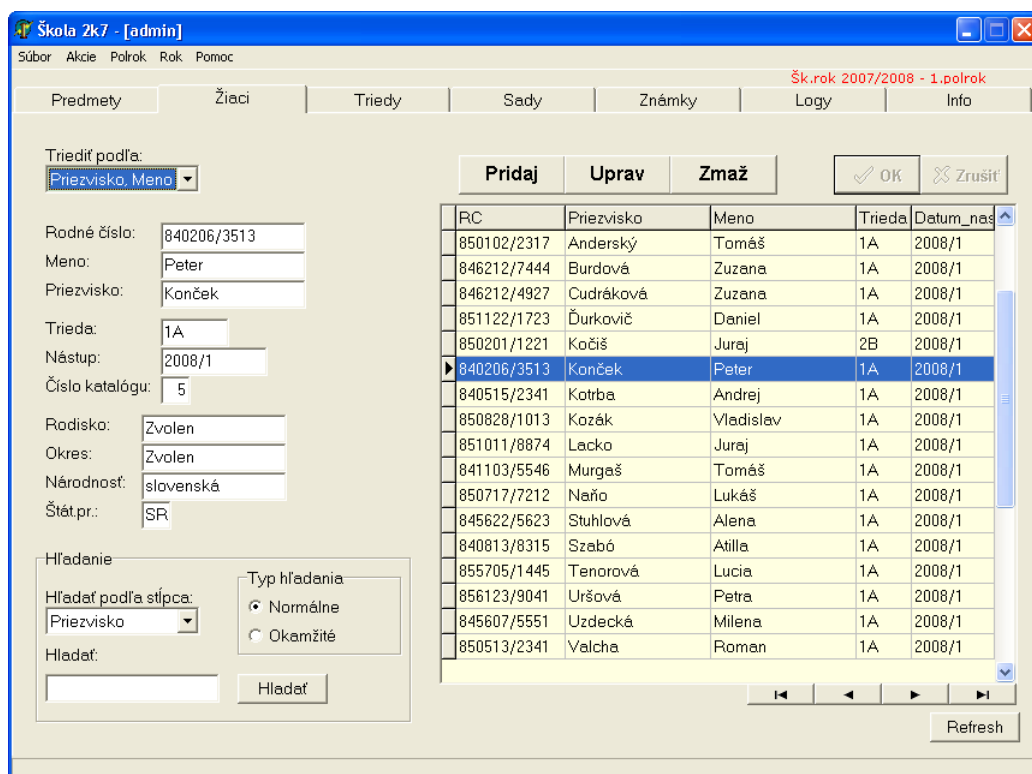
V ľavom dolnom rohu sa nachádza skupina prvkov slúžiacich na vyhľadávanie. Pri okamžitom hľadaní sa zamknú ovládacie prvky, ktoré by jeho funkciu logicky narušovali. V pravom dolnom rohu sa nachádza tlačidlo *Refresh*, ktoré pri stlačení obnoví platnosť dát. Náhľad na predmety sprostredkuje použitie komponenty *DBGrid*, ktorá je zviazaná s instanciou komponenty pre odoslanie dotazu *P_MSQuery*. Tá použitím metódy *Open* získava potrebné dáta a *P_DBGrid1* zobrazuje výsledok dotazu.

5.3. Záložka Žiaci

Popis

Táto záložka slúži na spravovanie údajov o žiakoch na škole. Užívateľ s príslušnými právami tu môže pridávať žiakov, upravovať ich údaje, nastavovať a upravovať predmety, ktoré žiak aktuálny rok navštevuje. Pracuje s tabuľkami Sk_Ziaci a Sk_Znamky.

Obrázok



obr.č. 10 – Záložka Žiaci

Ovládacie prvky a funkcie

Na výber triediaceho atribútu je použitá komponenta *ComboBox* (*Z_ComboboxOrder*), triediť možno podľa studentovho celého mena, jeho rodného čísla alebo triedy. Metóda *Z_ComboboxOrder.OnChange* zareaguje na zmenu triediaceho atribútu a vhodne upraví dotaz, ktorý následne získa správne zatriedené údaje. Informácie o študentoch získava z databázového serveru instanciou triedy *TMSQuery* s názvom *Z_MSQuery*, pričom informácie sa zobrazujú jednak v prehľadnej tabuľke *Z_DBGrid* a zároveň sú na zobrazenie jednotlivých údajov aktuálne označeného študenta použité *DBEdit*-y. Meniť aktuálne zvoleného študenta možno klikaním na *DBGrid*, alebo použitím *DBNavigatoru*, ktorý sa nachádza priamo pod náhľadom na študentov.

Tlačidlá *Pridaj*, *Uprav* a *Zmaž* slúžia na pridávanie žiakov, úpravu žiakových údajov a mazanie žiakov školy. Po stlačení *Zmaž* aplikácia zobrazí dialógové okno s potvrdením operácie a pri pozitívnej odpovedi vytvorí a

spustí SQL dotaz *QueryDel*, ktorý odstráni žiaka vrátane závislostí. Pri výbere *Pridaj* alebo *Uprav* najskôr aplikácia volaním procedúry *Z_Lock* zakáže niektoré ovládacie prvky, zobrazí tlačidlá pre potvrdenie/zrušenie operácie a sprístupní textové polia pre zadávanie/úpravu, ktoré sú vhodne inicializované. Zobrazí sa tiež *ListBox*¹⁰, ktorý slúži na nastavenie predmetov študenta pre aktuálny školský rok. Po stlačení tlačidla *Reset* sa v tomto výberovom zozname nachádzajú všetky predmety sady, ktorú má zvolenú jeho trieda. Stláčaním písmena "d" možno vyhadzovať označený predmet a tým odstraňovať z ponuky tie, ktoré mu nepatria. Po potvrdení operácie sa SQL dotazom zavolá databázová procedúra *Sk#_ZiakAdd* resp. *Sk#_ZiakEdit* s vhodnými parametrami. Táto procedúra skontroluje správnosť zadaných údajov. Prípadné chybové hlásenia aplikácia zachytí a zobrazí užívateľovi pomocou *MessageDlg*. Ak sú zadané údaje korektné, procedúra pridá študenta resp. upraví jeho údaje a zapíše jeho predmety do tabuľky *Sk_Znamky*. Skupina ovládacích prvkov pre vyhľadávanie sa správa identicky s vyhľadávaním u predmetov, preto sa nimi nebudem ďalej zaoberať.

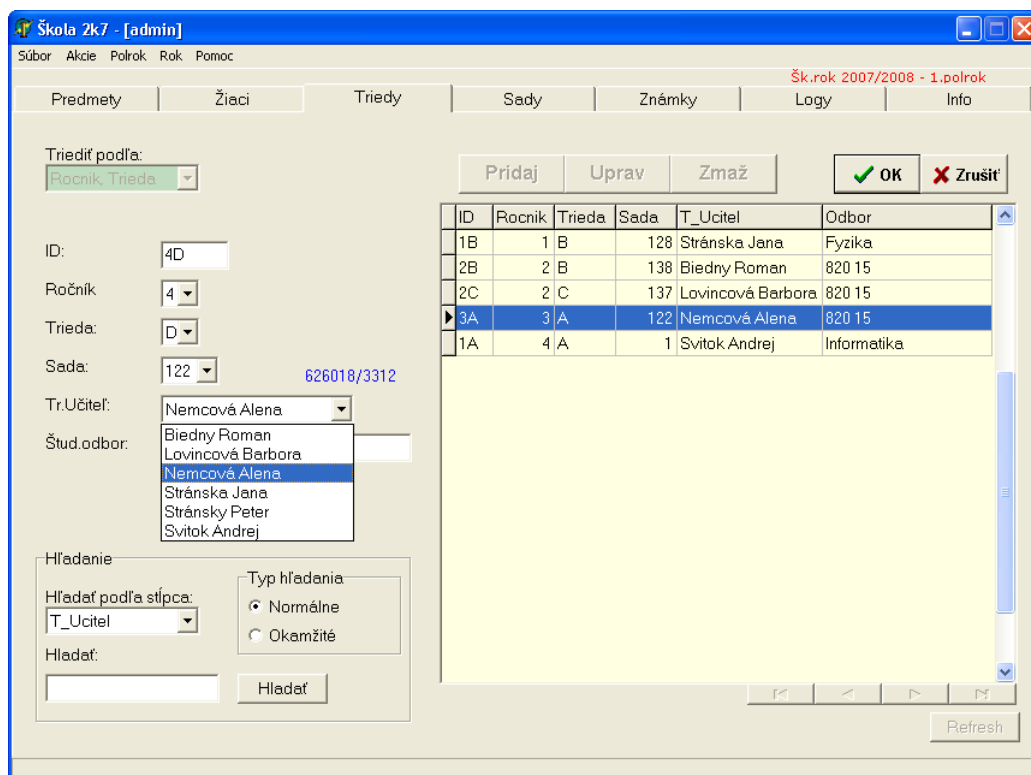
5.4. Záložka Triedy

Popis

Záložka pracuje s triedami. Každá trieda má práve jedného triedneho učiteľa a práve jednu sadu. Počet študentov v triede je obmedzený na 50 žiakov, čo je zrejme dostatočná rezerva aj pre extrémne „preľudnené“ triedy. Trieda je zadaná dvojznakovým identifikátorom. Záložka pracuje s tabuľkou *Sk_Triedy*.

¹⁰ Komponenta *TListBox* je zoznam, do ktorého možno pridávať prvky použitím metódy *TListBox.Items.Add*.

Obrázok



obr.č. 11 – Záložka Triedy

Ovládacie prvky a funkcie

Triedenie je riešené analogicky s predošlými prípadmi, triediace atribúty sú v tomto prípade *ID* triedy alebo celé meno triedneho profesora. Pridávanie a upravovanie triedy je zľahčené výberovými zoznamami pre triedneho učiteľa, ročník a triedu. Na jednej strane by bolo priam otrocké prípadné zadávanie odkazu na triedneho učiteľa rodným číslom, na strane druhej tým máme zabezpečenú korektnosť dát, keďže výberové zoznamy sú typu *DropDownList*¹¹ a sú naplnené výhradne korektnými dátami. Dôležité je uvedomiť si fakt, že „Priezvisko, Meno“ nie je jednoznačná identifikácia osoby a teda ani triedneho učiteľa. Preto aplikácia zobrazuje pomocou *T_LabelRC* doplnkovú informáciu o rodnom čísle vybranej osoby. Údaje z databázového serveru získavame opäť pomocou SQL dotazu, konkrétne vhodným nastavením parametru *T_MSQuery.SQL.Text* a zavolaním metódy *Open()* komponenty *TMSQuery*. Pri mazaní a potvrdení operácie aplikácia overí, či zmazaniu nebráni nejaká závislosť a následne spustí metódu *QueryDel.Execute*, ktorá zvolenú triedu odstráni. Zmazanie triedy v ktorej sú žiaci považujem za veľmi násilné, preto aplikácia pred zmazaním triedy vyžaduje zmazanie jej žiakov prípadne ich preradenie do inej triedy. Pridávanie a editovanie využíva vopred uložených databázových procedúr *Sk#_TriedyAdd* a *Sk#_TriedyEdit*, ktoré v prípade korektnosti zadávaných údajov triedu pridajú alebo upravujú, v prípade opačnom použijú funkciu

¹¹ Keď sa výberovému zoznamu (instancia triedy *TComboBox*) nastaví atribút *Style* na hodnotu *csDropDownList*, možno vyberať iba z prvkov zoznamu.

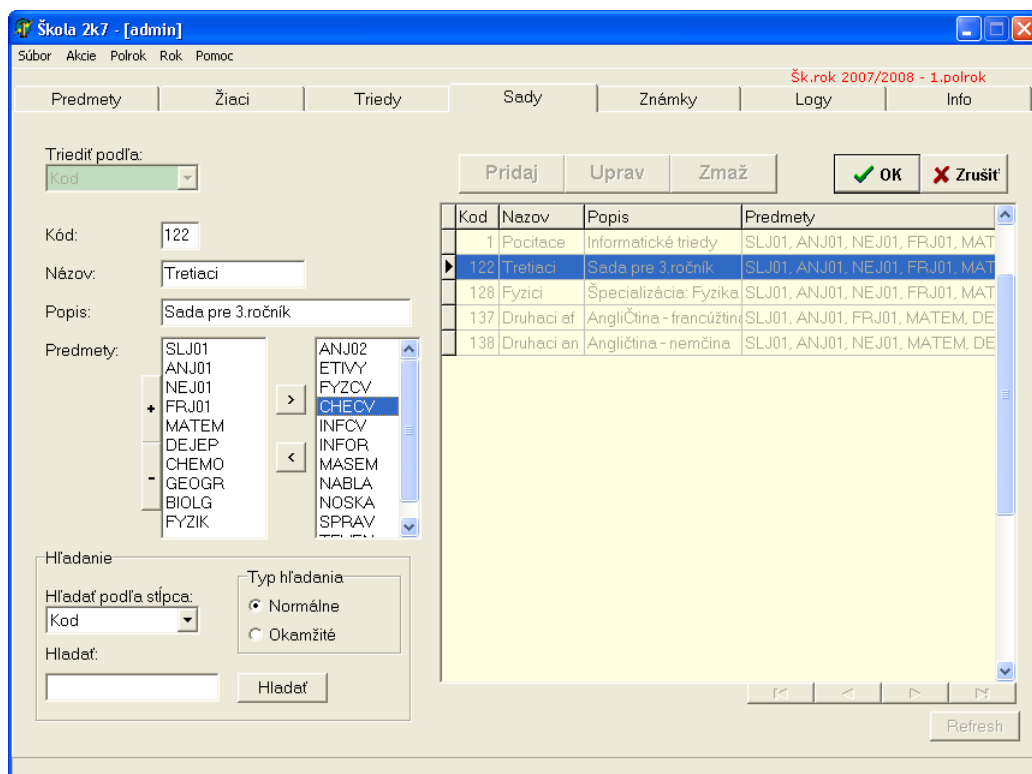
*RAISERROR*¹², ktorá slúži na výpis chybových hlásení. Toto hlásenie aplikácia zobrazí užívateľovi aplikácie.

5.5. Záložka Sady

Popis

Sadu triedy sme definovali ako postupnosť predmetov, ktorá kopíruje hlavičku triednej knihy. Hlavný dôvod, prečo som si sady definoval a implementoval do návrhu je uľahčenie zadávania známok žiakovi na konci roku. Rozsiahlejším popisu dôvodov a funkcii sád sa venuje kapitola Analýza návrhu. Sada je identifikovaná kódom alebo názvom. Záložka pracuje s tabuľkami *Sk_Sady* a *Sk_Pred_Sady*.

Obrázok



obr.č. 12 – Záložka Sady

Ovládacie prvky a funkcie

Záložka je opäť veľmi podobná tým predchádzajúcim – hlavnú časť okna tvorí náhľad na sady (komponenta *DBGrid*), vľavo sa nachádzajú textové polia pre zobrazenie údajov označenej sady a pre zadávanie a úpravu dát. Ovládanie je opäť intuitívne, preto sa pokúsím obmedziť jeho popis na minimum. Pridávanie, úpravu a mazanie sád zabezpečuje obvyklá trojica

¹² Táto databázová procedúra slúži na výpis chybových hlásení a má parametre: (text, význam chyby, [zoznam premenných použitých v texte]).

tlačidiel. Stlačením *Pridaj* alebo *Uprav* sa zavolá procedúra *S_Lock*, ktorá má veľmi obdobnú funkciu, ako tomu bolo pri predošlých záložkách u funkcií typu „*x_Lock*“. Do polí užívateľ zadá nové respektíve upraví existujúce údaje.

Na pridanie a úpravu premetov v sade slúži dvojica výberových zoznamov (instancie triedy *TListBox*) a pridružených tlačidiel, ktoré sa nachádzajú hneď pod poliami pre zadanie kódu, názvu a popisu sady. Ľavý zoznam odpovedá utriedeným predmetom v sade, vpravo sú predmety, ktoré môžu byť do sady ešte pridané. Použitím tlačidiel znázorňujúcich šípky užívateľ presúva označené predmety doľava alebo doprava. Metóda *OnClick* týchto tlačidiel skontroluje, či index označeného prvku leží v správnom intervale a s názvom označeného predmetu ako parametrom zavolá metódu *Items.Add* pre zoznam kam šípka smeruje, a *Items.Delete* pre zoznam druhý. Predmety možno v ľavom zozname usporiadať použitím tlačidiel so znakmi plus a mínus. Po potvrdení operácie aplikácia vygeneruje dotaz, ktorý zavolá databázovú procedúru *Sk#_SadaAdd* pre pridanie sady, *Sk#_SadaEdit* pre úpravu sady a *Sk#_PredmetySady* pre zmenu predmetov v sade. V prípade chyby tieto procedúry vygenerujú hlásenie s popisom chyby.

5.6. Záložka Znamky

Popis

Záložka *Znamky* slúži na zobrazenie uzavretých známok žiakov jednotlivých tried v zvolenom roku a polroku. Z tejto záložky tiež možno zahájiť zadávanie známok.

Obrázok

KC	Meno	SLJ01	ANJ01	NEJ01	FRJ01	MATEM	INFOR	INFCV	GEOGR	BIOLG	DEJEP	CH
1	Anderský Tomáš	1	1	1		1	1	1	1	1	1	1
2	Burdová Zuzana	2	3	2		3	3	2	1	2	2	3
3	Cudráková Zuzana											
4	Ďurkovič Daniel											
5	Konček Peter											
6	Kotrba Andrej											
7	Kozák Vladislav											
8	Murgaš Tomáš											
9	Naňo Lukáš											
10	Stuhlová Alena											
11	Szabó Atilla											
12	Tenorová Lucia											
13	Uršová Petra											
14	Uzdecká Milena											
15	Valcha Roman											
16	Zábušek Andrej											
17	Lacko Juraj											

obr.č. 13 – Záložka Známky

Ovládacie prvky a funkcie

V ľavom hornom rohu je roletový zoznam s existujúcimi triedami na škole. Pri zmene triedy aplikácia zavolá vhodný dotaz a vyžiada si od aplikácie triedny knihu danej triedy. Žiaci sú utriedení podľa katalógového čísla a atribútom odpovedajú jednotlivé predmety sady. Dôsledkom týchto faktov je úplná zhoda pri pohľade do triednej knihy a jej ekvivalentu na záložke *Známky*.

Zadávať predmety možno iba triedam, ktoré sú neprázdne. V každom okamžiku môžu byť známky ľubovoľnej triedy zadávané iba jedným užívateľom. Akonáhle užívateľ stlačí tlačidlo *Zadávať známky*, aplikácia pridá do pomocnej tabuľky *S#_Lock* informáciu o tom, ktorej triede sa pridávajú známky. Tým zabráni, aby niekto pridával známky v rovnakom čase ale aj manipulácii so žiakmi a sadou tejto triedy. Presný popis okna pre pridávanie známok nájdete v kapitole 4.4 Zadávanie známok.

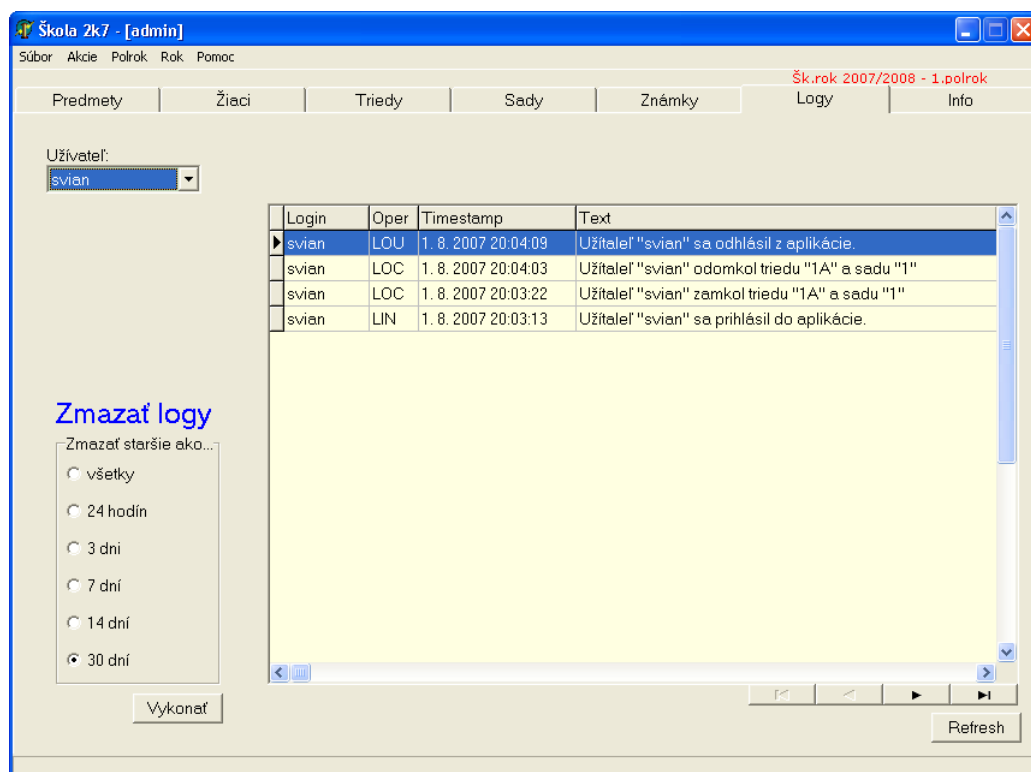
5.7. Záložka Logy

Popis

Logy slúžia na kontrolu toku užívateľov a spätnú kontrolu ich akcií. Medzi inými sa eviduje prihlasovanie a odhlasovanie užívateľov, čím sa zabráni súčasnému prístupu z dvoch rozličných miest pomocou jedného užívateľského účtu. Zaznamenávajú sa tiež nastavenia ako zmena hesla, ale

aj fakt, že užívateľ zahájil zadávanie známok a tým zakázal určité operácie s touto triedou a jej sadou.

Obrázok



obr.č. 14 – Správa logov

Ovládacie prvky a funkcie

V centrálnej časti okna sa nachádza prehľad logov, ktoré sú zoradené podľa časovej známky od najnovšieho po najstarší. Naľavo sa nachádza roletový zoznam, ktorý obsahuje zoznam aktívnych užívateľov aplikácie (tých, čo majú nejaké logy), ktorý slúži ako filter logov podľa užívateľa. Filter „deaktivujeme“ výberom prvej voľby „ALL“ tohto zoznamu.

Logy sa dosť rýchlo hromadia, preto je dobré občas mazať tie najstaršie. Túto akciu umožňuje tlačidlo *Vykonať* a skupina prepínačov *LRG_Del*, čo je instancia triedy *TRadioGroup*. Užívateľ zvolí vhodné obdobie a po potvrdení operácie aplikácia vytvorí a spustí dotaz, ktorým zmaže logy staršie ako vybrané obdobie. Súčasne pridá do databázy log o vykonaní tohto úkonu.

5.8. Ostatné funkcie

Zmena polroku

Zmenu polroku možno vyvolať výberom žiadaného polroku z položky v hlavnom menu s názvom *Polrok*. Aplikácia zmení globálnu premennú, v ktorej si ukladá aktuálny polrok a zavolá metódu *OnShow* pre aktívnu

záložku, ktorá obnoví dáta pre zmenený polrok. Zároveň sa zmení text s aktuálne zvoleným rokom a polrokom, ktorý sa nachádza v pravom hornom rohu.

Zmena roku

Pri potrebe nahliadnuť do archívu užívateľ vyberie príslušný ročník zo zoznamu, ktorý sa nachádza v položke *Rok* v hlavnom menu. Pri voľbe niektorého zo starších ročníkov aplikácia zakáže určité funkcie – napríklad správu užívateľov a nastavení. Potom zmení vlastnosti pripojenia a pripojí sa k databáze zo žiadanými dátami. Na ukončenie práce s archívom slúži voľba *Aktuálny*, ktorá sa nachádza v zozname s ročníkmi.

Prechod na nový rok

Prechod na nový rok je veľmi citlivá operácia. Aplikácia pri nej urobí zálohu súčasnej databázy a do novej prenesie všetkých užívateľov, predmety a sady. Žiakov posunie o ročník vyššie a tých, čo boli v poslednom ročníku, aplikácia vyradí z databázy.

6. Uživatelská príručka

6.1. Požiadavky

Hardwarové požiadavky

Beh aplikácie predpokladá existenciu servera a niekoľkých staníc. Server môže byť vzdialený alebo lokálny. Požiadavky na konfiguráciu servera môžeme nájsť v inštaláčnej príručke Microsoft SQL Server 2005 Express. Pre potreby našej aplikácie však postačí, keď bude mať konfiguráciu podobnú odporúčanej konfigurácii pre stanice.

Každá klientská stanica, by pre korektný beh aplikácie mala obsahovať nasledovnú minimálnu konfiguráciu PC:

- 1,5 GHz procesor
- 256 MB operačnej pamäte
- 10 MB diskového priestoru

V prípade používania vzdialeného SQL serveru sa prirodzene požaduje aj pripojenie do siete internet. Každá stanica na ktorej má prebiehať tlač vysvedčení musí mať prístup k tlačiarňam.

Pre plynulý a bezproblémový chod aplikácie je doporučená o niečo silnejšia konfigurácia:

- 2 GHz procesor
- 512 MB operačnej pamäte
- 20 MB diskového priestoru

Softwarové požiadavky

Informačný systém beží na platforme Windows, konkrétne je požadovaný operačný systém Microsoft Windows XP alebo Microsoft Windows Vista. Na serveri by mal byť nainštalovaný Microsoft SQL Server 2005, alebo jeho orezaná verzia Microsoft SQL Server 2005 Express, ktorá je pre nekomerčné účely voľne dostupná. Na tomto serveri by mal existovať účet s plnými právami k aspoň jednej databáze.

6.2. Spustenie

Inštalácia

Spustíte súbor „setup.exe“. Inštaláčny sprievodca vyzve k zadaniu cieľového adresára a nainštaluje aplikáciu. Na začatie práce s aplikáciou použijete spustiteľný súbor „skola2k7.exe“, ktorý sa nachádza v cieľovom adresári.

Prvé spustenie

Spustením súboru „skola2k7.exe“ sa nám otvorí prihlasovacie okno spolu s upozornením o nefunkčnosti spojenia, zobrazené na obr.č. 3. Je nutné nastaviť korektný prístup k databázovému serveru. Najskôr musíme vyvolať okno pre nastavenie funkčného databázového spojenia. Toto okno môžeme vyvolať z menu hlavného okna, položky *Súbor -> Nastavenia*, alebo priamo malým tlačidlom, ktoré sa nachádza pri hlásení o nefunkčnosti spojenia. Spojenie by mal nastavovať administrátor aplikácie.

Nastavenie spojenia

Po vyvolaní okna *Nastavenia* zvolíme záložku *Pripojenie*, ktorá je zobrazená na obr.č. 4. Tu sa nachádzajú kolónky pre nastavenie názvu servera, názvu databázy, prihlasovacieho mena a hesla s plnými právami k tejto databáze a kolónka pre nastavenie aktuálneho školského roku. Nájdeme tu tiež zaškrŕavacie políčko, ktoré určuje, či sa jedná o lokálny alebo vzdialený server.

V prípade, že už máte uložený súbor s nastaveniami, môžete nastavenia načítať kliknutím na *Načítať nastavenia* a zvolením príslušného súboru. Nastavenia môžete tiež ukladať, a to kliknutím na *Uložiť nastavenia*. Na obnovenie pôvodných hodnôt poslúži tlačidlo *Reset*, ktoré sa nachádza hneď pod textovými poliami.

Tlačidlo *Test* slúži na testovanie funkčnosti aktuálne nastaveného spojenia. Tlačidlo *Uložiť* uloží nastavenia a znovu spustí aplikáciu s novými nastaveniami. Tlačidlo *Vytvor kostru* slúži na vytvorenie kostry databázovej aplikácie – funkcia vytvorí na serveri všetky tabuľky a funkcie nutné pre fungovanie aplikácie a skopíruje inicializačné dáta. Túto akciu je nevyhnutné vykonať pri prvom spustení aplikácie na novom serveri. V prípade existencie bude pôvodná kostra prepísaná novou a všetky dáta stratené. Preto by sa malo toto tlačidlo používať iba v prípade, že nás k tomu aplikácia vyzve. Po vytvorení novej kostry aplikácia obsahuje iba 1 užívateľský účet, a síce účet „admin“ s inicializačným heslom „testin“. Toto heslo by si mal admin bezodkladne zmeniť.

Prihlasovanie

Ak je nastavené správne spojenie, na okne pre prihlásenie do aplikácie sa prístupní tlačidlo *Prihlásiť*, vid' obr.č. 2. Po zadaní správneho mena a hesla a stlačení tlačidla pre prihlásenie je užívateľ prihlásený do aplikácie a prístupnia sa mu funkcie v závislosti na jeho právach.

6.3. Zmeny nastavení a prostredia

Orientácia v prostredí

Aplikácia Škola2k7 sa skladá z hlavného okna a niekoľkých vedľajších okien. Hlavné okno má 7 záložiek, z ktorých jedna má čisto informatívny charakter. Zvyšných 6 obsahuje ovládacie prvky na obsluhu a správu istej

časti dát, ktoré súvisia s názvom tejto záložky. Tieto záložky sú: *Predmety*, *Žiaci*, *Triedy*, *Sady*, *Známky* a *Logy*. Pre plnú funkčnosť aplikácie je nevyhnutné používať aj hlavné menu, ktoré obsahuje položky: *Súbor*, *Akcie*, *Polrok*, *Rok*, *Pomoc*. Popis všetkých funkcií hlavného okna nájdeme v kapitole 6.5.

Zmena hesla

Zmeniť heslo si môže každý užívateľ v záložke *Môj profil* okna s nastaveniami. Okno *Nastavenia* vyvoláme pomocou menu hlavného okna: *Súbor* -> *Nastavenia*. Zadajte staré heslo, nové heslo a pre overenie zadajte nové heslo znovu. Ak ste zadali staré heslo správne a obidve zadania nového hesla sú zhodné, dojde k zmene hesla.

Odhlásenie z aplikácie

Odhlásenie z aplikácie vyvoláte pomocou voľby *Logout* položky *Súbor* v hlavnom menu aplikácie. Opätovne sa zobrazí okno pre prihlásenie do aplikácie a ďalší užívateľ sa môže prihlásiť.

Zmena roku a polroku

Aktuálne nastavený rok a polrok môžeme nájsť v pravom hornom rohu hlavného okna odlišený červeným písmom. Polrok zmeníme vybraním položky *Polrok* v hlavnom menu a zvolením prvého alebo druhého polroku. Ak chceme nahliadnuť do archívu, použijeme položku hlavného menu *Rok*, a zvolíme príslušný archív. Pri práci s archívom sú zakázané určité funkcie, napríklad meniť nastavenia alebo spravovať užívateľov. Pre ukončenie práce s archívom opäť použijeme položku *Rok* a zvolíme možnosť *Aktuálny*.

Prechod na nový školský rok

Po ukončení školského roku je potrebné prejsť na nový školský rok. Aplikácia sa postará o uloženie toho starého do archívu a vytvorí nový. Môžete tak urobiť použitím voľby *Prejsť na nový rok* z položky *Akcie* hlavného menu.

6.4. Správa užívateľov

Okno pre správu užívateľov

Ovládacie prvky pre spravovanie užívateľov sa nachádzajú v okne *Nastavenia* na záložke *Užívateľia*. Okno *Nastavenia* vyvoláme pomocou menu hlavného okna: *Súbor* -> *Nastavenia*. Okno obsahuje prehľadný zoznam užívateľov a textové polia pre zobrazenie údajov a práv aktuálne označeného užívateľa. Posúvanie po užívateľoch je možné klikaním na zoznam užívateľov alebo je možné sa posúvať použitím posuvného ovládanie pod zoznamom. Vytvárať, upravovať a mazať užívateľské účty môže iba administrátor aplikácie. Preto on je jediný užívateľ, ktorému bude záložka *Užívateľia* zobrazená.

Pridanie nového užívateľa

Stlačením tlačidla *Pridaj*, ktoré sa nachádza v pravom dolnom rohu záložky pre správu užívateľov sa nám zobrazia čisté textové polia. Administrátor vyplní užívateľské údaje: meno, priezvisko a rodné číslo nového užívateľa, ako aj login a inicializačné heslo. Ešte je potrebné nastaviť práva užívateľa. Každý záložke odpovedá jedno zaškrťavacie políčko. Fakt, že užívateľ má právo obsluhovať príslušnú záložku vyjadríme zaškrtnutím tohto políčka. Po nastavení všetkých potrebných údajov potvrďte pridanie tlačidlom *OK*. Pridávanie môžete kedykoľvek zrušiť pomocou tlačidla *Zrušiť* alebo klávesy *ESC*.

Úprava existujúceho užívateľa

V prípade, že chceme užívateľovi nejaké údaje zmeniť, musíme ho označiť. Úpravu vyvoláme kliknutím na tlačidlo *Uprav*. Všetky polia s údajmi sa nám sprístupnia na upravovanie. Potom, čo dokončíme úpravy potvrdíme zmeny kliknutím na *OK*, prípadne môžeme upravovanie stornovať použitím tlačidla *Zrušiť*.

Odstránenie užívateľa

Ak chceme užívateľa odstrániť, musíme sa opäť použitím zoznamu alebo posuvného ovládania nastaviť na správneho užívateľa. Po stlačení *Zmaž nás* aplikácia vyzve k potvrdeniu operácie. Užívateľ „admin“ nemôže byť zmazaný.

6.5. Hlavné funkcie aplikácie

Prevažná väčšina ovládacích prvkov sa nachádza v hlavnom okne na niektorej zo záložiek. Rozloženie a funkcia ovládacích prvkov na záložkách je takmer zhodná, čo zjednodušuje a zrýchľuje prácu s nimi. Každá záložka obsahuje prehľadnú tabuľku s dátami a možnosťou voľby atribútu, podľa ktorého majú byť dáta triedené. Nachádzajú sa tam aj textové polia pre zobrazenie údajov aktuálne zvoleného záznamu a pre úpravu a zadávanie nových hodnôt.

Skupina ovládacích prvkov pre vyhľadávanie sa nachádza v ľavom dolnom rohu na tých záložkách, u ktorých má vyhľadávanie v dátach nejaký zmysel. Vyhľadávať možno na úplnú zhodu, alebo možno použiť okamžité vyhľadávanie na aktuálnu zhodu. Pri vyhľadávaní na úplnú zhodu zvolte atribút podľa ktorého chcete hľadať, zadajte hľadaný reťazec a stlačte *Hľadať*. V prípade úspechu vás aplikácia nastaví na vyhľadaný záznam, v opačnom prípade sa zobrazí hlásenie o neúspešnom hľadaní.

Pri prepnutí na okamžité hľadanie tabuľka s dátami zmení farbu na modrú, čo nám signalizuje, že posúvanie po záznamoch bude automatické a nemožno meniť aktuálne zvolený záznam klikaním naň ani používaním posuvného ovládania. Textové pole pre vyhľadávaný reťazec reaguje na zmenu a v každom okamžiku je zvolený záznam s aktuálnou zhodou.

V pravom hornom rohu sa nachádzajú tlačidlá *Pridaj*, *Uprav* a *Zmaž* a tlačidlá *OK* a *Zrušiť* pre potvrdenie respektíve zrušenie operácie. Vpravo dole sa nachádza tlačidlo *Refresh* pre obnovenie aktuálnosti dát.

Spravovanie tried

Na pridávanie, úpravu a mazanie tried slúži záložka *Triedy*. Trieda má 6 atribútov: jednoznačný identifikátor *ID*, *Ročník*, *Trieda*, sada triedy *Sada*, jej triedny učiteľ *T_Ucitel* a študijný odbor *Odbor*. Náhľad na triedy možno zoradiť podľa triedneho učiteľa, alebo podľa ročníku a označenia triedy. Záložka je zobrazená na obr.č. 11.

Pri pridávaní triedy stlačte *Pridaj*. Textové polia s údajmi sa sprístupnia a sú pripravené na zadanie nových hodnôt. Identifikátor triedy a študijný odbor zadajte z klávesnice, na voľbu ostatných položiek vám poslúžia roletové zoznamy. V prípade, že sa na škole nachádzajú dvaja a viac učiteľov s rovnakým menom, je potrebné pri voľbe triedneho učiteľa sledovať pomocnú informáciu o rodnom čísle aktuálne zvoleného učiteľa, aby nedošlo k zámene. Pridanie potvrdíte tlačidlom *OK*, prípadne kedykoľvek počas pridávania zrušte tlačidlo *Zrušiť* alebo klávesov *ESC*.

Na úpravu študentov slúži tlačidlo *Uprav*. Upravuje sa aktuálne zvolená trieda. Operácia je podobná operácii pridávania, rozdiel je v tom, že textové polia budú inicializované pôvodnými hodnotami.

Pre zmazanie aktuálne zvolenej triedy stlačte *Zmaž*. Aplikácia vyzve k potvrdeniu operácie. Myslite na to, že pred zmazaním je nutné zmazať všetkých jej študentov, prípadne ich preradiť do inej triedy. Mazanie neprázdnej triedy nie je povolené.

Spravovanie sád

Spravovanie sád zabezpečuje záložka *Sady*, zobrazená na obr.č. 12. Každá sada má svoj jednoznačný kód a názov, popis a utriedený zoznam predmetov.

Pridanie sady vyvoláme stlačením *Pridaj*. Do prázdnych polí zadáme kód, názov a popis sady. Zostáva už iba pridať a usporiadať predmety. Na to nám poslúžia 2 zoznamové polia a pridružené ovládacie prvky. V pravom zoznamovom poli nájdeme zoznam predmetov na škole. Ľavé zoznamové pole odpovedá usporiadanému zoznamu predmetov v sade. Používaním šípkov, ktoré sa nachádzajú medzi týmito zoznamami môžeme presúvať označený predmet sprava doľava a opačne, tým ho pridávame, respektíve odoberáme zo sady. Použitím tlačidiel „+“ a „-“ môžeme posúvať označený predmet v ľavom zozname hore a dole a tým meniť poradie predmetov v sade. Pridanie sady potvrdíte pomocou *OK*. Pridanie možno v ktorejkoľvek fáze zrušiť tlačidlom pre zrušenie operácie alebo klávesov *ESC*.

Úprava prebieha analogicky s operáciou pridávania s tým rozdielom, že textové polia a zoznamové polia sú predvyplnené údajmi upravovanej sady.

Pre zmazanie označenej sady použite tlačidlo *Zmaž*. Aplikácia vyzve k potvrdeniu úkonu. Sadu, ktorú majú zapísané nejaké triedy nemožno zmazať, čo aplikácia oznámi príslušným hlásením.

Spravovanie žiakov

Na správu žiakov na škole nám poslúži záložka *Žiaci*, ktorá je vyobrazená na obr.č. 10. Žiaka môžeme priradiť len do existujúcej triedy. Ak žiakova trieda na škole ešte neexistuje, musíme vytvoriť najskôr túto triedu. Postup nájdeme v odseku Spravovanie tried.

Na pridávanie žiakov slúži tlačidlo *Pridaj*. Zobrazia sa nám prázdne textové polia pre zadanie hodnôt. Vyplňte rodné číslo, meno a priezvisko žiaka. Z roletového zoznamu vyberte triedu, do ktorej chcete žiaka pridať. V tomto zozname sa nachádzajú všetky existujúce triedy na škole. Zadajte dátum nástupu v tvare „rok/polrok“, napríklad 2007/1. Doplňte číslo v katalógu, rodisko, okres, národnosť a štátnu príslušnosť. Na výber okresu slúži roletový zoznam. V prípade, že ste v tomto zozname okres žiaka nenašli, prejdite na odsek Okresy a kraje. Na záver ešte zo zoznamového poľa vyberte, ktoré predmety žiak bude navštevovať. V tomto zozname sa nachádzajú iba predmety sady jeho triedy, preto sa tento zoznam mení pri každej zmene cieľovej triedy žiaka. Akciu potvrdíme tlačidlom *OK*.

Úpravu iniciujeme tlačidlom *Uprav*. Textové polia a roletové zoznamy sú vyplnené hodnotami zvoleného študenta. Pri menení potrebných údajov postupujte analogicky s procesom pridávania, zmeny potvrdíte tlačidlom *OK*. Na zrušenie akcie slúži tlačidlo *Zrušiť*.

Tlačidlom *Zmaž* spustíte overenie správnosti tohto úkonu a pri kladnom výsledku budete vyzvaný k potvrdeniu operácie. V opačnom prípade aplikácia zobrazí dôvody, ktoré bránia zmazaniu študenta.

Spravovanie predmetov

Na spravovanie predmetov slúži rovnomenná záložka. Predmety majú svoj kód, názov a typ. Pridávanie nového predmetu vyvoláme tlačidlom *Pridaj*. Napíšeme kód a plný názov pridávaného predmetu a zvolíme jeho typ. Vedľa roletového zoznamu pre výber typu predmetu sa nám zobrazuje doplnková informácia o type predmetu. Akciu potvrdíme alebo zrušíme pomocou tlačidiel *OK* alebo *Zrušiť*. Záložka je zobrazená na obr.č. 9 – Záložka pre správu predmetov.

Úpravu aktuálne označeného predmetu vyvoláme tlačidlom *Uprav*. Zmeníme potrebné údaje a akciu potvrdíme pomocou *OK*. Úpravu môžeme kedykoľvek odvolať tlačidlom *Zrušiť* alebo klávesov *ESC*. Pre zmazanie zvoleného predmetu stlačte *Zmaž*. Musíte ale myslieť na to, že nemožno zmazať predmet, ktorý je zapísaný v nejakej sade. Je potrebné najskôr predmet zo všetkých sád odstrániť. Fakt, že predmet sa nachádza v nejakých sádach môžeme zistiť z textového poľa pod údajmi o predmete, ktoré obsahuje zoznam sád, v ktorých sa predmet nachádza. Ak nič nebráni zmazaniu, aplikácia vyzve k potvrdeniu operácie.

Práca s logmi

Logy slúžia na kontrolu toku užívateľov a ich akcií. Na prácu s nimi slúži záložka *Logy*, zobrazená na obr.č. 14 – Správa logov. Právo prezerat' logy ako aj mazať staré logy by nemal mať každý užívateľ. Predpokladá sa, že

administrátor dá toto právo maximálne dvom až trom zástupcom. Zaznamenávajú sa tu akcie ako prihlásenie a odhlásenie užívateľa, zmena hesla, ale aj zamknutie triedy a sady pri zadávaní známok nejakej triedy alebo tlačí vysvedčení. V ľavom hornom rohu sa nachádza filter podľa užívateľa. V ľavom dolnom rohu sa nachádza skupina ovládacích prvkov na mazanie logov starších ako určité obdobie. Zvoľte obdobie a stlačte *Vykonať*.

Okresy a kraje

Aplikácia obsahuje samostatné okno pre prácu s okresmi a kraji. Aj keď po inštalácii by aplikácia mala poznať všetky potrebné kraje a okresy daného štátu, je možné, že žiak sa narodil v inom štáte, preto aplikácia umožňuje upravovať a pridávať nové okresy a kraje. Pritom rozlišuje 2 typy okresov. Okresy, ktoré sú v aplikácii od inštalácie a okresy, ktoré boli pridané užívateľmi. Prvý typ okresov dokáže spravovať iba administrátor, pretože ich častá úprava sa nepredpokladá.

Okno pre spravovanie okresov môžete zobrazit' vybratím voľby *Správa okresov* položky *Akcie* menu hlavného okna aplikácie. Výzor okna môžete vidieť na obr.č. 5. Administrátorovi sa zobrazí aj zaškrŕavacie políčko „ukázať predvolené“, ktoré určuje či bude pracovať s oboma typmi okresov. Vpravo sa nachádza náhľad na okresy a kraje. Vľavo dole sú tlačidlá *Pridaj*, *Uprav* a *Zmaž*, ktoré majú podobnú funkciu ako pri spravovaní záložiek a ich použitie by mal zvládnuť bez väčších problémov každý užívateľ aplikácie.

Zadávanie známok

Záložka *Známky* nám umožňuje náhľad na triedne knihy s uzavretými známami pre aktuálny rok a polrok pre zvolenú triedu. V ľavom hornom rohu sa nachádza roletový zoznam na voľbu triedy.

Na dolnej hrane záložky je tlačidlo, ktoré otvorí okno na zadávanie známok zvolenej triede, vyobrazené na obr.č. 6. V jednom okamžiku môže každej triede zadávať známky najviac jeden užívateľ. V prípade, že danej triede už niekto známky zadáva, bude tlačilo pri voľbe tejto triedy zakázané. Musíte počkať, kým iný užívateľ skončí zadávanie a tlačidlo sa sprístupní.

Po otvorení okna na zadávanie známok zvoľte žiaka, prípadne stlačte tlačidlo *Ďalší*. Označí sa vám kolónka pre prvú známku. Po zadaní korektnej hodnoty ukazovateľ automaticky preskočí na ďalšie políčko so známou. Po zadaní všetkých známok ukazovateľ preskočí na tlačidlo *Ďalší*, takže stlačením klávesu *Enter* sa presunieme na ďalšieho žiaka, prípadne môžeme tlačidlo stlačiť myšou, alebo vyberieme konkrétneho žiaka z roletového zoznamu s menami žiakov. Po ukončení pridávania známok stlačte *Uložiť* a okno pre zadávanie známok môžete zavrieť.

6.6. Tlač vysvedčení

Keď chcete tlačit' vysvedčenia, musíte overit', či má stanica na ktorej pracujete prístup k tlačiarni. Hlavné okno pre obsluhu tlače, zobrazené na

obr.č. 7, vyvoláte z položky hlavného menu *Akcie* vybraním voľby *Tlačiť vysvedčenia*.

Používanie šablón

Pri tlačení sa využívajú šablóny. Šablóna je kompletná sada nastavení potrebná pre vytlačenie vysvedčenia. Uložená je v jednom súbore a obsahuje nastavenie výstupu, nastavenie slovného pomenovania známok, nastavenie typu písma a hlavičky školy. Pri nastavení tlače si môžete vytvoriť novú šablónu, prípadne otvoriť a upraviť šablónu už existujúcu. Tlačidlo *Nová šablóna* vytvorí novú šablónu, následne je potrebné nastaviť všetky hodnoty a určiť formát výstupu.

Oveľa častejšie budete šablónu otvárať zo súboru. Na to slúži tlačidlo *Načítať šablónu*. Po zvolení správneho súboru sa načítajú všetky nastavenia vrátane formátu výstupu. Následne môžete hodnoty a nastavenia upravovať. Popis úprav nájdete v nasledujúcich odsekoch. Keď si šablónu nastavíte podľa vašich predstáv, je dobré si ju do budúcnosti uložiť. Stlačte *Uložiť šablónu* a zvolte názov súboru.

Nastavenie formátu výstupu

Stlačte *Upraviť výstup*. Zobrazí sa okno *Formát vysvedčenia*, vid' obr.č. 8. Hlavnú časť okna zaberá biela plocha, ktorá znázorňuje tlačivo pre vysvedčenie. Na tejto ploche sa nachádzajú 2 typy objektov: údajové a predmetové. Ak ste vytvorili novú šablónu, bude táto plocha prázdna. V pravej časti sa nachádza ovládací panel pre nastavenie vlastností týchto objektov a sada tlačidiel pre pridávanie objektov a uloženie zmien.

Údajový objekt má 1 kolónku. Do tejto kolónky bude pri tlačení vysvedčenia doplnený údaj, ktorý zvolíte z roletového zoznamu na ovládacom paneli. Ďalšie parametre tohto objektu sú veľkosť písma, ktorou bude vytlačený a súradnice na ploche.

Predmetový objekt má 3 kolónky. Prvá kolónka určuje, kde bude vytlačený názov predmetu, druhá a tretia určujú kde bude vytlačená známka z prvého a druhého polroku z tohto predmetu. Parametre sa nastavujú na ovládacom paneli. V textovom poli *Názov* zadajte názov tohto objektu. Názov je to čisto pomocná informácia a o jeho voľbe rozhoduje užívateľ, mal by však vypovedať o skupine predmetov, ktoré do neho možno priradiť. Nižšie sa nachádzajú polia pre zadanie súradníc a veľkosti písma. Ďalej nájdeme polia pre nastavenie vzdialenosti medzi jednotlivými kolónkami. Nasledujú 3 zaškrtačacie políčka. Prvé z nich určuje, či sa má tlačiť aj hlavička predmetu, alebo len samotné známky. Toto políčko zaškrtnite v prípade, že sa názov predmetu na tlačivo nenachádza. Druhé políčko určuje, či má byť hlavička predmetu vytlačená celá, alebo iba jeho časť. To sa využíva v prípade, že tlačivo obsahuje príponu podobnú „výchova“ alebo „jazyk“ a chcete aby sa tlačila iba prvá časť názvu. Posledné určuje, či sa v prípade neuspokojenia požiadavky na vytlačenie niektorého z predmetov má doplniť najbližší voľný predmet. Na spodnej časti ovládacieho panelu sa nachádzajú 2 zoznamové polia pre nastavenie predmetov a ich priority pri tlači. Ľavý zoznam určuje, ktoré predmety a v akom poradí má program pri vyhodnocovaní doplniť. Z

pravého zoznamu použitím šípok presúvajte predmety a použitím tlačidiel so symbolmi plus a mínus posúvajte označený predmet hore či dole.

Ak chcete s objektom vykonávať akúkoľvek akciu, najskôr ho musíte mať označený. Objekt označíme tak, že naň ukážeme myšou a stlačíme jej ľavé tlačidlo. Fakt, že je objekt označený spoznáme podľa toho, že je jeho kolónka vyfarbená na modro. Ovládací panel pre nastavenie objektov obsahuje údaje aktuálne označeného objektu a umožňuje tieto údaje upravovať.

Posúvanie objektu je možné vykonať dvoma spôsobmi. Najskôr musíte objekt označiť. Pozíciu môžete meniť zadávaním x-ovej a y-ovej súradnice do príslušných polí na ovládacom paneli, alebo kliknutím na dané miesto.

Pre umiestnenie nového údajového objektu stlačte *Pridať nový údaj*. V ľavom hornom rohu sa vám objaví nový objekt – kolónka pre údaj. Na ovládacom paneli v pravej časti okna použitím roletového zoznamu zvolte údaj, ktorý bude do kolónky dopĺňaný. Následne zvolte veľkosť písma, ktorou bude údaj vytlačený. Nezabudnite objekt posunúť na potrebné miesto.

Nový predmetový objekt pridáte stlačením *Pridať novú známku*. Opäť sa objaví nový objekt v ľavom hornom rohu bieleho panelu. Nastavte príslušné parametre a posuňte objekt na zvolené miesto.

Známkovanie

Použitím tlačidla *Známkovanie* vyvoláte okno pre nastavenie slovného popisu známok. Nastavuje sa odlišný slovný popis pre známky z bežných predmetov a pre známky zo správania. Dopĺňajte ku každej známke jej textový ekvivalent a pre zachovanie zmien po ukončení zadávania stlačte *Uložiť*.

Ostatné nastavenia

Pred tlačou je ešte nutné nastaviť, či chcete tlačiť vysvedčenia pre celú školu, alebo iba konkrétnu triedu. Potom zvolte, či chcete vytlačiť známky z prvého, druhého alebo oboch polrokov. Nakoniec je ešte nutné zadať hlavičku školy a zvoliť typ písma, ktoré má byť použité na tlač.

Spustenie tlače

Pre spustenie tlače použite tlačidlo *Tlačiť* a zvolte funkčnú tlačiareň. Tlač môžete kedykoľvek prerušiť stlačením tlačidla pre prerušenie tlače.

7. Záver

7.1. Zmeny pri vývoji

Analýza a tvorba návrhu zabrali značnú časť vývoja. Uvedomoval som si najmä skutočnosť, že keď si návrh nepremyslím dostatočne, môžu ma neskoršie úpravy stáť nemalé časové náklady. Napriek faktu, že som si predsavzal aplikáciu vyvíjať štýlom „dvakrát meraj a raz rež“, musel som sa opakovane pri implementácii vracat' k návrhu a často meniť podstatné časti riešenia. Ukážkovým omylom bol predpoklad, že okresy a kraje budú aplikácii dodané bez možnosti ich úpravy vo forme konfiguračného súboru. Neskôr sa ukázalo, že toto riešenie nezohľadňuje zahraničných študentov, čo si vyžadovalo zásah do databázového modelu aplikácie a riešenie s tým súvisiacich komplikácií.

Podmienka spraviť aplikáciu odolnú voči užívateľom bez skúseností s prácou na PC a bez zmyslu pre poriadok a správne používanie bola časovo náročná. Bohužiaľ často táto požiadavka súťažila s požiadavkou na praktickú a rýchlu obsluhu, preto sa museli pri návrhu užívateľského rozhrania robiť kompromisy.

Zabezpečenie aplikácie prihlasovacím menom a heslom zabraňuje bežným užívateľom, aby neoprávnene používali aplikáciu. Prihlasovacie údaje na databázový server sú však uložené v aplikácii, preto by nemal byť prístup k počítačom s aplikáciou voľný.

7.2. Zhrnutie

Cieľom bakalárskej práce bolo vytvoriť informačný systém pre stredné a základné školy, ktorý by uľahčoval problematiku archivácie a vystavovania vysvedčení a ktorého užívateľské rozhranie by bolo tak praktické a intuitívne, aby časovú náročnosť týchto úkonov zmenšil na minimum. Prácu s týmto systémom musí zvládať aj užívateľ s podpriemernou znalosťou práce na PC. Dôležité bolo implementovať prehľadné a ľahko ovládateľné grafické rozhranie pre nastavenie výstupu na tlačiareň, keďže tlačivá pre vysvedčenia sú rôzne a môžu sa meniť, ale aj kvôli rozličným výstupom odlišných tlačiarní. Aplikácia je schopná už po relatívne krátkom zoznámení s prostredím výrazne urýchľovať rutinné práce. Myslím, že aplikáciu sa napriek problémom podarilo dotiahnuť do stavu, kedy by mohla byť nasadená do prevádzky.

Praktické sa ukázalo rozhodnutie rozdeliť vývoj aplikácie na 2 vrstvy – aplikačnú a databázovú – a stratégia naprogramovať na databázovej úrovni čo najviac z funkcií a taktiež všetku kontrolu obmedzení a korektnosti dát.

8. Bibliografia

- [1] Cantù M. (2002): Myslíme v jazyku Delphi 6 – 1.díl. Grada Publishing a.s., Praha.
- [2] Cantù M. (2002): Myslíme v jazyku Delphi 6 – 2.díl. Grada Publishing a.s., Praha.
- [3] Hernandez M. J. (2006): Návrh databází. Grada Publishing a.s., Praha.
- [4] Pacheco X., Teixeira S. (2002): Mistrovství v Delphi 6. Computer Press, Praha.