СЕРИЈСКО ШТАМПАЊЕ ФОРМИ ИЗ БАЗЕ ПОДАТАКА (ПРИКАЗАНО НА ПРИМЕРУ ШКОЛСКИХ ДОКУМЕНАТА)

SERAIL PRINTING OF FORMS FROM DATABASE (DEMONSTRATED ON SCHOOL DOCUMENTS)

Аутор:
ПЕТАР ПРВУЛОВИЋ, ученик IV разреда Паланачке гимназије
Смедеревска Паланка, Регионални центар за таленте Пожаревац[1]
Ментор:
НАТАША МАРКОВИЋ, професор рачунарства и информатике
Паланачка гимназија, Смедеревска Паланка

РЕЗИМЕ:

У последњих пар година повећањем опремљености рачунарском техником у школама запажена је тежња за дигитализацијом података ради лакшег вођења разних докумената, при чему је то највише изражено у свођењу оцена на крају класификационих периода, где рачунање просека, процената и сл. одузима доста времена. Нажалост софтверске подршке на овом пољу готово да нема те се професори сналазе на разне начине, најчешће коришћењем Excel табела. У раду су приказани неки концепти и проблеми на које је потребно обратити пажњу током развоја апликације ове намене, праћени њеним развојем и рализацијом. Предност приказане апликације над нпр. Excel табелама је управо могућност аутоматизације штампе (нпр. листе оцена за сваког ученика на тромесечју) и чување података у базама.

Кључне речи: општи број, база података, форма, апликација, едитор

SUMMARY:

In the last few years increase of accomplishment with computer technology in schools affected tendency for digitalization of data seeking easier handling of miscellaneous documents, whereby is mostly expressed in recording and calculating grades at the end of classification periods, where calculating average grades, percents etc takes a lot of time. Unfortunately software maintenance on this field nearly doesn't exist, so proffesors are forced to find other ways, most often using various Excel tables. In this paper some main concerts and problems which needs to be heed during development of an application of this type, followed by it's development, are represented. Privilege of illustrated application over, for example Excel tables, is just the abillity of automation of printing (for example grades list for each student in class at the end of term) and keeping data in databases.

Keywords: generic number, database, form, application, editor

---

[1] ***Adresa centra**: Regionalni centar za talente, Požarevac, Sinđelićeva 4*

1. Увод

Пратећи трендове дигитализације података приметна су настојања разредних старешина за организовањем и уређивањем разних докумената на рачунару. При томе се првенствено мисли на разне форме за уписивање и штампање оцена. Што се софтвера тиче, на овом пољу подршка је, бар на нашим просторима, заиста слаба, и огледа се у пар програма за вођење дневника и слично. Један од већих проблема било је штампање сведочанстава, оцена ученика на тромесечјима и извештаја за седнице који су, уз наведено, захтевали и рачунање разних просека и процената као и сабирање изостанака. Ово је донекле олакшано прављењем разних табела у Excel-у, које делимично олакшавају сав тај рад, али имајуману да им форма штампања није прилагодива, те се унете оцене могу штампати само у облику табеле, па сав труд око уношења оцена остаје заробљен у тој "крутој" форми. Намера овог рада је управо покушај да се сви ови проблеми (или бар неки најзначајнији) реше, и то повећањем корисности утрошене енергије у односу на добијени резултат. Програм је дизајниран тако да се све оцене одељења садрже у једној табели базе података, налик на Excel-ове, какве се често користе, с тим што су у овој садржане све закључне оцене током школовања на полугођу и крају године. За њихово попуњавање утрошиће се исто времена колико и за попуњавање Excel табеле, али ће могућности коришћења података бити вишеструке. То је омогућено едитором, такође садржаним у програму, који омогућава прављење произвољних форми за штампање података и оцена ученика, при чему се штампање врши серијски, тј. аутоматизовано се попуњава садржај објеката у форми. У пракси, са корисничке стране, ово значи да разредни старешна (циљни корисник) треба да периодично допуњава табелу свог одељења и затим је повеже са потребном формом за штампање (тј. документом - сведочанство, оцене са тромесечја, извештај и сл.), при чему има слободу израде форме каква му је потребна (у едитору), одреди листу ученика из табеле чије податке/оцене жели да штампа, и пусти програм да одштампа тај документ(форму) за изабране ученике. Звучи згодно, зар не?

2. Материјал и методи рада

2.1 Апликација за штампање

2.1.1 Захтеви

Апликација треба да омогући повезивање са табелом базе и штампање података из ње на одређеним местима на хартији. Што се тиче повезивања са табелом потребно је омогућити прављење листе ученика чији ће се подаци у штампању користити (за случај да је потребно одштампати податке само за одређен број ученика). Потребно је развити и могућност израде произвољних форми за штампање, тј. едитор. Фокус програма је на едитору, који је и главна апликација, чија је намена дизајнирање форме и штампање, а из којег се позивају остали потребни елементи програма.

2.1.2 Повезивање са табелом базе

Потребно је отворити табелу чији подаци ће се штампати. Успут, а и даље током рада у едитору, могуће је позивање листе и њено попуњавање ученицима из табеле чији ће се подаци штампати.

2.1.3 Едитор

Едитор изгледа као на слици испод. Састоји се из три дела. У левом пољу налази се приказ стране како ће изгледати приликом штампе. У средњем пољу се задају особине објеката, а десно се налази листа свих објеката који ће се штампати.

Приказ стране, условно речено, има три слоја. Први слој подразумева изглед листа на коме ће се штампати подаци и то је у ствари та страна скенирана и постављена као подлога ради лакшег сналажења и убацивања објеката. Други слој представља графичке елементе које је потребно доштампати, нпр. потпис, знак школе и сл. Оба слоја су битмапе које се уносе у програм из спољњег фајла. Трећи слој садржи објекте.

Постоји три врсте објеката – статични текст, динамични текст и функције.

Статични објект је текст који се штампа на сваком листу непромењен. То је на пример листа предмета, неки текст (нпр. " Редован-Ванредан", "Датум рођења") или наслов типа – "Оцене са првог полугођа".

Динамични објекти садрже податке које читају директно из базе. Одређени су именом поља и бројем ученика чији податак садрже.

Број ученика (колоне у табели) може бити одређен и општи. Одређен подразумева да ће се у дати објекат учитавати податак из поља колоне са тим бројем, и његова ће вредност бити непроменљива. Овакви објекти ипак нису сврстани у статичне јер су повезани са базом, па им садржај није одређен у форми већ зависи од вредности поља у бази.

Општи број одређен је страном која је тренутно у фокусу и кораком. Ако је корак 1, онда ће на првој страни тај број бити 1, на другој 2, на трећој 3 итд. Ако је корак 6, на првој страни број ће бити 1, на другој 7, на трећој 13 итд. Ово је у ствари основни принцип који омогућава серијско штампање, јер се на овај начин омогућава приказивање података из различитих колона на различитим странама. Увођење корака потребно је да би се омогућио и случај да је потребно на странама одштампати податке по неколико ученика. Важно је поменути да општи број "броји" само колоне које су у листи за штампање.

Функције чувају податке добијене применом предефинисаних функција на податке у једном или више поља у бази или објектима. Укључене су следеће функције: просек, број, проценат и текстуални опис оцене.

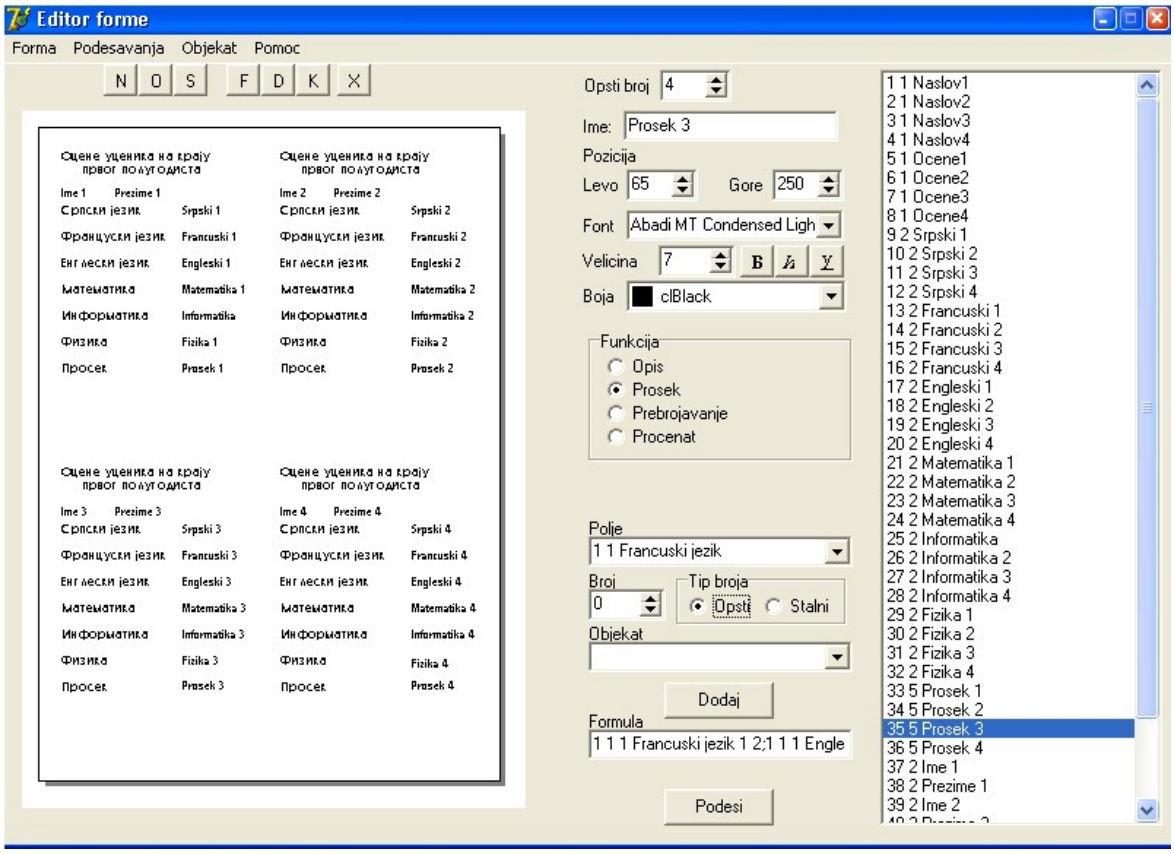
Функција просек вратиће средњу вредност оцена из датих поља.

Функција број вратиће укупан број оцене у датим пољима.

Функција проценат даје проценат одређене оцене у датим пољима. Функција узима као параметре дата поља, и на основу броја оцена и броја поља која је узела одређује простом пропорцијом проценат те оцене.

Текстуални опис оцене за вредност одређеног поља 5,4,3,2,1,0 враћа, респективно, неку од следећих вредности – одличан, врло добар, добар, довољан, недовољан, неоцењен.

Објекти који чувају и приказују податке који ће се штампати на страни чувају се у низу, чиме се омогућава лако приступање и манипулација. Они ће бити типа TLabel и приказиваће име објекта или статичан текст , форматиран.

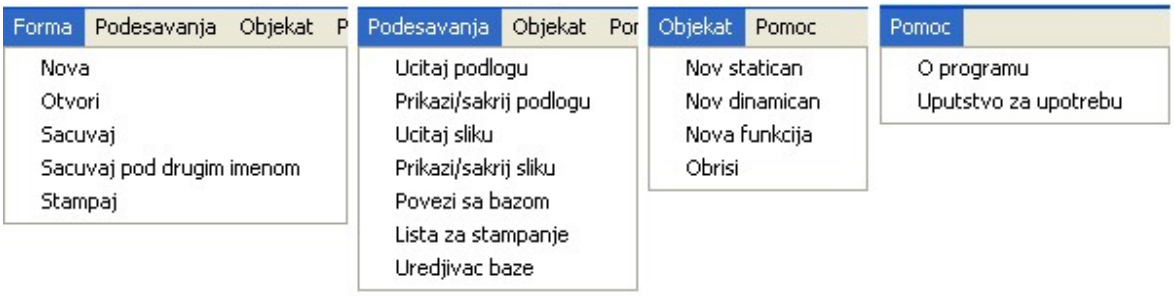


Слика 1. Изглед едитора

2.1.4 Елементи едитора

Главни мени

Главни мени садржи следеће ставке



Слика 2. Главни мени

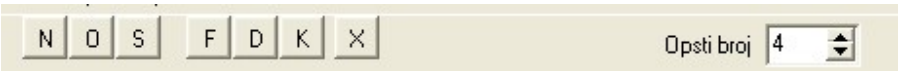
Форма » Предефинисане форме садржи неколико најчешће коришћених форми: сведочанство, извештај са класификационог периода за разредне старешине, оцене одељења за класификациони период у виду табеле и оцене за сваког ученика на тромесечју упаковане по четири на листу (који се сече и даје ученицима).

Подешавања » Учитај подлогу омогућава учитавање слике у први слој, тј. форме на којој ће се штампати. Подешавања » Учитај графику омогућава учитавање слике у други слој.

Подешавања » Листа за штампање служи за едитовање листе ученика за штампање. Подешавања » Уређивач базе отвара једноставну апликацију за манипулацију базама података, чије су могућности сведене на основне – прављење нове базе, отварање и измена постојеће и чување промена.

2.1.5 Линија алата

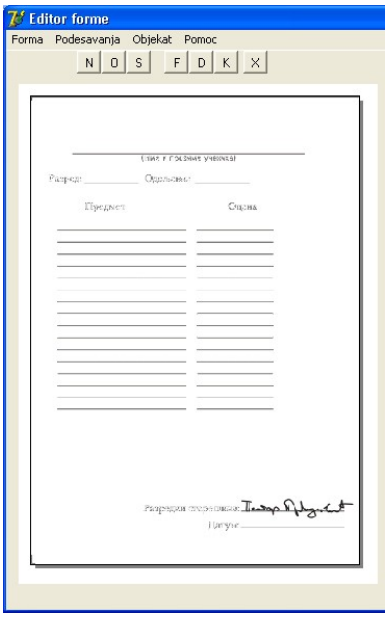
Садржи следеће пречице: Нова форма, Отвори форму, Сачувај форму, Додај статичан објекат, Додај динамичан објекат, Додај функцију, Обриши изабрани објекат. Представљене су SpeedButton контролама чије су сличице додате променом Glyph особине. Десно од њих налази се и Едит контрола у коју је потребно уписати корак општег броја.



Слика 3. Линија алата

2.1.6 Приказ форме

Омогућава кориснику манипулацију објектима на форми и њен изглед. За приказивање објеката користе се TLabel контроле које се постављају на своје X и Y координате и приказују име објекта који представљају или текст који садрже, ако представљају статичне објекте. Текст је форматиран као текст објекта који контрола приказује. Подлога је приказана као правоугани TShape објекат беле боје на коме се налази други димензионисан пропорционално димензијама A4 формата. Преко другог се налазе два TImage објекта истих димензија као и он. Први садржи подлогу, која се учитава преко ставке на главном менију, док други садржи додатну графику, која се на исти начин учитава, али му је и подешена особина Transparent, тако да ће сви пиксели обојени бојом као пиксел у доњем левом углу слике бити провидни.



Слика 3. Приказ форме

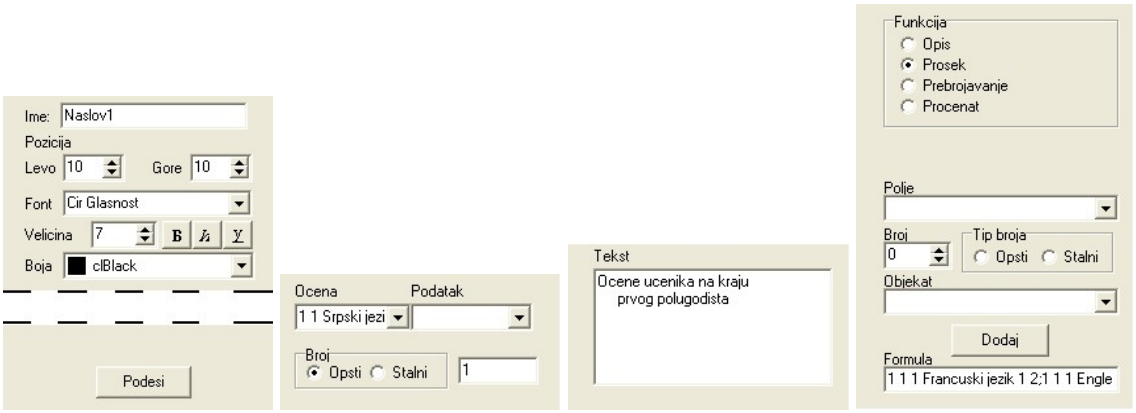
На слици је приказан део прозора за приказ форме. Формулар се налази у првом слоју, и он се неће штампати. Потпис у доњем десном углу је у другом слоју и он ће бити одштампан.

2.1.7 Особине објекта

Приказује особине објекта – изглед текста и садржај контроле.

Што се тиче изгледа текста подржава стандардне особине: промена фонта, величине, боје, стила. На сликама испод приказан је део форме са контролама за изглед текста који се увек налази на форми и скупови контрола који су приказани по потреби, зависно од типа објекта.

Објекти зависно од типа имају и своје особине. Као што видимо на сликама испод, сваки од три врсте објеката има део особина специфичан за своју врсту.



Слика 4. Изглед контрола за подешавање општих особина, специфично подешавање константних контрола, специфично подешавање динамичних контрола, специфично подешавање функцијских контрола

Што се тиче визуелног дизајна програма напоменимо да је омогућено приказивање, тј. сакривање одређених скупова контрола (пример - када се приказује константни објекат контроле за специфичне особине функција и динамичних објеката су невидљиве). Подешавање видљивости контрола уређено је коришћењем три процедуре: PodesiKonstantneKontrole, PodesiFunkcijeKontrole и Podesi DinamicneKontrole. Испод је дат листинг прве процедуре као пример.

```
procedure TPrinter.PodesiKonstantneKontrole();
begin
  if Konstantne then
    begin
```

```
      TxtStaticnogObjekta.Visible:=True;
      PodesiDinamicneKontrole;
      PodesiFunkcijeKontrole;
    end
    else TxtStaticnogObjekta.Visible:=False;
end;
```

Такође манипулацију видљивости контрола имају и неколико Radio дугмади којима се одређују неке од особина, рецимо да ли се користи општи или константни број.

2.1.8 Листа објеката

Садржи имена свих објеката који су убачени. Представљена је ListBox контролом. Поред ње користе се и два низа, један који чува Label компоненте за гарфички приказ, и други који чува додатне информације везане за опис функција и поља динамичких објеката.

2.1.8.1 Записивање особина објеката

Графичке особине објекта чуваће се у својствима лабеле која их представља. Додатне особине биће смештене у паралелни низ и уписане у стринг, док ће се у листи налазити име објекта у облику БројИндексаУНизу ТипОбјекта: Име Објекта.

Додатне особине укључују садржај објекта зависно од типа. Ако је објекат статичан биће садржан само текст који објекат исписује. Ако је динамичан објекат потребно је да се упамти поље у бази из којег се чита податак, тип броја (општи-константни) и број (ако се памти општи онда се уписује број који се додаје општем броју). Ако је функција потребно је навести име функције и атрибуте.

Размотримо елементе у листи: ТипОбјекта представљен је једним од слова K, D, F која означавају да је објекат статичан, динамичан или функција. ИмеОбјекта биће име објекта које се приказује у листи и на лабели, док је ИндексЛабеле интегер вредност која представља индекс лабеле у низу која служи за представљање објекта.

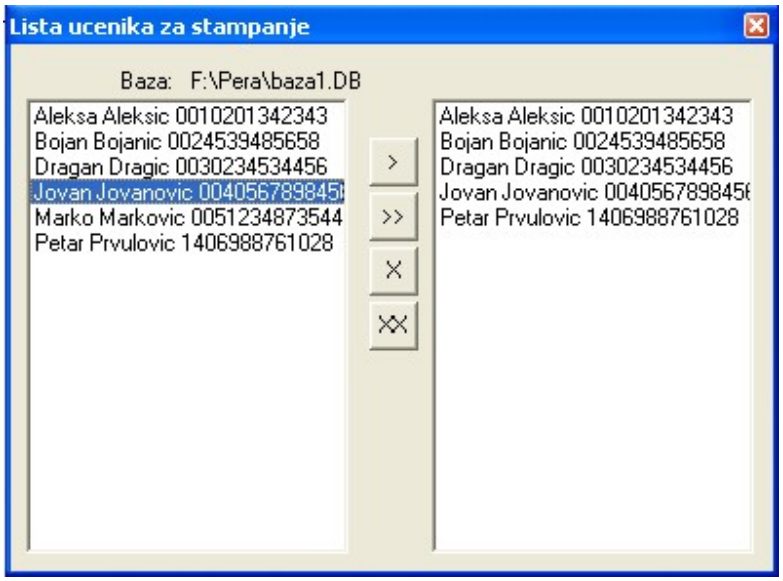
Током развоја апликације јавио се и следећи проблем: како разликовати да ли особине објекта треба сачувати као нов објекат или променити особине неког већ постојећег. Овај проблем решен је једноставно додавањем две интегер променљиве које ће чувати индекс објекта који је истовремено индекс лабеле која представља објекат и индекс у листи објеката, у који ће промене и бити сачуване. Брисањем објекта лабели се поставља својство Visible на False и брише се из листе ставка која садржи име тог објекта.

Приликом прављења новог објекта потребно је у процедурама подесити видљивост потребних контрола и подесити фокус на ListBox1.Items.Count + 1 да би се овај објекат додао на крај листе, док ће фокус објекта за листу са лабелама бити индекс прве лабеле у низу којој је својство Visible подешено на False. Ово је изведено простим пролазом for петљом кроз низ.

Током мењања особина постојећег објекта, приликом бирања објекта на листи хвата се акција OnSelect и подешава фокус на свој индекс, а фокус низа на број садржан у опису објекта у листи. Приликом чувања особина потребно је додати и променљиву која ће чувати информацију о типу објекта, да би се омогућило подешавање контрола и начина записивања особина.

2.1.9 Листа ученика из базе за штампање

Пре штампања неопходно је повезати форму са базом података из које ће се подаци штампати. Ово је могуће урадити бирањем ставке на главном менију Подешавања » Повежи са базом. Отвориће се стандардни Open дијалог у коме је потребно наћи базу. Затим треба направити листу ученика чији ће се подаци узимати у обзир и које ће бројач укључити у бројање. Ово се ради бирањем са менија Подешавања » Листа за штампање. Отвара се дијалог у коме се налазе две листе, једна попуњена ученицима у бази, друга празна, коју је потребно попунити и која је у ствари листа за штампање. Дијалог омогућава пребацивње обележених ставки у листу или свих ставки као и брисање обележених или свих ставки из листе.



Слика 5. Дијалог за уређивање листе ученика

2.1.10 Штампање

2.1.10.1 Приказ штампања

Врло битна могућност је и приказивање страна пре штампања, да би се избегле могуће грешке настале у раду. Апликација омогућава приказивање свих страна посебно, при чему ће бити приказана по једна на екрану, а помоћу контролних тастера могуће је мењати страну која се приказује. У овом ступњу рада програма генеришу се све стране за штампање, при чему се у Caption особине лабела уписују вредности из базе и резултати функција, па се оне приказују у форми дијалога за приказ штампања сразмерно мање негоу природи, одштампане. Пошто се стране приказују какве ће изгледати кад буду одштампане, укључено је и приказивање подлоге. Ово се може искључити одчекирањем опције Прикажи подлогу.

2.1.10.2 Подешавање штампања

Апликација је подешена тако да штампа само у А4 формату, пошто циљни корисник тренутно нема могућности за штампање већих формата (које захтева једино упис у матичне књиге, који је А3), док се већина докумената ради на А4, а мали број на А6 (оцене са тромесечја), при чему се и то штампа по неколико примерака на једном А4 листу па се после сече. Што се тиче програмског дела, приликом штампања потребно је одредити резолуцију штампања и маргине листа које зависе од штампача, и на основу њих подесити позиције објеката, који се штампају стандардним Delphi процедурама садржаним у Printer модулу, при чему се користи Printer.Canvas.

2.2 Апликација за управљање базом података

Направљена је класичним чаробњаком уграђеним у Delphi и садржи стандардне могућности – отварање нове базе и измена постојећих. Што се тиче организације базе, база садржи једну табелу, чија су поља подаци ученика, напомене, оцена из владања, изостанци и оцене из свх предмета редом за све четири године, по четири оцене годишње, за свако тромесечје. Приликом уписивања оцена поља предмета које ученик нема треба оставити празна, да их едитор не би узимао у обзир, док се на место неоцењених предмета уписује 0.

3. Резултати истраживања и дискусија

Као резултат добијен је програм задовољавајућих могућности и интерфејса који пружа сасвим довољну (не малу) слободу дизајна форме, док је истовремено и лак за коришћење. Сагледајмо на крају могућности које програм пружа кориснику.

Кључни део програма садржан је у едитору, у коме корисник (разредни старешина) прави потребну форму за штампање - при чему је омогућено чување те форме и њено поновно учитавање, и позивање осталих елемената програма. Поред едитора, кориснику је дата могућност повезивања изабране форме са базом података ученика одељења, измена података у бази и прављење листе ученика из те базе чије податке је потребно штампати, као и штампање. Могућност серијског штампања, тј. штампање низа форми попуњених редом подацима ученика изабраних из базе је истовремено и главна карактеристика програма, односно циљ овог рада.

4. Закључак

Осврнимо се на крају на садржај овог рада – у краћим цртама приказани су неки основни концепти потребни за развој апликације ове намене, као и њених могућности. Управо због слабе софтверске подршке на овом пољу надања су аутора да ће један овакав пројекат, можда не толико обиман колико би требало да буде, али у сваком случају ништа мање користан, послужити као покретач и идеја водиља другим ауторима за израду апликација сличне намене. Што се будућности програма тиче, у плану је његово даље развијање у виду повећања могућности и побољшања рада самог програма, као и евентуалних естетских модификација које би допринеле угоднијем коришћењу програма. Даљи развој пројекта биће праћен на адреси www.peruzzi.dzaba.com/project1.htm, где ће се налазити разне информације о програму, нове додате могућности као и примерак програма за download.

5. Литература

[1.] delphi.about.com

[2.] Mastering Delphi 5 – Marco Cantu

[3.] Delphi Database Application Developer’s Guide