

Personalizirani pismeni ispit

ČengiĆ, Marijan

Undergraduate thesis / Završni rad

2016

Degree Grantor / Ustanova koja je dodijelila akademski / stručni stupanj: **Josip Juraj Strossmayer University of Osijek, Faculty of Electrical Engineering, Computer Science and Information Technology Osijek / Sveučilište Josipa Jurja Strossmayera u Osijeku, Fakultet elektrotehnike, računarstva i informacijskih tehnologija Osijek**

Permanent link / Trajna poveznica: <https://um.nsk.hr/um:nbn:hr:200:189804>

Rights / Prava: [In copyright](#) / [Zaštićeno autorskim pravom.](#)

Download date / Datum preuzimanja: **2024-04-25**

Repository / Repozitorij:

[Faculty of Electrical Engineering, Computer Science and Information Technology Osijek](#)



SVEUČILIŠTE JOSIPA JURJA STROSSMAYERA U OSIJEKU
ELEKTROTEHNIČKI FAKULTET

Preddiplomski stručni studij, smjer informatika

Personalizirani pismeni ispit

Završni rad

Marijan Čengić

Osijek, 2016.

Obrazac za ocjenu završnog rada na stručnom/preddiplomskom studiju.

Izjava o originalnosti završnog rada.

Sadržaj

1.	UVOD	1
1.1.	Zadatak završnog rada	1
2.	PRIMJENJENE TEHNOLOGIJE I ALATI	2
2.1.	C# programsko okruženje.....	2
2.2.	Objektno orijentirano programiranje	2
2.3.	XML uređivanje teksta	4
3.	REALIZACIJA APLIKACIJE ZA PERSONALIZIRANI PISMENI ISPIT	5
3.1.	CSV baza podataka.....	5
3.2.	Arhitektura sustava	5
3.2.1.	Baza podataka.....	7
3.2.2.	Fotografije studenata	8
3.2.3.	Podatci u template-u	8
3.3.	C# aplikacija za personalizirani ispit.....	10
3.3.1.	Dohvaćanje podataka i rezultati	10
3.3.2.	Inicijaliziranje podataka o studentu	11
3.3.3.	Spremanje kao Word dokument	11
3.3.4.	Umetanje slika i podataka u tablicu.....	12
3.4.	Primjer izrade ispita.....	13
3.4.1.	Primjer template-a	13
3.4.2.	Izgled pismenog ispita kao rezultat aplikacije.....	14
4.	ZAKLJUČAK	15
	LITERATURA.....	16
	SAŽETAK.....	17
	ABSTRACT	18
	ŽIVOTOPIS	19
	PRILOZI.....	20

1. UVOD

Cilj ovog završnog rada je, kao što i samo ime kaže, personalizirati pismeni ispit, odnosno pojednostavljeni način pripreme profesorima i nastavnicima za pismeni ispit. Posao nastavnika iliti profesora biti će samo da ispiše pitanja za pismeni ispit, dok će se ostali podatci o studentima sami umetati u dokument. Program koristi *template* pismenog ispita i u njega ubacuje podatke studenata koji su prijavili ispit. Objasnit će se problemi na koje se nailazi tokom rada te kroz opis rada razmotrit će se i način rješavanja istih.

1.1. Zadatak završnog rada

Zadatak ovog završnog rada je da uz pomoć već definirane baze podataka, u kojoj se nalaze podatci studenata i njihove slike, sastavimo pismeni ispit koji će biti personaliziran osobno za svakog studenta. Koristimo template ispita koji će samo zahtijevati unos ispitnih pitanja, dok ostatak unosa kao imena, prezimena, slike i jmbag-ove studenata unosi program.

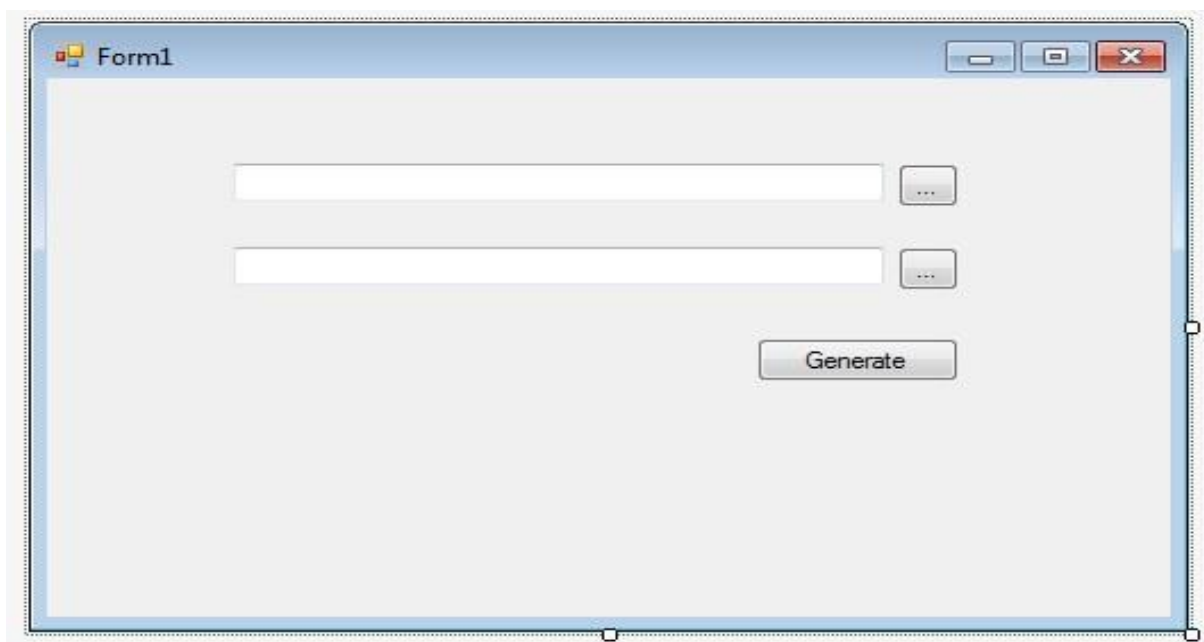
2. PRIMJENJENE TEHNOLOGIJE I ALATI

2.1. C# programsko okruženje

C# (*eng. C sharp*) je objektno orijentirani program. Za svoje prethodnike ima C i C++ od kojih je preuzeo sve kvalitete koje nisu zahtijevale poboljšanja. Izrazi, naredbe i skoro cijela sintaksa, što čini većinu tipičnih programa, je ostala nepromijenjena. Jezik je opće primjene i namijenjen je izradi aplikacija za .NET Framework platformu. C# je programski jezik koji nam omogućuje pisanje, kako konzolnih, tako i Windows aplikacija, pa čak i aplikacija za web (slika 2.1).

2.2. Objektno orijentirano programiranje

OOP (*eng. Object-oriented programming*) je način programiranja u kojem vlada koncept stvaranja objekata. Svaki objekt može imati svoje članove i funkcije koje možemo pozvati nad tim objektom. Funkcije koje pozivamo nad nekim objektom zovemo metode. Objektno orijentirani pristup omogućuje pisanje koda koji je lako izmjenjiv i nadogradiv te ga je moguće ponovno koristiti. Ovakav pristup programiranju donosi neka važna svojstva koja danas smatramo osnovnim načelima OOP-a (slika 2.2).



Slika 2.1 Dizajn forme pri stvaranju aplikacije

```

Program.cs  Form1.cs*  Form1.cs [Design]*
GenerateDocx.Form1  Form1()

using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
//using System.Threading.Tasks;
using System.Windows.Forms;
using Microsoft.Office.Interop.Word;
using System.IO;

namespace GenerateDocx
{
    public partial class Form1 : Form
    {
        public Form1()
        {

```

Slika 2.2 Izgled kodnog sučelja aplikacije

2.3. XML uređivanje teksta

XML (eng. *Extensible Markup Language*), odnosno jezik za označavanje podataka. Format oznaka u XML-u vrlo je sličan formatu oznaka u HTML (eng. *HyperText Markup Language*) jeziku. Danas je XML jezik vrlo raširen i koristi se za različite namjene kao što su: odvajanje podataka od prezentacije, razmjenu podataka, pohranu podataka, povećavanje dostupnosti podataka i izradu novih specijaliziranih jezika za označavanje. XML je standardizirani jezik. On objedinjuje jednostavnost HTML-a i izražajnu snagu SGML-a (eng. *Standard Generalized Markup Language*). Odredilo se 10 ciljeva kojih su se u razvoju trudili pridržavati:

1. XML mora biti izravno primjenjiv preko interneta.
2. XML mora podržavati širok spektar primjena.
3. XML mora biti kompatibilan s SGML-om.
4. Moraju se pisati programi koji lako procesiraju XML dokumente.
5. Broj opcionalnih svojstva u XML-u mora biti apsolutno minimalan, u idealnom slučaju jednak nuli.
6. XML dokumenti moraju biti čitljivi ljudima te u razumnoj mjeri jednostavni
7. Standard mora biti specificiran što prije
8. Dizajn XML-a mora biti formalan i precizan
9. Kreiranje XML dokumenata mora biti jednostavno
10. Sažetost kod označavanja dokumenta XML-om je od minimalnog značaja

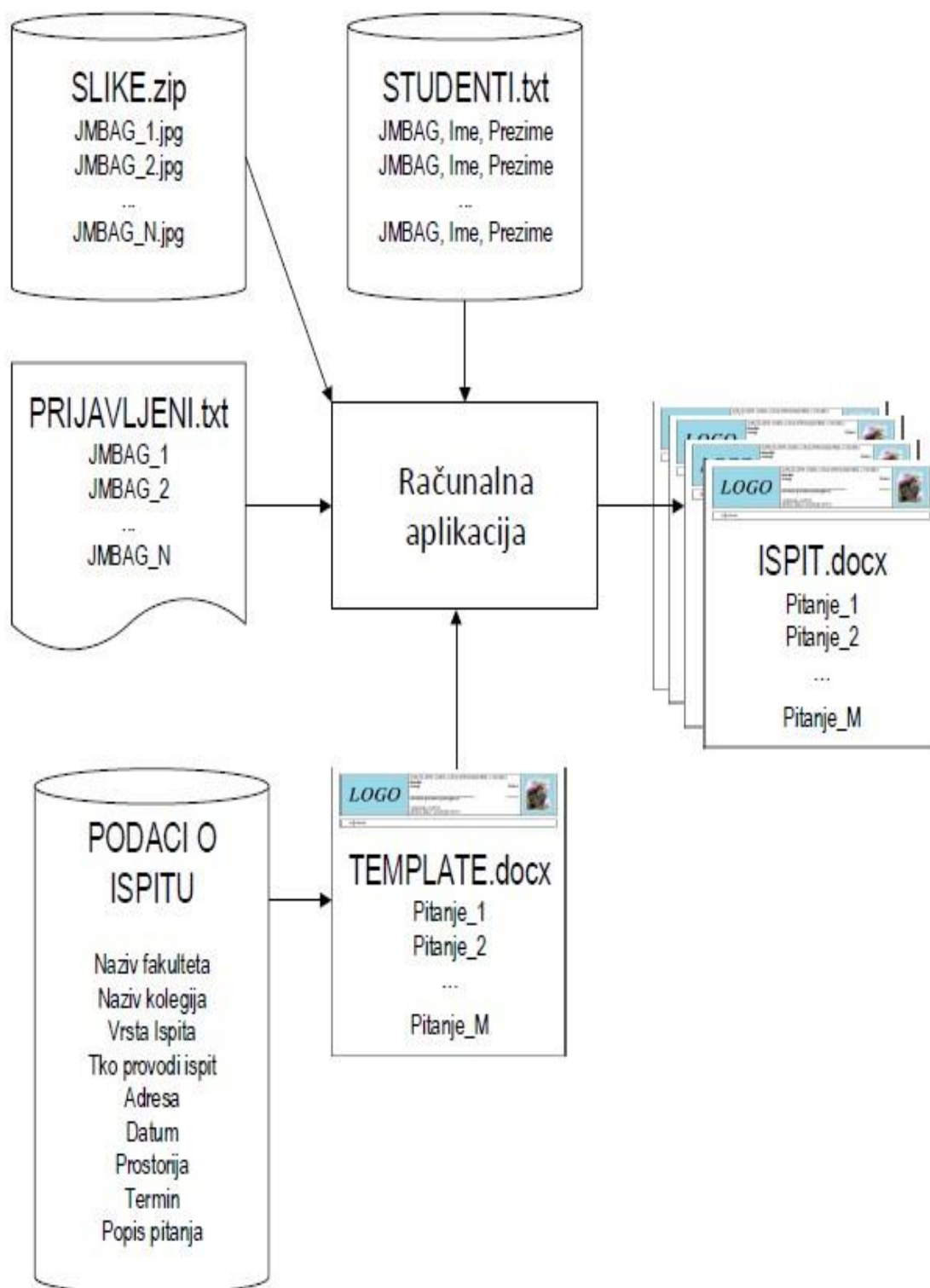
3. REALIZACIJA APLIKACIJE ZA PERSONALIZIRANI PISMENI ISPIT

3.1. CSV baza podataka

Skraćenica CSV (*eng. Comma-Separated Values*) u prijevodu znači „zarezom odvojene vrijednosti“, odnosno u novije vrijeme „znakom odvojene vrijednosti“ (*eng. comma-separated values / character-separated values*). Takva datoteka sadrži tablične podatke u obliku običnog teksta. CSV datoteke se koriste za razmjenu podataka između baza podataka različitih arhitektura. Ujedno, najveća prednost CSV datoteke je čitljivost podataka tekstualnim preglednikom koji omogućava pojedincu da vizualno analizira podatke koji se u njoj nalaze. Također se mogu odrediti karakteristike pojedinih zapisa koje su potrebne za daljnji rad. Pod karakteristike podatka misli se na to je li je podatak cijeli broj, realni broj, nekakav tekstualni zapis i sl.

3.2. Arhitektura sustava

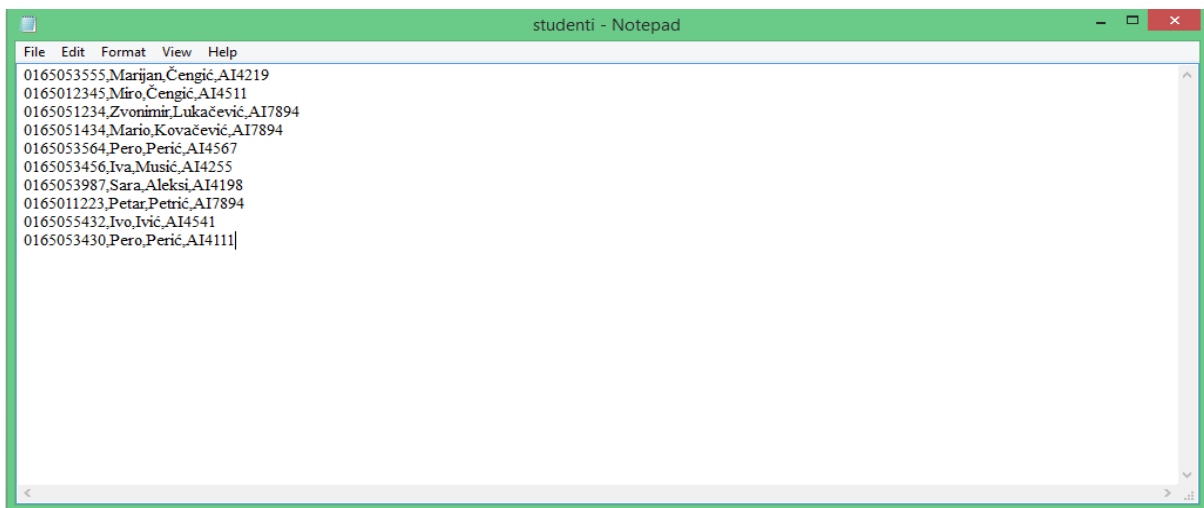
Sustav se sastoji od više dijelova (slika 3.1): template-a u kojemu se nalaze podatci o ispitu (naziv fakulteta, naziv kolegija, vrsta ispita, adresa, datum, pitanja), fotografije svakog studenta u .jpg formatu, .txt datoteka sa podacima o studentu (jmbag, ime, prezime) i drugom .txt datotekom koja sadrži jmbag-ove studenata koji su prijavili pismeni ispit. Glavna poveznica bi bila računalna aplikacija koja sve nabrojene podatke spaja u .docx dokument koji je onda personaliziran za svakoga studenta osobno te spreman za printanje.



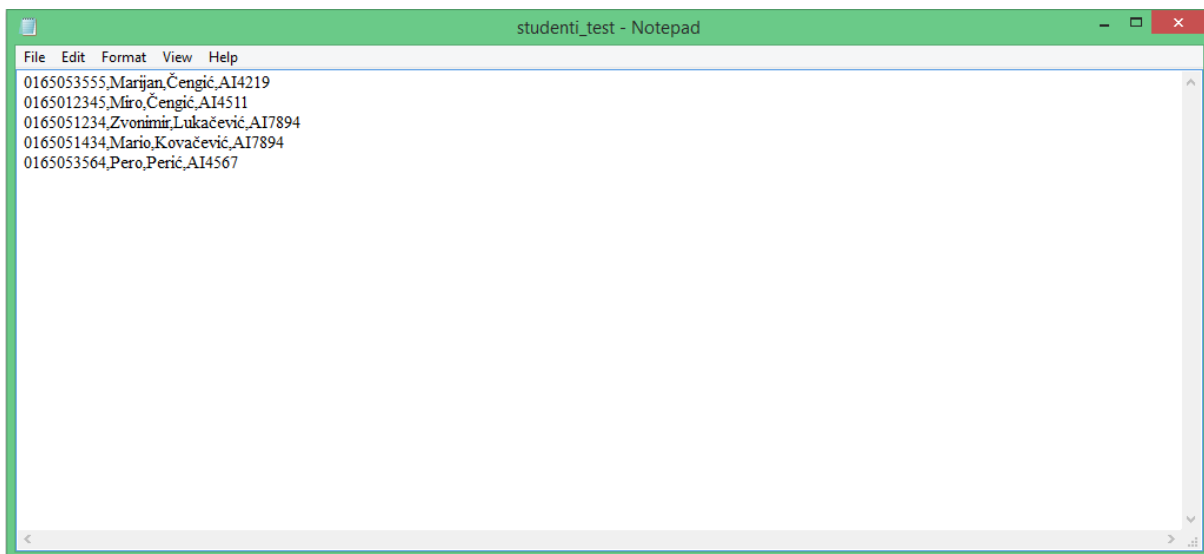
Slika 3.1 Arhitektura sustava

3.2.1. Baza podataka

Za bazu podataka korisimo CSV datoteku studenti.txt u kojoj se nalazi popis svih studenata zajedno sa svojim podacima, koji mogu prijaviti pismeni ispit (slika 3.2). Uz studenti.txt koristimo još jednu CSV datoteku, a to je studenti_test.txt u kojoj se nalazi popis studenata koji su prijavili taj pismeni ispit (slika 3.3).



Slika 3.2 Popis studenata



Slika 3.3 Popis studenata koji su prijavili pismeni ispit

3.2.2. Fotografije studenata

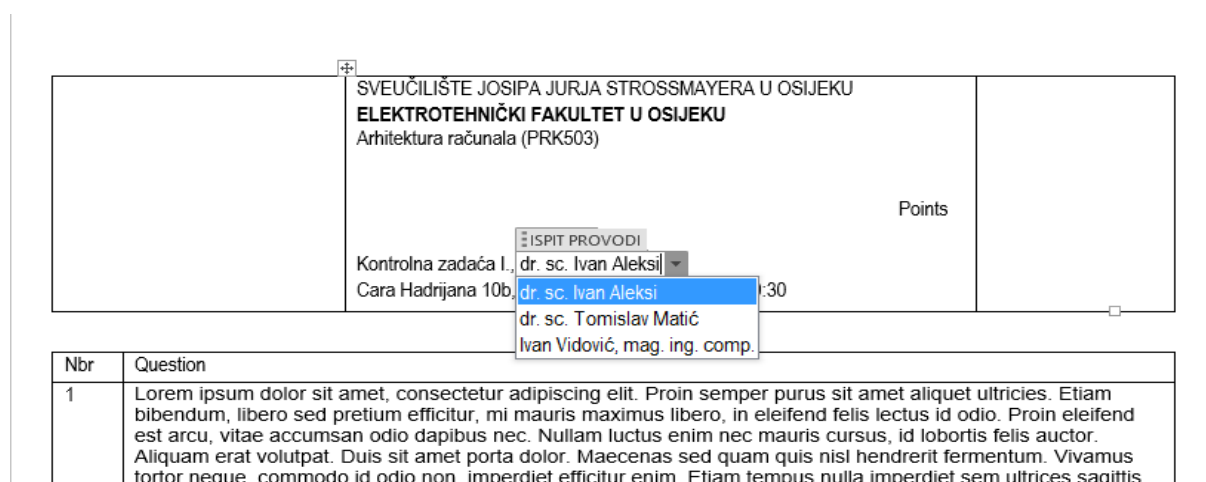
Svaki student ima svoju fotografiju, koja je povezana sa studentom preko njegovog jmbag-a.



Slika 3.4 Fotografije povezane jmbag-om za svakog studenta

3.2.3. Podatci u template-u

U template-u se, uz ispitna pitanja, nalaze i dodatne informacije kao što su naziv kolegija (slika 3.6), vrsta pismenog ispita, tko provodi ispit (slika 3.5), adresa (slika 3.8), datum (slika 3.7), učionica, vrijeme početka i kraja ispita. Sve te informacije se mogu izmjenjivati sa jednim klikom miša, jer se koristi padajuća lista koja omogućuje odabir između više ponuđenih opcija. Tu opciju smo omogućili preko developer kartice u Word-u koja nam dopušta ubacivanje dizajnerskih opcija u dokument.



Slika 3.5 Izbor profesora koji provodi ispit u template-u

	SVEUČILIŠTE JOSIPA JURJA STROSSMAYERA U OSIJEKU TEHNIČKI FAKULTET U OSIJEKU Arhitektura računalna (PRK503)	Points
	Arhitektura računalnih sustava (SARIE301)	
Kontrolna zadaća I., dr. sc. Ivan Aleksi Cara Hadrijana 10b, 25.2.2015., K2-1, od 8:00 do 9:30		

Nbr	Question
1	Lorem ipsum dolor sit amet, consectetur adipiscing elit. Proin semper purus sit amet aliquet ultricies. Etiam bibendum, libero sed pretium efficitur, mi mauris maximus libero, in eleifend felis lectus id odio. Proin eleifend est arcu, vitae accumsan odio danihus nec. Nullam luctus enim nec mauris cursus, id lobortis felis auctor

Slika 3.6 Odabir kolegija koji provodi ispit

	SVEUČILIŠTE JOSIPA JURJA STROSSMAYERA U OSIJEKU ELEKTROTEHNIČKI FAKULTET U OSIJEKU Arhitektura računalna (PRK503)	Points
	Kontrolna zadaća I., DATUM in Aleksi Cara Hadrijana 10b, 25.2.2015., od 8:00 do 9:30	

Nbr	Question
1	Lorem ipsum dolor sit amet, consectetur adipiscing elit. Proin semper purus sit amet aliquet ultricies. Etiam bibendum, libero sed pretium efficitur, mi mauris maximus libero, in eleifend felis lectus id odio. Proin eleifend est arcu, vitae accumsan odio danihus nec. Nullam luctus enim nec mauris cursus, id lobortis felis auctor.
2	Phasellus sit amet vehicula tellus. Duis sit amet metus arcu. Fusce ac nibh aliquam. tincidunt urna id. tincidunt

Slika 3.7 Odabir datuma pisanja ispita

SVEUČILIŠTE JOSIPA JURJA STROSSMAYERA U OSIJEKU ELEKTROTEHNIČKI FAKULTET U OSIJEKU Arhitektura računalā (PRK503)		Points
ADRESA zadaća I., dr. sc. Ivan Aleksi Cara Hadrijana 10b, 2.2.2015., K2-1, od 8:00 do 9:30 Kneza Trpimira 2b Cara Hadrijana 10b		

Nbr	Question
1	Lorem ipsum dolor sit amet, consectetur adipiscing elit. Proin semper purus sit amet aliquet ultricies. Etiam bibendum, libero sed pretium efficitur, mi mauris maximus libero, in eleifend felis lectus id odio. Proin eleifend est arcu, vitae accumsan odio dapibus nec. Nullam luctus enim nec mauris cursus, id lobortis felis auctor. Aliquam erat volutpat. Duis sit amet porta dolor. Maecenas sed quam quis nisl hendrerit fermentum. Vivamus

Slika 3.8 Odabir adrese provođenja ispita

3.3. C# aplikacija za personalizirani ispit

3.3.1. Dohvaćanje podataka i rezultati

Primarna radnja u programu zasigurno je ispisivanje naredbe kojom želimo povezati naš program sa .txt dokumentima, odnosno traženim podacima. Ti podatci su zapravo osobne informacije svakog pojedinog studenta koje želimo ubaciti u template ispita. Nakon toga, pripremljene ispite spremamo u mapu Result (slika 3.2).

```

namespace GenerateDocx
{
    public partial class Form1 : Form
    {
        public Form1()
        {
            InitializeComponent();

            m_txtTextFilePath.Text = System.IO.Directory.GetCurrentDirectory() + "\\studenti.txt";
            this.m_txtWordFilePath.Text = System.IO.Directory.GetCurrentDirectory() + "\\Result";
        }

        private void m_btnGenerate_Click(object sender, EventArgs e)
        {
            /*
            if( this.m_txtWordFilePath.Text == "" || this.m_txtTextFilePath.Text == "" )
            {
                MessageBox.Show("Please input the Text file path and folder for Docx correctly.");
                return;
            }
            */
        }
    }
}

```

Slika 3.2 Dohvaćanje podataka i spremanje rezultata

3.3.2. Inicijaliziranje podataka o studentu

Varijabla tipa string označava da se koristimo znakovnim nizovima promjenjive duljine. Također koristimo jednodimenzionalno polje kojim inicijaliziramo podatke o studentu. Obrađene podatke o studentu kao i njihove fotografije, logo fakulteta i template ispita spajamo u jedno (slika 3.3).

```
string[] allLines = File.ReadAllLines(m_txtFilePath.Text);

var query = from line in allLines
    let data = line.Split(',')
    select new
    {
        Jmbag = data[0],
        FirstName = data[1],
        SecondName = data[2],
        IndexNumebr = data[3]
    };

string strDocFileName = "";
string strTemplateFileName = System.IO.Directory.GetCurrentDirectory() + "\\template.docx";
string strLogoImageFileName = System.IO.Directory.GetCurrentDirectory() + "\\logo.jpg";
string strPictureImageFolder = System.IO.Directory.GetCurrentDirectory() + "\\Pictures";
string strPictureImageFileName = "";

object missing = System.Reflection.Missing.Value;
object readOnly = false;
object isVisible = false;
```

Slika 3.3 Inicijalizacija podataka

3.3.3. Spremanje kao Word dokument

Dio koda koji govori da gotovi ispiti budu spremljeni kao Word dokument (slika 3.4).

```
//Save as Docx
Microsoft.Office.Interop.Word.Application appWord = new Microsoft.Office.Interop.Word.Application();
Microsoft.Office.Interop.Word.Document wordDocument = null;
wordDocument = appWord.Documents.Open(strTemplateFileName, Format: WdOpenFormat.wdOpenFormatAuto);
```

Slika 3.4 Spremanje rezultata kao .docx dokument

3.3.4. Umetanje slika i podataka u tablicu

Iz slike 3.5 vidimo proces inicijalizacije tablice te raspoređivanje slika i podataka po tablici. Kao dodatak tome u komentaru je dodana i mogućnost ubacivanja pitanja u drugu tablicu template-a u kojoj se nalaze ispitna pitanja, što znači da ispitna pitanja mogu, a i ne moraju već biti u template-u.

```
int nWordCount = wordDocument.Words.Count;
int nTableCount = wordDocument.Tables.Count;

Table vTable1 = wordDocument.Tables[1];

foreach ( var vItem in query)
{
    Cell vLogoCell = vTable1.Cell(1, 1);
    Range vLogoRange = vLogoCell.Range;
    vLogoRange.Delete();
    vLogoRange.InlineShapes.AddPicture(strLogoImageFileName, ref missing, ref missing, ref missing);

    strPictureImageFileName = strPictureImageFolder + "\\\" + vItem.Jmbag + ".jpg";
    Cell vPictureCell = vTable1.Cell(1, 3);
    Range vPictureRange = vPictureCell.Range;
    vPictureRange.Delete();
    vPictureRange.InlineShapes.AddPicture(strPictureImageFileName, ref missing, ref missing, ref missing);

    Table vTable_1_1 = vTable1.Tables[1];
    Table vTable_1_1_1 = vTable_1_1.Tables[1];

    Cell vFirstNameCell = vTable_1_1_1.Cell(1, 1);
    Range vFirstNameRange = vFirstNameCell.Range;
    vFirstNameRange.Text = vItem.FirstName;

    Cell vSecondNameCell = vTable_1_1_1.Cell(1, 2);
    Range vSecondNameRange = vSecondNameCell.Range;
    vSecondNameRange.Text = vItem.SecondName;

    Cell vJmbagCell = vTable_1_1_1.Cell(1, 3);
    Range vJmbagRange = vJmbagCell.Range;
    vJmbagRange.Text = vItem.Jmbag;

    Table vTable2 = wordDocument.Tables[2];
    for ( int i = 0; i < 5; i ++ )
    {
        if( vTable2.Rows.Count < i + 2 )
            vTable2.Rows.Add();

        Cell vNBrCell = vTable2.Cell(i + 2, 1);
        Range vNBrRange = vNBrCell.Range;
        vNBrRange.Text = String.Format("{0}", i + 1 );

        Cell vQuestionCell = vTable2.Cell(i + 2, 2);
        Range vQuestionRange = vQuestionCell.Range;
        vQuestionRange.Text = "What questions do you have now? " + String.Format(" {0}", i + 1);
    }

    strDocFileName = m_txtWordFilePath.Text + "\\\" + vItem.Jmbag + ".Docx";
    wordDocument.SaveAs2(strDocFileName, WdSaveFormat.wdFormatXMLDocument);
    wordDocument.SaveAs(strDocFileName);
}

wordDocument.Close();
```

Slika 3.5 Ubacivanje slika i podatka u tablicu

3.4. Primjer izrade ispita

3.4.1. Primjer template-a

Primjer template-a (slika 3.6) prije ubacivanja svih već spomenutih podataka studenata te njihovih fotografija.

	+ SVEUCILISTE JOSIPA JURJA STROSSMAYERA U OSIJEKU ELEKTROTEHNIČKI FAKULTET U OSIJEKU Arhitektura računalâ (PRK503) Bodovi Kontrolna zadaća I., dr. sc. Ivan Aleksi Cara Hadrijana 10b, 25.2.2015., K2-1, od 8:00 do 9:30	
--	---	--

Rbr	Pitanja
1.	Lorem ipsum dolor sit amet, consectetur adipiscing elit. Proin semper purus sit amet aliquet ultricies. Etiam bibendum, libero sed pretium efficitur, mi mauris maximus libero, in eleifend felis lectus id odio. Proin eleifend est arcu, vitae accumsan odio dapibus nec. Nullam luctus enim nec mauris cursus, id lobortis felis auctor. Aliquam erat volutpat. Duis sit amet porta dolor. Maecenas sed quam quis nisl hendrerit fermentum. Vivamus tortor neque, commodo id odio non, imperdiet efficitur enim. Etiam tempus nulla imperdiet sem ultrices sagittis. Donec pretium, ante et hendrerit maximus, elit massa consectetur ipsum, at rhoncus nunc est non libero. Maecenas id lectus ut tortor dapibus semper.
2.	Phasellus sit amet vehicula tellus. Duis sit amet metus arcu. Fusce ac nibh aliquam, tincidunt urna id, tincidunt elit. Nulla facilisi. Duis gravida malesuada molestie. Vestibulum gravida dui dui, eu accumsan neque viverra et. Vestibulum viverra bibendum ante convallis hendrerit. Donec dui diam, ornare in dolor ac, gravida feugiat dolor. In eu placerat lectus. Interdum et malesuada fames ac ante ipsum primis in faucibus.
3.	Nullam iaculis vel mi id viverra. Nulla facilisi. Sed et laoreet quam. Proin dapibus libero non enim iaculis

Slika 3.6 Izgled template-a

3.4.2. Izgled pismenog ispita kao rezultat aplikacije

Pismeni ispit nakon što su se obradili svi podatci i umetnuli u template pismenog ispita. Ubacivši sve podatke, dobivamo konačni izgled personaliziranog pismenog ispita (slika 3.7).

	SVEUCILISTE JOSIPA JURJA STROSSMAYERA U OSIJEKU ELEKTROTEHNIČKI FAKULTET U OSIJEKU Arhitektura računalā (PRK503)		
	<u>Petar</u>	<u>Petrić</u>	
Kontrolna zadaća I., dr. sc. Ivan Aleksi Cara Hadrijana 10b, 25.2.2015., K2-1, od 8:00 do 9:30			

Rbr	Pitanja
1.	Lorem ipsum dolor sit amet, consectetur adipiscing elit. Proin semper purus sit amet aliquet ultricies. Etiam bibendum, libero sed pretium efficitur, mi mauris maximus libero, in eleifend felis lectus id odio. Proin eleifend est arcu, vitae accumsan odio dapibus nec. Nullam luctus enim nec mauris cursus, id lobortis felis auctor. Aliquam erat volutpat. Duis sit amet porta dolor. Maecenas sed quam quis nisl hendrerit fermentum. Vivamus tortor neque, commodo id odio non, imperdiet efficitur enim. Etiam tempus nulla imperdiet sem ultrices sagittis. Donec pretium, ante et hendrerit maximus, elit massa consectetur ipsum, at rhoncus nunc est non libero. Maecenas id lectus ut tortor dapibus semper.
2.	Phasellus sit amet vehicula tellus. Duis sit amet metus arcu. Fusce ac nibh aliquam, tincidunt urna id, tincidunt elit. Nulla facilisi. Duis gravida malesuada molestie. Vestibulum gravida dui dui, eu accumsan neque viverra et. Vestibulum viverra bibendum ante convallis hendrerit. Donec dui diam, ornare in dolor ac, gravida feugiat dolor. In eu placerat lectus. Interdum et malesuada fames ac ante ipsum primis in faucibus.
3.	Nullam iaculis vel mi id viverra. Nulla facilisi. Sed et laoreet quam. Proin dapibus libero non enim iaculis

Slika 3.7 Primjer gotovog personaliziranog ispita

4. ZAKLJUČAK

U ovom radu izrađena je aplikacija koja omogućuje olakšanu pripremu nastavniku ili profesoru za pismeni ispit. Sustav se sastoji od same računalne aplikacije, template-a pismenog ispita, fotografija i baze podataka studenata. Aplikacija je realizirana i radi kako je i zamišljena. Prednost ove aplikacije je ubrzan proces pripreme ispita te dodatna personalizacija ispita za svakog studenta osobno. Mana aplikacije je baza podataka - svaki pismeni ispit je povezan sa osobnim podacima studenta kao i njegovom fotografijom. To govori da baza podataka već mora postojati. Da bi cijeli ovaj program imao smisla, najveći dio posla mora se napraviti u početku – to je izrada baze podataka. No isto tako, te podatke potrebno je i ažurirati kako bi uvijek bili točni. Uzevši u obzir sve navedeno, aplikacija za personalizirani pismeni ispit ubrzava proces provođenja pismenih ispita, tako što zahtjeva samo sastavljanje ispitnih pitanja, dok ostatak procesa obavlja program.

LITERATURA

[1] Programski jezik C# & Windows forms, <http://goo.gl/O3GFjj>, lipanj 2016.

[2] C#, <https://goo.gl/vZBQKZ>, lipanj 2016.

[3] Programski jezik C#, <http://goo.gl/vOliXr>, lipanj 2016.

[4] C# programiranje, <http://goo.gl/5ZqMnp>, lipanj 2016.

[5] XML, <https://goo.gl/9kWqD7>, lipanj 2016.

[6] CSV datoteke, <http://goo.gl/OwGolp>, lipanj 2016.

SAŽETAK

Naslov: Personalizirani pismeni ispit

U ovom radu izrađena je aplikacija koja uz pomoć template-a pismenog ispita i baze podataka sa studentima omogućuje izradu pismenog ispita presonaliziranog za svakog studenta osobno. Kroz rad objašnjene su tehnologije i alati koji su korišteni za izradu aplikacije. Nakon opisa tehnologija, objašnjena je realizacija aplikacije te na kraju i primjer izrade ispita.

Ključne riječi: C#, objektno orijentirano programiranje, CSV baza podataka, XML uređivanje teksta, aplikacija, programiranje

ABSTRACT

Title: Personalized writing exam

In this work application was made that uses template of an exam and data base with students to make a personalized exam of every student that is taking exam. Trough this paper it's explained what technologies and tool were used to make application. After decription of technologies, the realization was briefly explained and on the end there is a example of finished exam.

Keywords: C#, object-oriented programming, CSV data base, XML text editor, application, programing

ŽIVOTOPIS

Marijan Čengiđ rođen 25. prosinca 1993. godine u Zagrebu. Od svoga rođnja živi u mjestu Garešnički Brestovac koji spada pod grad Garešnicu. U 2008. godini upisuje se u Tehničku školu Daruvar, gdje se prvi put susreće sa programiranjem. Srednju školu završava redovno u 2012. godini te iste godine upisuje stručni studij, smjer informatika na Elektrotehničkom fakultetu Osijek, i završava ga 2016. godine nakon četiri godine studiranja.

PRILOZI

Kompleti izvorni kod aplikacije kao i svi ostali materijali nalaze se na CD-u priloženom uz ovaj rad.