



SISTEM ZA UPRAVLJANJE RADOM PRIVATNE ŠKOLE RAZVIJEN U DESKTOP I VEB TEHNOLOGIJI

PRIVATE SCHOOL WORK MANAGEMENT SYSTEM DEVELOPED IN DESKTOP AND WEB TECHNOLOGY

Milica Terzić, *Fakultet tehničkih nauka, Novi Sad*

Oblast – INŽENJERSTVO INFORMACIONIH SISTEMA

Kratak sadržaj – U okviru ovog rada izvršen je razvoj desktop i veb aplikacije čija je namena olakšan i ubrzan proces poslovanja privatne škole. Prikazan je postupak implementacije aplikacije u desktop WPF tehnologiji i veb tehnologiji, ASP.NET MVC, koja predstavlja sastavni deo Microsoft-ovog .NET okruženja.

Ključne reči: Automatizacija, ASP.NET, MVC, Privatna škola

Abstract – This paper gives an overview of the developing desktop and web application whose purpose is to manage the work of a private school. Implementation of the application is developed using WPF technology and ASP.NET MVC technology, which is part of .NET framework developed by Microsoft company.

Keywords – Automation, WPF, ASP.NET, MVC, Private school

1. UVOD

U modernom svetu koji se neprestano menja neophodno je ići u korak sa vremenom i stalno usvajati nova znanja. Privatna škola svakako predstavlja inspirativnu sredinu za mlade koji su ambiciozni i željni novih informacija i znanja. Kroz spoj savremenih metodičkih trendova, nastavnog programa i iskusnog nastavnog kadra polaznici na kreativan i moderan način mogu da savladaju razne vrste kurseva koji se nude u više nivoa znanja, zavisno od pripremljenosti samih učenika.

Postojanje softvera je veoma bitno za pružanje pomoći organizacijama u upravljanju podacima, resursima i performansama. Kako bi jedan poslovni sistem kao što je škola olakšano obavljao svakodnevne zadatke, pratio uspešnost poslovanja i rad svojih zaposlenih, neophodno je postojanje softvera koji će omogućiti automatizaciju poslovanja. U okviru ovog rada izvršen je razvoj desktop i veb aplikacije koja ima za cilj olakšan rad jedne privatne škole, u smislu lakšeg praćenja načina poslovanja i organizacije rada privatne škole.

NAPOMENA:

Ovaj rad proistekao je iz master rada čiji mentor je bio dr Darko Stefanović, vanr.prof.

Cilj ovog rada jeste objašnjenje korišćenih tehnologija u svrhu razvoja odgovarajućeg softvera za jedan poslovni sistem – privatnu školu, kao i opis funkcionalnosti razvijene desktop i veb aplikacije.

2. OPIS KORIŠĆENIH TEHNOLOGIJA

Aplikacija za automatizaciju rada privatne škole, razvijana je korišćenjem sledećih tehnologija:

- Microsoft SQL Server
- Entity Framework
- WPF
- ASP.NET MVC

koje će biti opisane u nastavku rada.

2.1. Microsoft SQL Server

Microsoft SQL Server je sistem za upravljanje relacionom bazom podataka, razvijen od strane Microsoft kompanije. Predstavlja softver čija je primarna funkcija da skladišti i vraća podatke na zahtev drugih softverskih aplikacija.

2.2. Entity Framework

Entity Framework predstavlja Microsoft-ovu tehnologiju i alat .NET Framework-a, koji omogućava pristupanje relacionoj bazi podataka pomoću objektno orijentisanih koncepata. Jezgro Entity Framework-a je model podataka (engl. Entity Data Model - EDM) koji omogućava prilagodavanje preslikavanja između objekta klase i konkretne tabele iz baze podataka. Postoji tri načina za definisanje EDM-a:

- Baza – prvo (engl. Database first)
- Kôd– prvo (engl. Code first)
- Model – prvo (engl. Model first)

Database first, koji je korišćen u ovom radu, automatski generiše sve neophodne objekte u kodu, na osnovu modela baze podataka odnosno odgovarajućih SQL (Structured Query Language) tabela. EDM će biti kreiran iz šeme baze podataka. Ovaj pristup je najpogodniji za aplikacije koje koriste već postojeću bazu podataka.

2.3. WPF

WPF(Windows Presentation Foundation) predstavlja korisnički okvir(engl. User Interface - UI) koji kreira aplikacije za desktop računare. Razvojna platforma WPF podržava širok skup funkcija za razvoj aplikacija, uključujući aplikativni model, resurse, kontrole, grafiku,

izgled, vezivanje podataka, dokumente i sigurnost. Predstavlja podskup .NET platforme. WPF koristi XAML (*Extensible Application Markup Language*) kako bi se obezbedio deklarativni model za programiranje aplikacija.

2.4. ASP.NET MVC

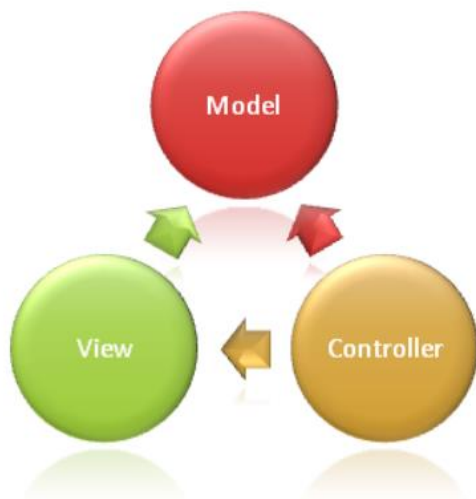
ASP.NET (*Active Server Pages .NET*) je veb tehnologija razvijena od strane *Microsoft* kompanije, koja omogućava programerima da razvijaju dinamičke veb sajtove, veb aplikacije i veb servise.

MVC (*Model-View-Controller*) je arhitekturni obrazac koji deli aplikaciju na tri međusobno povezana dela kako bi se odvojile interne reprezentacije informacija od načina na koji su informacije predstavljene korisnicima.

MVC obrazac čine tri glavne logičke komponente:

1. **Model** sadrži skup klasa koje predstavljaju poslovnu logiku. Takođe, definiše poslovna pravila za podatke, kako se podaci mogu menjati i manipulirati.
2. **Kontroler** upravlja korisničkim zahtevima. Kada korisnik stupi u interakciju sa pogledom (eng. *View*) formira se *URL* (*Universal Resource Locator*) zahtev, koji će biti obrađen od strane odgovarajuće metode kontrolera. Kao odgovor, kontroler daje odgovarajući pogled sa podacima modela. Kontroler predstavlja vezu koja omogućava uspešnu saradnju između pogleda i modela [1].
3. **Pogled** predstavlja komponentu odgovornu za prikaz korisničkog interfejsa i omogućava korisniku da pristupi podacima modela preko veb pretraživača.

Koristeći *MVC*, promene u svakom sloju mogu se napraviti nezavisno jedna od druge, na primer, promene u izgledu ne podrazumevaju promene u druga dva sloja [2].



Slika 1. MVC obrazac

2.4.1. Bootstrap

Bootstrap je okruženje koje koristi *HTML*, *CSS*, i *JavaScript* biblioteke, za kreiranje veb stranica. *MVC* koristi *bootstrap* koji omogućava elegantan i odgovarajući izgled aplikacije koji se može lako prilagoditi [3].

2.4.2. jQuery AJAX

jQuery je brza, mala i bogata *JavaScript* biblioteka. Omogućava operacije poput manipulacije *HTML* dokumenata, rukovanja događajima, animacije i *Ajax*-a mnogo jedno-

stavnije pomoću jednostavnog *API*-ja koji radi u mnoštvu pretraživača.

AJAX (*Asynchronous JavaScript and XML*) koristi mogućnost *JavaScript*-a da šalje asinhronu *http* zahteve i uzima *XML* podatke kao odgovor (ili podatke nekog drugog formata) i osvežava deo veb stranice (koristeći *JavaScript*) bez potrebe da se osvežava cela veb stranica.

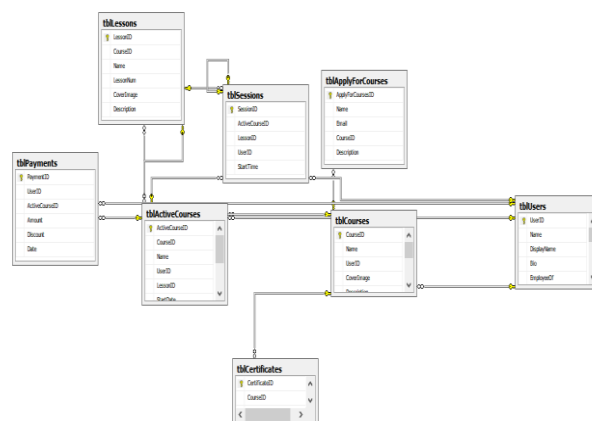
3. OPIS FUNKCIONALNOSTI APLIKACIJE

U ovom poglavlju će biti predstavljene bitne funkcionalnosti baze podataka i aplikacije koja je razvijena u *WPF* tehnologiji i u *ASP.NET* tehnologiji koja je bazirana na *MVC* dizajnerskom obrascu.

3.1. Opis funkcionalnosti baze podataka

Prvi korak u kreiranju projekta je bio kreiranje baze podataka. Za kreiranje baze podataka korišćen je *SQL Server Management Studio* (*SSMS*) koji je omogućio dodavanje tabela u bazu podataka i manipulaciju među podacima. *SSMS* predstavlja integrisano okruženje za upravljanje infrastrukturom *SQL* servera. Obezbeđuje korisnički interfejs i grupu alata sa bogatim skript editorima za rad sa objektima servera.

Baza podataka je modelovana tako da pruži podršku sistemu za upravljanje radom privatne škole. Baza podataka sadrži 8 tabela, koje su zajedno sa relacijama između njih prikazane na slici 2.



Slika 2. Šema baze podataka

3.2. Opis funkcionalnosti desktop aplikacije

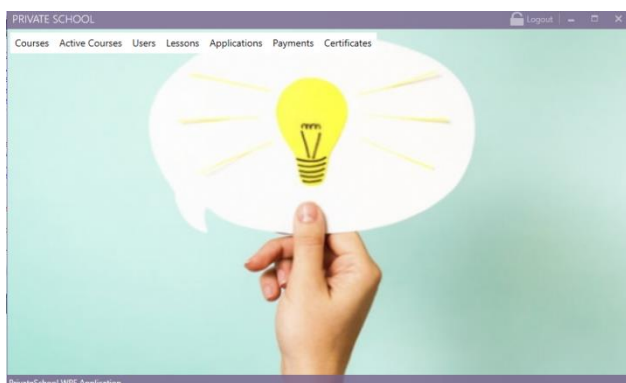
Desktop aplikacija razvijana je unutar razvojnog okruženja *Microsoft Visual Studio 2013*. U prvom koraku napravljen je *WCF* (eng. *Windows Communication Foundation*) servis. On predstavlja servisno-orijentisan sistem razmene poruka koji je omogućio razvoj veb servisa, tj. komunikaciju između klijentske aplikacije i windows servisa.

Nakon kreiranog servisa, isti se povezuje sa ranije kreiranom bazom podataka dodavanjem *Entity Data Model*-a servisu.

U nastavku rada pristupa se kreiranju *WPF* (eng. *Windows Presentation Foundation*) aplikacije koja će komunicirati sa prethodno kreiranim *WCF* servisom.

Na samom početku aplikacije, otvara se naslovna strana aplikacije, odnosno početni prozor. Na početnom prozoru

(Slika 3) možemo videti opcije koje nudi privatna škola: Kursevi, Aktivni kursevi, Sertifikati, Korisnici, Plaćanja, Lekcije i Apliciranje za pohađanje kurseva.



Slika 3. Naslovna strana desktop aplikacije

Klikom na bilo koji od dugmića prikazuje se meni sa dve ponuđene opcije. Klikom na dugme Courses može da se bira između prikaza svih kurseva koji postoje u školi ili da se izabere dodavanje novog kursa u bazu privatne škole.

Ukoliko neko želi da se prijavi na neki od ponuđenih kurseva, klikom na Application dobija se opcija prikaza svih prijava ili dodavanje odnosno kreiranje nove prijave. Korisnik koji se uloguje kao polaznik ima pravo da kreira novu prijavu.

Lekcije se odnose na pomoć pri savladavanju gradiva koje se prolazi tokom pohađanja kursa. Svaki polaznik ima mogućnost neograničenog pristupa lekcijama. Aktivni kursevi predstavljaju kurseve koji su trenutno aktivni i nakon popunjavanja prijave za kurs polaznik se može odmah priključiti slušanju kursa.

Takođe jedan od uslova za pohađanje kursa jeste da se kurs plati. Kreiranje uplata je dozvoljeno predavačima koji su dužni da isprate koji je polaznik izvršio uplatu kako bi mogli da ga priključe predavanjima.

Nakon završenog kursa i položenog završnog testa svaki polaznik koji je zadovoljio uslove dobija sertifikat o položenom kursu.

3.3. Opis funkcionalnosti veb aplikacije

Nakon kreiranja desktop aplikacije, sledeći korak je bio kreiranje veb aplikacije i generisanje *Entity Framework* modela iz tabela kreirane baze podataka. Ovi modeli su klase koje služe za rad sa podacima i svaka klasa predstavlja tabelu u bazi podataka i sadrži svojstva koja odgovaraju kolonama u tabeli [4]. Nakon toga su kreirani kontroleri i pogledi.

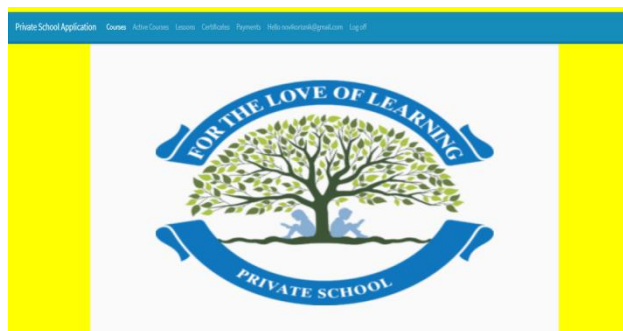
Sistem procesa poslovanja privatne škole predstavlja veb aplikaciju. Svrha veb aplikacije je olakšano praćenje poslovanja privatne škole, spiskova koji se popunjavaju kao što su spisak postojećih kurseva, spisak kurseva koji su trenutno aktivni, podaci o sertifikatima koji se stiču nakon završenog kursa i tako dalje, i smanjenje grešaka koje nastaju usled velike količine administrativnih podataka.

Korišćenjem aplikacije, zaposleni imaju mogućnost dodavanja korisnika u bazu podataka, kreiranje novog kursa, kreiranje trenutno aktivnih kurseva, apliciranje za kurs, kreiranje uplata koje korisnik odnosno polaznik kursa plaća, detalji o sertifikatima koje može da dobije nakon

položenog testa na kraju kursa koji je pohađao, lekcije koje su vezane za kurs i služe za učenje predenog gradiva.

Na stranici za logovanje, korisnik unosi korisničko ime i lozinku nakon čega se izvršava autentifikacija korisnika. Ukoliko je logovanje neuspešno, korisnik dobija informaciju o grešci koja može nastati usled unosa neispravnog korisničkog imena ili lozinke.

Inicijalni prikaz aplikacije nakon uspešnog logovanja korisnika prikazan je na slici 4.



Slika 4. Naslovna strana veb aplikacije

4. DISKUSIJA REZULTATA

Za potrebe ovog rada razvijana je veb aplikacija nakon što je prethodno već bila razvijena desktop aplikacija. Da li je bolja veb ili desktop aplikacija, šta je jednostavnije, šta je brže i praktičnije? Poslednjih godina industrija forsira veb aplikacije. Veb aplikacije imaju neke očigledne prednosti. Običnom korisniku računara je svakako jednostavnije da koristi veb aplikacije[5]:

- ✓ ništa ne treba da se instalira,
- ✓ nove verzije softvera su automatski dostupne
- ✓ cena mesečnog zakupa predstavlja samo delić cene ekvivalentne desktop aplikacije

Sa druge strane, ukoliko pitate programere, veliku prednost daje to što tim koji je razvio softver može da prati statističke podatke vezane za aplikaciju u realnom vremenu. To je velika prednost, jer omogućava uvid u interakciju korisnika sa vašim proizvodom.

Na kraju, možda i najjači argument - novac. Veb aplikacija funkcioniše na različitim platformama i operativnim sistemima. Sa druge strane, scenario u kome treba da se napravi desktop aplikacija u verzijama za Windows, Linux i OS X. Dobri programeri nisu jeftini, tako da razvoj istog softvera za različite platforme mnogo košta.

Čak i u današnje vreme, kada je internet signal dostupan skoro na svakom koraku, postoje trenuci kada nismo na mreži. To su situacije kada posao može da se završi samo pomoću desktop ili native aplikacija. Inženjeri u svakodnevnom radu za modeliranje, projektovanje i simulaciju različitih procesa koriste aplikacije koje zahtevaju veliku računarsku snagu. To su desktop aplikacije.

Desktop aplikacije su tradicionalno ograničene hardverom na kojem se pokreću. One moraju biti razvijene i instalirane na određenom operativnom sistemu i mogu imati stroge hardverske zahteve koji moraju biti ispunjeni kako bi se osiguralo pravilno funkcionisanje. Ažuriranja aplikacija mora korisnik primeniti direktno na njihovu insta-

laciju, a za njihovo funkcionisanje mogu biti potrebne nadogradnje hardvera ili druge promene. Ova zavisnost od hardvera, kao i nasleđe glavnih terminala aplikacija, obično ograničava nivo složenosti u korisničkim interfejsima za desktop aplikacije [6].

5. ZAKLJUČAK

U ovom radu je opisano razvijeno rešenje za automatizaciju poslovanja privatne škole. Na ovaj način omogućeno je da se lako i jednostavno vodi evidencija o kursevima koji su u ponudi škole, trenutno aktivnim kursevima koji su na raspolaganju zainteresovanim polaznicima, sertifikatima koji se stiču nakon završetka kursa, vođenje evidencije o plaćanju kurseva, pristup literaturi koja se koristi u vidu sažetih lekcija za svaki deo kursa koji predavač ispredaje. Takođe, moguće je preko aplikacije i izvršiti prijavu za željeni kurs te se može imati uvid u zainteresovan broj polaznika za jedan određeni kurs.

Aplikacija je prvo realizovana u programskom okruženju Microsoft Visual Studio 2013 uz korišćenje fundamentalnih osnova WPF - Windows Presentation Foundation tehnologije, i uz korišćenje Microsoft SQL Server Management-a kako bi se kreirala zadovoljavajuća baza podataka.

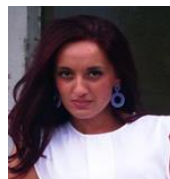
Aplikacija je zatim realizovana i na drugi način, u programskom okruženju Microsoft Visual Studio 2013 uz korišćenje znanja i tehnologija koja zahtevaju aplikacijski okvir ASP.NET MVC, zatim Entity Framework kao Object Relational Mapper, HTML-a, programskog jezika C# i sličnih potrebnih tehnologija. Korišćenjem navedenih tehnologija i pristupa, programski je ostvarena veb aplikacija i baza podataka za odgovarajuću primenu.

Ova aplikacija ima potencijala da se proširi i da zaposlenima dodatno olakša posao. Jedna od nadogradnji vezanih za aplikaciju moglo bi biti razvijanje mobilne aplikacije i integracija sve tri aplikacije koristeći jedan API.

6. LITERATURA

- [1] Srđan Sladojević, Mirjana Dulić, Predrag Jelovac, Ivan Edelinski, Darko Stefanovic (2015), „One solution of web application for data acquisition from remote sensing devices“, Infoteh – Jahorina 2015, Republika Srpska, Jahorina, Vol. 14, str. 788-792.
- [2] Kamil Śmietana, Michał Szermer, Wojciech Zabierowski, „Web Technologies in development of working whith time reporting system using ASP.NET,“ in Technical University of Lodz, 2016
- [3] Internet: Microsoft “ASP.NET MVC 5“. Dostupno na: <https://docs.microsoft.com/en-us/aspnet/mvc/mvc5> (pristupljeno u oktobru 2019.)
- [4] Internet: Microsoft „Getting Started with Entity Framework 6 Database First Using MVC 5“. Dostupno na: <https://docs.microsoft.com/en-us/aspnet/mvc/overview/getting-started/database-first-development/> (pristupljeno u oktobru 2019.)
- [5] Internet: <http://sr.magma.rs/blog/web-aplikacije/web-aplikacija-ili-desktop-aplikacija> (pristupljeno u oktobru 2019.)
- [6] Internet: <https://www.seguetech.com/desktop-vs-web-applications/> (pristupljeno u oktobru 2019.)

Kratka biografija:



Milica Terzić rođena je 1989. godine u Novom Sadu. Gimnaziju „Isidora Sekulić“ društveno-jezički smer završila je u Novom Sadu. 2008. godine je upisala osnovne akademske studije na Fakultetu tehničkih nauka u Novom Sadu, smer Inženjerski menadžment; a 2014. je i završila master studije takođe na smeru za Inženjerski menadžment, iz oblasti Investicioni menadžment. Godine 2015. je upisala master studije na smeru Inženjerstvo informacionih sistema na Fakultetu tehničkih nauka.