

UNIVERZA V LJUBLJANI
EKONOMSKA FAKULTETA

MAGISTRSKO DELO

**INFORMATIZACIJA SPREMLJANJA IZOBRAŽEVANJ
ZAPOSLENIH V IZBRANI SREDNJI ŠOLI**

Ljubljana, november 2015

PETER KREBELJ

IZJAVA O AVTORSTVU

Spodaj podpisani Peter Krebelj, študent Ekonomske fakultete Univerze v Ljubljani, izjavljam, da sem avtor zaključnega magistrskega dela z naslovom Informatizacija spremljanja izobraževanj v izbrani srednji šoli, pripravljene(-ga) v sodelovanju z mentorjem prof. dr. Tomaž Turkom.

Izrecno izjavljam, da v skladu z določili Zakona o avtorski in sorodnih pravicah (Ur. l. RS, št. 21/1995 s spremembami) dovolim objavo magistrskega dela na fakultetnih spletnih straneh.

S svojim podpisom zagotavljam, da

- je predloženo besedilo rezultat izključno mojega lastnega raziskovalnega dela;
- je predloženo besedilo jezikovno korektno in tehnično pripravljeno v skladu z Navodili za izdelavo zaključnih nalog Ekonomske fakultete Univerze v Ljubljani, kar pomeni, da sem
 - poskrbel, da so dela in mnenja drugih avtorjev oziroma avtoric, ki jih uporabljam v magistrskem delu, citirana oziroma navedena v skladu z Navodili za izdelavo zaključnih nalog Ekonomske fakultete Univerze v Ljubljani, in
 - pridobil vsa dovoljenja za uporabo avtorskih del, ki so v celoti (v pisni ali grafični obliki) uporabljena v tekstu, in sem to v besedilu tudi jasno zapisal;
- se zavedam, da je plagiatorstvo – predstavljanje tujih del (v pisni ali grafični obliki) kot mojih lastnih – kaznivo po Kazenskem zakoniku (Ur. l. RS, št. 55/2008 s spremembami);
- se zavedam posledic, ki bi jih na osnovi predloženega magistrskega dela dokazano plagiatorstvo lahko predstavljalo za moj status na Ekonomski fakulteti Univerze v Ljubljani v skladu z relevantnim pravilnikom.

V Ljubljani, dne _____

Podpis avtorja: _____

KAZALO

UVOD	1
1 PREDSTAVITEV ORGANIZACIJE	3
1.1 Opredelitev organizacije.....	3
1.1.1 Organizacija dela	4
1.1.2 Organizacijska shema	5
1.2 Cilji organizacije.....	5
1.3 Poslovni procesi v organizaciji.....	6
1.4 Izobraževanja kadrov.....	9
2 ANALIZA TEHNOLOŠKE ARHITEKTURE.....	10
2.1 Strežniško okolje	12
2.1.1 Načrtovanje virov podjetja	12
2.1.2 Pregled celotnih stroškov lastništva	12
2.1.3 Razširljivost informacijskega okolja	13
2.2 Krajevno omrežje	15
2.2.1 Topologije omrežij.....	16
2.2.2 Varnost krajevnega omrežja	17
2.3 Programska oprema in licence.....	19
2.3.1 Namenski ali uporabniški programi.....	19
2.3.2 Programska orodja za izgradnjo namenskih programov	19
2.3.3 Sistemski programi	20
2.3.4 Programska oprema in licence v organizaciji.....	20
2.4 Obstoječe rešitve	21
2.5 Ocena stanja in možnosti prehoda.....	21
2.5.1 Ključni dejavniki uspeha in analiza informacijskih potreb	21
2.5.2 Trenutno tehnološko okolje	23
2.5.3 Poslovno modeliranje ključnih procesov.....	23
2.5.4 Predlagana prenova s pomočjo informatizacije.....	24
3 ANALIZA ZAHTEV	26
3.1 Zakonska osnova	26
3.2 Omejitve	29
3.3 Analiza poslovnih procesov	30
4 IZDELAVA USTREZNE REŠITVE	30
4.1 Načrtovanje projekta.....	31

4.2	Izbira ustrezne tehnologije	35
4.3	Izdelava modela.....	42
4.4	Izdelava rešitve.....	46
4.4.1	Konfiguracija strežniškega okolja	46
4.4.1.1	Aktivacija podpora IIS	46
4.4.1.2	Postavitev strežnika SQL.....	48
4.4.1.3	Konfiguracija požarnega zidu.....	49
4.4.2	Izdelava programske rešitve	51
4.4.2.1	Podatkovna baza SQL	51
4.4.2.2	Konfiguracija Aktivnega imenika (AD).....	53
4.4.2.3	Konfiguracija IIS	53
4.4.2.4	Izdelava spletne aplikacije.....	56
4.5	Analiza rešitve.....	61
	SKLEP.....	64
	LITERATURA IN VIRI.....	67
	PRILOGE	

KAZALO SLIK

Slika 1:	Organigram SŠTS Šiška	5
Slika 2:	Prikaz preprostega poslovnega procesa	6
Slika 3:	Potek prenove procesa	8
Slika 4:	Proces izobraževanj zaposlenih AS-IS	9
Slika 5:	Življenjski cikel informacijskega sistema.....	11
Slika 6:	Prikaz strežniške omare	13
Slika 7:	Poenostavljena shema omrežja	16
Slika 8:	Poslovni proces izobraževanj TO-BE	24
Slika 9:	Sodobni, uporabniško usmerjeni model.....	32
Slika 10:	Grafični vmesnik TO-BE spletne aplikacije	34
Slika 11:	Osnovni koraki v razvoju aplikacije	35
Slika 12:	Izbira parametrov za odločanje	37
Slika 13:	Variante izbire.....	37
Slika 14:	Prikaz povprečnih uteži kriterijev	38
Slika 15:	Rezultati vrednotenja z orodjem Dexi	38
Slika 16:	Prikaz grafa, kot rezultat vrednotenja	39
Slika 17:	Model naredi ali kupi	40
Slika 18:	Izbira naredi ali kupi z vrednostmi	40
Slika 19:	Povprečne uteži za naredi ali kupi	41
Slika 20:	Prikaz rezultatov vrednotenja za naredi ali kupi.....	41
Slika 21:	Lasten razvoj ali zunanji razvoj odločitvenega grafa	41
Slika 22:	Postopek priprave podatkovne baze.....	43

Slika 23: E-R model v programu SAP Power Designer	44
Slika 24: Objektno usmerjen model C#.....	45
Slika 25: Konceptualni model	45
Slika 26: Namestitev spletnega strežnika IIS	46
Slika 27: Upravljanje spletnega strežnika	48
Slika 28: Urejanje pravil v požarnem zidu	50
Slika 29: Vnos spletne strani v korensko območje DNS.....	51
Slika 30: SQL Server Management Studio	52
Slika 31: Prikaz tabele Dokumenti	53
Slika 32: Določanje parametrov za lastnosti aplikacije znotraj IIS.....	54
Slika 33: Nastavitev povezave do podatkovne baze.....	55
Slika 34: Nastavitev podpore 32 bitni aplikaciji	55
Slika 35: Visual Studio MasterPage	57
Slika 36: Preverjanje uporabnika v domenskem okolju	57
Slika 37: Prikaz vodenja izobraževanj.....	59
Slika 38: Dodajanje izobraževanja	59
Slika 39: Administratorski pogled nad izobraževanji.....	60
Slika 40: Prikaz izobraževanj zaposlenega.....	61

KAZALO TABEL

<i>Tabela 1: Pregled zahtev in doseženih ciljev aplikacije.....</i>	62
---	----

UVOD

Kolektivna pogodba za dejavnost vzgoje in izobraževanja v Republiki Sloveniji (Ur.l. RS št. 40/12 in 46/13, v nadaljevanju KPDVI) za zaposlene v srednjih šolah poleg poučevanja in drugega dela, ki je odrejeno s strani ravnatelja, določa tudi redno strokovno izpopolnjevanje. Izobraževanja so tudi eden izmed načinov, ki zaposlenim prinesejo točke za napredovanje v povezavi s trajanjem strokovnega izpopolnjevanja. Zaposleni mora tako letno opraviti 5 dni izobraževanj ali 15 dni na vsaka 3 leta.

Učitelji se lahko strokovno izpopolnjujejo na izobraževanjih, ki jih najdemo v Katalogu programov nadaljnega izobraževanja in usposabljanja strokovnih delavcev v vzgoji in izobraževanju (Katis - katalog programov nadaljnega izobraževanja in usposabljanja strokovnih delavcev, 2015) v tekočem šolskem letu. Izobraževanja so razpisana in potrjena s strani Ministrstva za izobraževanje, znanost in šport. Dolžina izpopolnjevanja oz. konkretneje število ur, ki jih zaposleni izvede v sklopu izobraževanja prinaša točke za napredovanje in potrdilo o udeležbi na izobraževanju. Zaposleni v srednji šoli se lahko udeležijo strokovnih izpopolnjevanj tudi izven razpisanega kataloga, vendar mora smer in način izvedbe izobraževanj ustrezati strokovnemu področju in hkrati mora zaposleni pridobiti ustrezno potrdilo, kjer je razvidno število opravljenih ur in utemeljitev, kje in na kakšen način bi izobraževanje pripomoglo zaposlenemu pri njegovem delu, kar je opredeljeno v Pravilniku o napredovanjih zaposlenih v vzgoji in izobraževanju v nazive (Ur.l. RS št. 48/11 in 92/12, v nadaljevanju PNZVIN).

Razvoj informacijskega okolja, ki bo podpiralo vodenje izobraževanj ob upoštevanju zahtev kolektivne pogodbe, je lahko v veliko pomoč pri vodenju in načrtovanju kadrov. Na kadrovskem področju lahko tako pričakujemo dvig razpoložljivosti in produktivnosti obstoječih kadrovskih potencialov. Kadri predstavljajo tudi konkurenčno prednost, saj bodo učinkoviteje pristopali do podatkov in bodo tako imeli na voljo dodatna izobraževanja, ko bo le to potrebno. Delo kadrov bo boljše zaradi razvoja orodij in tehnologije in hkrati bodo delali bolje, če bodo stimulirani in motivirani. Preko novih informacijskih procesov se zmanjša časovna odzivnost tako obstoječih kot novih procesov (Kovačič, Jaklič, Indihar Štemberger & Groznik, 2004).

Na tem mestu si lahko zastavimo vprašanje, kakšen bo dejanski vpliv uporabe informatike. Podatki, ki jih ne moremo nadzirati in ne vemo kje in v kakšni obliki so shranjeni, nam ne bodo koristili dovolj in nam s tem podajali ustrezno poslovno strategijo. Kot je navedeno v literaturi (Kovačič & Boksilj Vukšič, 2005), se sama prenova poslovanja pričinja z zavedanjem managementa o nujnosti po spremembah. Sama prenova je sicer celovit proces, vendar se bomo osredotočili na problem, ki je bil prikazan tudi s strani vodstva izbrane organizacije, in sicer zmožnost nadzora izobraževanj zaposlenih.

V praksi lahko s pregledom literature hitro ugotovimo, da ne obstaja neka skupna rešitev, ki bi reševala prikazani problem. Vodenje zaposlenih se izvaja preko zbiranja potrdil v osebnih mapah zaposlenih, alternativno tudi preko orodja Microsoft Excel, kjer se beleži podatke o številu izobraževanj in številu opravljenih ur. Ob primerni uporabi orodja Microsoft Excel bi sicer vse te podatke lahko vodili, vendar je nadzor vnosa v nek skupen dokument neprimeren, vse podatke bi videli vsi zaposleni in vodenje pridobljenih potrdil bi bilo težavno. Tudi sočasna dostopnost do takšnega dokumenta predstavlja omejitve. Implementirana rešitev mora zato zadoščati trenutnemu informacijskemu okolju, biti mora relativno enostavna za uporabo brez dolgih izobraževanj in usposabljanj, zagotavljati mora zanesljivo rešitev, ki bi predstavljala hitrejše, učinkovitejše in zanesljivejše orodje za delo.

Rešitev bi tudi omogočila hitrejši pregled stanja vseh zaposlenih in hkrati omogočila posameznikom, da izvedejo vpogled v svojo osebno mapo preko aplikacije.

Ker iskana rešitev še ni bila pripravljena, je moj cilj narediti celovito analizo poslovnega okolja v izbrani organizaciji in informacijskega okolja v katerem bi bila pripravljena rešitev. Pripravil bi model, po katerem bi bila izvedena končna implementacija. Skozi analizo poslovnega in informacijskega okolja se osredotočam tudi na zahteve, ki jih pričakuje vodstvo in končni uporabniki. S primerno rešitvijo bi se izboljšal poslovni proces znotraj organizacije, obveščanje postane hitrejše ter učinkovitejše, s čimer se tudi lažje in hitreje sprejema odločitve in zaposlenim omogoči ustrezno področje in časovno trajanje izobraževanja.

Na podlagi primerov in najboljših praks, ki so zbrani in tudi kritično opredeljeni s strani znanstvene literature (Bajaj & Wrycza, 2009), bom preučil okolje naše organizacije in pripravil model, ki bo ustrezal končni rešitvi problema. Skozi proces razvoja življenjskega cikla nekega sistema, ki so ga povzeli (Valacich, George & Hoffer, 2009), sami začetki pojave modela pa segajo v šestdeseta leta, bom skozi potrebne korake preučil izbrano poslovno okolje in zmožnost razvoja ustrezne platforme za izdelavo končne rešitve. Na podlagi omenjenega modela bom pripravil intervju z vodstvom šole o željah in potrebah ter ob samem zaključku analizo primernosti aplikacije za naročnika ob uvedbi nove rešitve. Izvedena bo tudi ocena uspešnosti vpeljave in ocena izboljšanja poslovnega procesa izobraževanj.

Končna aplikacija bo omogočala vpogled trenutnega stanja izobraževanj tako vodstvu kot tudi posameznikom, kjer bodo zaposleni dostopali le do svojih podatkov s pomočjo uporabe informacijske infrastrukture. Vodstvo se bo lažje odločilo o primernosti posameznega izobraževanja glede na področje zaposlenega, imelo bo vpogled v stanje izpolnjevanja zahtev pogodbe in zakonsko predpisane osnove za izobraževanja zaposlenih v šolstvu. Posameznik bo dodatno motiviran, da se udeleži manjkajočih izobraževanj ali pridobi manjkajoče točke za napredovanje. Končna rešitev bo tako podprla proces izobraževanj in s tem izboljšala obstoječega.

Magistrska naloga bo na začetku vsebovala predstavitev organizacije, kjer bodo predstavljene temeljne značilnosti in cilji njenega delovanja. Zaposleni v izbrani organizaciji se morajo izobraževati na podlagi pogodbe o zaposlitvi. Ker je potrebno izobraževanja voditi za vsakega posameznika, se bom osredotočil na trenutni proces vodenja izobraževanj za posameznega zaposlenega. V tem delu se bom opiral predvsem na zakonsko izhodišče ter samo šolsko zakonodajo, ki je ob tem temeljno izhodišče prenove.

Sledila bo analiza poslovnega okolja podjetja, kjer bo potekala tudi analiza trenutnega informacijskega okolja in programskih rešitev, ki so trenutno v uporabi. S pomočjo kritične ocene, ki bo slonela na strokovni literaturi, bo izvedena ureditev informacijske infrastrukture in možnosti uporabe ustrezne platforme, ki bo osnova za nadaljnje reševanje problema (Valacich, George & Hoffer, 2009). Poleg strokovne literature, ki bo temelj analize, se bom v tej fazi moral opreti tudi na pridobljena znanja podiplomskega študija, pomembna pa bodo tudi znanja področja informatike, kjer sem tako v preteklosti kot tudi danes opravljal različna dela pri uvajanju novosti kot tudi vzdrževanju obstoječih sistemov.

Za pridobivanje nujno potrebnih informacij, ki so potrebne za razumevanje problema in ustrezno strukturiranje rešitve, je temeljno tudi poznavanje osnovnega problema, in sicer razlogi za uvedbo aplikacije kot tudi temeljni elementi, ki jih moramo v aplikaciji zajeti in jih ustrezno rešiti. V ta namen bom v naprej pripravil vprašanja in preko intervjuja pridobil odgovore na ustrezna vprašanja s strani vodstva organizacije kot tudi poslovne sekretarke,

ki podatke o izobraževanjih vodi. S tem mi bo omogočeno celovito razumevanje problema in nadaljnje razvijanje končne rešitve, ki bo tudi vključeno v trenutno informacijsko okolje ter bo uspešno pokrivalo specifični poslovni proces izobraževanj.

Zadnji del naloge predstavlja izdelava ustrezne rešitve, ki bo temeljila na analizi vseh pridobljenih podatkov in analizi zahtev, ki bodo v celoti podane. Rešitev bo upoštevala tudi koncepte razvoja informacijskih rešitev (Turk, 2013), kjer bo potrebno vložiti tudi druga znanja programskega razvoja in različne tehnologije, ki mi bodo omogočile doseg končnega cilja izdelave aplikacije. Končna ocena uspešnosti bo ocena uspešnosti vpeljave nove aplikacije in pregled zadovoljstva uporabnikov aplikacije.

Namen izdelave magistrskega dela je:

- Analizirati poslovno okolje izbrane organizacije in ureditev informacijske infrastrukture.
- Izdelati načrt prenove podpore izobraževanju učiteljev.
- Informacijsko podpreti proces izobraževanja učiteljev.
- Analizirati in urediti varnost v krajevnem omrežju in razvojni tehnologiji.

Cilji magistrskega dela:

- Izdelati aplikacijo, ki bo podprla proces izobraževanja učiteljev.
- Aplikacijo vpeljati v praktično uporabo.
- Izvesti oceno uspešnosti vpeljave in oceniti vpeljano rešitev s stališča izboljšanja poslovanja.

1 PREDSTAVITEV ORGANIZACIJE

1.1 Opredelitev organizacije

Kot je bilo opredeljeno s strani Zupan (2003), bi za organizacijo lahko rekli iz treh različnih vidikov, da:

1. Jo opredelimo kot sistem, ki ga oblikujemo s kombiniranjem sestavnih elementov organizacije glede na zastavljene cilje znotraj nje.
2. Jo opredelimo kot proces, kjer govorimo predvsem o organiziranju in urejanju dela.
3. Jo opredelimo kot konkretno organizacijsko obliko, torej govorimo o podjetju, zavodu, ustanovi, društvu itn.

Srednja šola tehniških strok Šiška (v nadaljevanju SŠTS Šiška) je bila ustanovljena kot Industrijska kovinarska šola Litostroj. Ustanovitev je bila potrjena z odločbo Glavne direkcije kovinske industrije Slovenije z dne 27. oktober 1947 ter z odločbo Ministrstva za industrijo in rudarstvo Ljudske republike Slovenije na dan 29. januar 1948 (Ustanovitev šole in začetni koraki, 2015).

V letih 1950 – 1960 je šola razvijala modele teoretičnega in praktičnega pouka ter je skrbela za kulturo in vzgojo učencev. Šola je postala eksperimentalna šola Zavoda za strokovno izobraževanje Ljudske republike Slovenije oz. Zavoda republike Slovenije za šolstvo. V letih 1960 – 1973 je s širitvijo Litostroja in s tem povečane proizvodnje, prihajalo do večje potrebe po izobraževanju in izpopolnjevanju odraslih. Šola je tako postala izobraževalni center, ki je nudil tako rednim kot izrednim dijakom enake možnosti.

V letih 1973 – 1981 so sledile šolske reforme. Okrepile so se interesne dejavnosti, hkrati pa je šola začela s podporo pri materialnih dejavnostih učencev. V letih 1981 – 1987 je šola močno sodelovala z Litostrojem in takratno Iskro. V tistem času so se zaradi reform šolstva pojavile tudi težave z izvajanjem praktičnega pouka, kjer je bila težava vezana predvsem na število ur praktičnega pouka. Drugi problem je bilo združevanje različnih programov in različnih poklicev v skupen učni načrt. V tem času je šola močno izboljšala materialno osnovo, zgrajen pa je bil tudi dodatni del poslopja s 14 specializiranimi učilnicami za strokovno – teoretični pouk. S tem je bil omogočen pouk zgolj v enem terminu, pretežno v dopoldanskem času. V zadnjih letih Litostroja, torej med 1987 in 1991, je prišlo do sprememb v odnosu Litostroja do same šole, ki so bile posledica predvsem slabših gospodarskih razmer. Sledilo je obdobje, ki ni bilo najbolj spodbudno za nadaljnje delo in razvoj šole.

V letu 1992 je tako prišlo do spremembe lastništva. Dne 25.3.1992 je lastnica šole postala Republika Slovenija. Sodelovanje z Litostrojem je zaradi slabšanja gospodarskih razmer prešlo v zgolj simbolno.

Ukinitev štiriletnih izobraževalnih programov je predstavljala nov zagon in motivacijo šoli, da se z uvedbo programov 3 + 2 ponovno aktivno vključi v okolje in začne s povečevanjem svoje vloge.

Danes na šoli potekajo različni izobraževalni programi, in sicer:

- tehnik mehatronike (4. letni program srednjega strokovnega izobraževanja, v nadaljevanju SSI),
- električar (3 letni program srednjega poklicnega izobraževanja, v nadaljevanju SPI),
- računalnikar (3 letni program SPI),
- elektrotehnik (3 + 2 program poklicno tehniškega izobraževanja, v nadaljevanju PTI) in
- tehnik računalništva (3 + 2 program PTI).

V šolskem letu 2014/2015 je bilo na šoli 28 oddelkov, kjer je bilo skupaj redno vpisanih 787 dijakov. Poleg rednega izobraževanja potekajo tudi izobraževanja odraslih, kjer je bilo vpisanih 150 udeležencev izobraževanj.

V Srednji šoli tehniških strok Šiška je zaposlenih 60 oseb, od tega je strokovnih delavcev 53. Povprečna starost zaposlenih je 47 let, vendar se le-ta z menjavo kadrov malenkostno spreminja. V zadnjem času se je številka predvsem znižala, zaradi prihoda mlajšega kadra.

1.1.1 Organizacija dela

Zavod opravlja javno službo na področju vzgoje in izobraževanja. Zavod lahko opravlja tudi druge dejavnosti, ki so določene v aktu o ustanovitvi zavoda št. 01403-123/2008/4.

Pri svojem delu mora zavod upoštevati organiziranost, ki se prilagaja potrebam dela za izvajanje dejavnosti zavoda v skladu s finančnimi sredstvi, ki so na voljo za delovanje.

Delo v zavodu opravljajo javni uslužbenci, ki imajo sklenjeno delovno razmerje v zavodu. V skladu s predpisi lahko določena dela izvaja tudi pripravnik, ki mora imeti določenega mentorja. Naloga mentorja je pomoč pri izvajanju nalog, kot tudi usmerjanje pri njegovem delu. Službe znotraj zavoda so razdeljene na tri področja, in sicer:

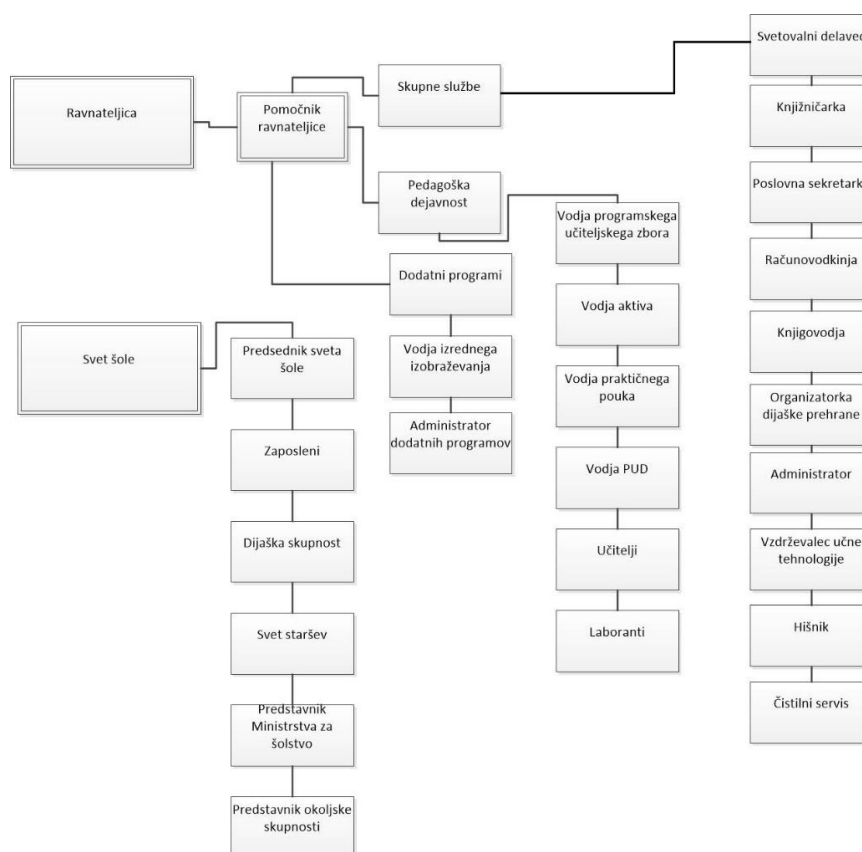
- Skupne službe,

- pedagoška dejavnost in
- dodatni programi.

Razdelitev del znotraj zavoda se opravi na podlagi posameznega delovnega mesta ali na podlagi vrste dela, ki lahko predstavlja naloge več delovnih mest. Delovna mesta znotraj zavoda so sistematizirana, glede na polni ali krajši delovni čas.

1.1.2 Organizacijska shema

Slika 1: Organigram SŠTS Šiška



1.2 Cilji organizacije

Poleg izobraževanj na omenjenih programih, potekajo tudi izobraževanja na področju obnovljivih virov energije. V okviru globalnih ciljev in strategije ustanovitelja šole, gospodarske ljubljanske urbane regije (v nadaljevanju LUR) ter zaposlenih v organizaciji, so glavni cilji predvsem:

- Biti prepoznaven v LUR na področju izobraževanja mladih in usposabljanja odraslih na področju programov šole.
- Na srednji rok postati izobraževalni center za obnovljive vire energije.
- Nadgraditi obstoječe programe z višješolskimi programi, kjer je korak naprej že skupna ustanovitev Šolskega centra Ljubljana, ki predstavlja višje strokovno izobraževanje.

- Zagotavljati ustrezno kakovost dela, s čimer so cilji predvsem usmerjeni v pričakovanja dijakov, staršev, delodajalcev, države in celotne družbe.

Poleg globalnih ciljev, ki smo jih ravnokar navedli, so pomembni tudi splošni cilji, ki so usmerjeni predvsem v strokovne programe ter posebni strokovni cilji, ki so vezani na izvajanje programov SPI, PTI in SSI.

V sklopu splošnih usmerjevalnih ciljev lahko tukaj govorimo predvsem o:

- Zagotavljanju poklicne in strokovne usposobljenosti skupaj s funkcionalnimi kompetencami in vrednotami, ki bodo dijakom zagotavljale uspešno samostojno življenje.
- Zaposlenim na SŠTS Šiška zagotoviti materialne in strokovne osnove, kar bo omogočalo uspešno načrtovanje dela, izvajanje in vrednotenje rezultatov.

V sklopu posebnih strokovnih ciljev Srednje šole tehniških strok Šiška so to predvsem:

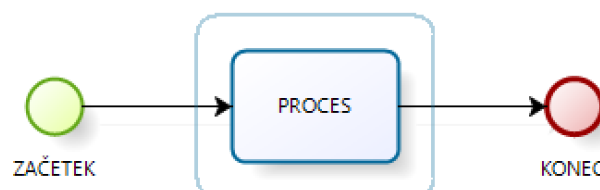
- Ključne kompetence in izdelki, ki so potrebni za opravljanje poklica,
- obvladovanje časa in stroškov, ki bodo zagotavljali izdelavo kakovostnih izdelkov in storitev znotraj razpoložljivih virov,
- izvedbo dela na varen način v varnem okolju.

1.3 Poslovni procesi v organizaciji

Ključna funkcija, ki se je uveljavila v organizacijah, v primeru prehoda na procesno organiziranost, je načrtovanje, organiziranje, informatizacija in skrbništvo poslovnih procesov. Skrbništvo poslovnih procesov nam omogoča zagotavljanje optimalnega poteka izvajanja procesnih aktivnosti in hkrati ustrezno informatizacijo podpore izvajalcem aktivnosti, vključenih v samem procesu. Management poslovnih procesov se najpogosteje razvije kot organizacijska oblika službe oz. sektorja za informatiko s ciljem obvladovanja poslovnih procesov podjetja (Kovačič, Groznik & Ribičič, 2009).

Poslovni proces predstavlja skupek logično povezanih izvajalskih in nadzornih aktivnosti. Rezultat takšnega procesa je izdelek ali storitev, ki smo ga načrtovali. Opredelimo ga lahko tudi kot povezan nabor dejavnosti in nalog, ki vhodnim elementom dodajo neko dodatno uporabno vrednost, ko se tak proces zaključi. Proces je prepoznaven tako po aktivnostih izvajalcev kot tudi zaporedju opravil in aktivnosti, ki jih je potrebno izvesti. Ureditev predstavlja aktivnosti časa in prostora, ki se z nekim prožilcem pričnejo in kasneje tudi končajo, vhodi in izhodi so jasno zaznani (Kovačič & Boksilj Vukšić, 2005).

Slika 2: Prikaz preprostega poslovnega procesa



Vir: A. Kovačič & V. Boksilj Vukšić, Management poslovnih procesov, 2005, str. 224.

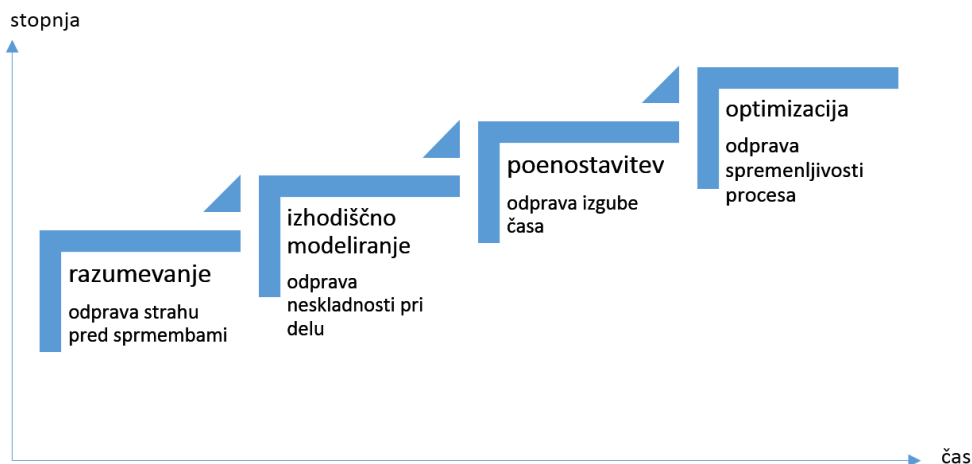
Vsaka aktivnost v podjetju ali zunaj njega lahko predstavlja proces. Smiselno je opredeliti in upoštevati le tiste aktivnosti, ki posredno ali neposredno vplivajo in prispevajo k dodani vrednosti našega končnega izdelka oz. storitve. Upoštevanje osnovnih značilnosti nam tako pomaga pri analizi in prenovi procesov (Kovačič & Boksilj Vukšič, 2005):

- Opredelitev cilja procesa,
- določanje lastnika procesa,
- določanje začetka in konca procesa,
- določanje vhodov in izhodov procesa,
- določanje zaporedja izvajanja samega procesa,
- ravnanje v primeru neskladnosti znotraj procesa,
- določitev merljivih vrednosti, ki nam bodo omogočale ugotavljanje učinkovitosti procesa,
- prepoznani končni uporabniki procesa in
- preprečevanje izboljševanja procesa.

Zavedanje managementa o potrebi po prenovi informacijskega sistema je običajno prvi korak v samem procesu prenove. V procesu prenove je potrebno poleg tehnološkega vidika obravnave problema, upoštevati tudi delovanje organizacije kot celote. Opredeliti je potrebno poslovne procese, ki potekajo znotraj organizacije. Večina organizacij je usmerjena funkcijsko, kjer procesi potekajo skozi več različnih organizacijskih enot, kar predstavlja zahteven in hkrati pomemben korak. Izvajalci procesov običajno poznajo le delčke, ob sami identifikaciji teh procesov zato vključujemo posameznike, ki nam bodo takšen proces v celoti pomagali razumeti. Pri tem si pomagamo z izdelavo modela trenutnega stanja »AS-IS«, pri katerih si pomagamo s tehniko modeliranja. Kasnejša izvedba modeliranja procesov sledi tudi z uporabo orodij, ki nam običajno omogočijo izdelavo grafičnega poteka procesa. Skozi analizo procesov podjetje le-te bolje spozna in jih lahko izboljša. Orodja nam ponujajo simulacijo teh procesov, s čimer preverimo ali v obstoječih ali novo načrtovanih procesih obstajajo ozka grla, ki jih tako lažje odkrijemo in po potrebi preoblikujemo. Na podlagi analize obstoječih procesov lahko organizacija razmišlja o njihovi prenovi. Ustvarjene modele lahko analiziramo skozi orodja za modeliranje in s tem ponovno ustvarimo simulacijo poteka. Končni modeli predstavljajo model predlogov prenove oz. »TO-BE« model. Končni model preko informacije vpeljemo v organizacijo (Kovačič, Jaklič, Indihar Štemberger & Groznik, 2004).

Potek prenove poslovnih procesov mora biti v naprej jasno znan. Da bi lahko učinkovito spremljali razvoj in končne rezultate prenove, si lahko pomagamo s štirimi stopnjami znotraj časa razvoja, ki nam bodo podale končno sliko naše uspešnosti.

Slika 3: Potek prenove procesa



Vir: A. Kovačič & V. Boksilj Vukšić, *Management poslovnih procesov*, 2005, str. 50.

Na sliki je prikazan potek prenove poslovnih procesov. Procesi so v tem primeru predstavljeni s pomočjo konceptualnega modela. Poudarek prve stopnje prenove je na razumevanju vzrokov za prenovo, potek prenove in posledice, v katere bo prenova vodila.

Proces upravljanja sprememb je običajno povezan z zaznavanjem nekega problema znotraj podjetja. Takšen proces mora biti prepoznan in tudi identificiran. V primeru naše izbrane organizacije, je bil problem zaznan s strani vodstva šole in je povezan s procesom izobraževanj zaposlenih v izbrani organizaciji.

Da bi bila sprememba procesa uspešna, mora biti pobuda za uvajanje sprememb sprejeta tako na nivoju zaposlenih kot tudi nivoju managementa (Kovačič & Boksilj Vukšić, 2005). Ker je bila potreba po prenovi izražena s strani managementa, sem lahko prepričan, da bo proces prenove v celoti podprt. Vsekakor se zavedam, da to vseeno ne pomeni neomejenih sredstev, temveč je potrebno najti rešitev v okviru trenutne tehnološke arhitekture in hkrati brez vplivanja na delovanje obstoječega tehnološkega sistema in obstoječih procesov.

Po drugi strani so zaposleni tisti, ki bi proces prenove podpore izobraževanj zavrnili. S pripravo končne rešitve, je moj cilj, zaposlene s pomočjo prikaza pozitivnih vplivov spletne aplikacije prepričati o nujnosti uvedbe sistema in hkrati zmanjšanju časa, ki ga sami potrebujejo za trenutno vodenje in arhiviranje.

Da bi lahko razumeli trenutne procese v izbrani organizaciji, bi bilo potrebno pripraviti analizo obstoječih procesov kot osnovo za nadaljnje delo.

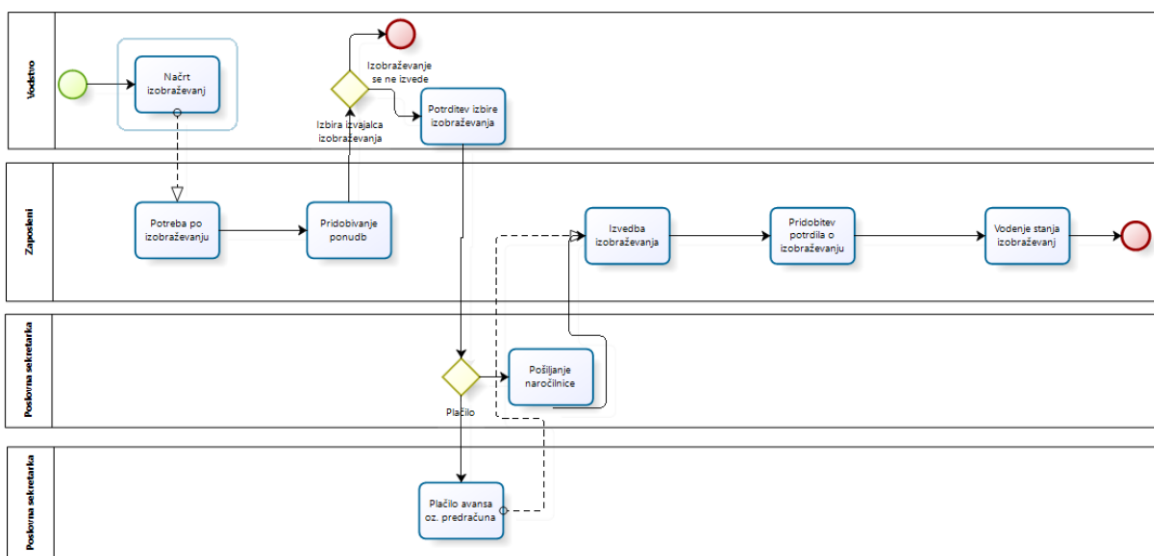
Ker gre za vzgojno izobraževalno ustanovo oz. lahko rečemo kar šolo, so temeljni procesi povezani z osnovno dejavnostjo, in sicer so to:

- Redno izobraževanje dijakov v programih SSI, SPI in PTI,
- izredno izobraževanje dijakov v programih SSI, SPI in PTI,
- izobraževanja za podjetja in ustanove na področju fotonapetostnih sistemov,
- nabava materiala, opreme in drobnega inventarja.

Mi se bomo skozi nalogo osredotočili na podporni proces, in sicer gre za proces dodatnega izobraževanja in strokovnega izpopolnjevanja zaposlenih znotraj organizacije.

Na naslednji sliki je predstavljen poslovni proces izobraževanj zaposlenih znotraj izbrane organizacije. Poslovni proces se lahko začne s strani vodstva ali zaposlenih glede na njihove želje in potrebe. Izobraževanja so opredeljena v letnem delovnem načrtu (v nadaljevanju LDN), kar pomeni, da mora biti izbira izobraževanj skladna z načrtovanjem šolskega leta.

Slika 4: Proces izobraževanj zaposlenih AS-IS



V kolikor se želja po izobraževanju izrazi s strani zaposlenega in je bilo izobraževanje predvideno v LDN, potem se takšen model malenkostno spremeni in se začetek izvedbe prestavi iz dela vodstva v predel zaposlenega.

Četudi gre za poenostavljen model lahko vidimo, da je celoten proces kompleksen od samega začetka do konca izvajanja.

Vidimo lahko tudi, da se večina procesa odvije v samem vodstvu, kjer že sedaj vidimo, da bi tukaj lahko proces poenostavili in zmanjšali ozko grlo. Na drugi strani, kjer gre za zaključek procesa vidimo, da je vodenje izvedbe težko izvedljivo, saj se dokumentacija lahko ohrani v arhivu šole kot dokument v osebni mapi zaposlenega, ali na drugi strani, v lastnem arhivu zaposlenega. Potrdilo o udeležbi je lahko poslano preko navadne pošte ali vročeno udeležencu izobraževanja osebno. Tukaj takoj vidimo problem, in sicer četudi bi zaposleni moral potrdilo izročiti poslovni sekretarki, le-to ne pride nujno do nje. Tudi če je potrdilo izročeno, je uvrščeno v osebno mapo zaposlenega, kar pomeni, da je vodenje težavno, saj je ob vseh zaposlenih skoraj nemogoče voditi stanje izobraževanj. Preprosta oblika vodenja se izvaja v programskem okolju Microsoft Excel, vendar to ne zadošča potrebam.

1.4 Izobraževanja kadrov

Izobraževalni programi, ki se izvajajo na šoli, predstavljajo tehnične poklice, kjer se tehnologije in načini dela neprestano spreminjajo. Za uspešno delo mora zaposleni tako prevzeti odgovornost, ki predstavlja neprestano izobraževanje in izpopolnjevanje. Razvoj zaposlenega je zato pomembna naloga, ki omogoča obvladovanje kakovosti delovnega

procesa izbrane organizacije. V sklopu pričakovanih načinov izobraževanja s strani izbrane šole, so naloge podane tudi v letnem delovnem načrtu šole in predstavljajo:

- Izvajanje novih in prenovljenih izobraževalnih programov šole, povezovanje in razvijanje tehnoloških procesov v gospodarstvu, razvijanje novih učnih tehnologij, upoštevanje šolske zakonodaje ter zmožnosti, ki so vezane na materialne sposobnosti šole.
- Sodelovanje na različnih seminarjih ter tečajih, ki so lahko izvedeni znotraj ali zunaj šole.
- Sodelovanje z zunanjimi izvajalci, ki lahko nudijo svetovanje ali pomoč.
- Dodatno usposabljanje, ki se izvede preko sodelovanja z delodajalcem.
- Izvajanje evalvacije in samoevalvacije, kjer se kakovost dela ocenjuje preko evalvacije ravnateljice, drugih učiteljev ali preko interne komisije za izvajanje kakovosti dela.
- Sodelovanje na strokovnih ekskurzijah, kjer se spozna način dela in okolje znotraj obiskane organizacije.
- Sodelovanje z ostalimi zaposlenimi, delo na projektih šole ali v povezavi s fakultetami, izdelava raziskovalnih nalog, načrtovanje in sodelovanje s strokovnim aktivom itn.

2 ANALIZA TEHNOLOŠKE ARHITEKTURE

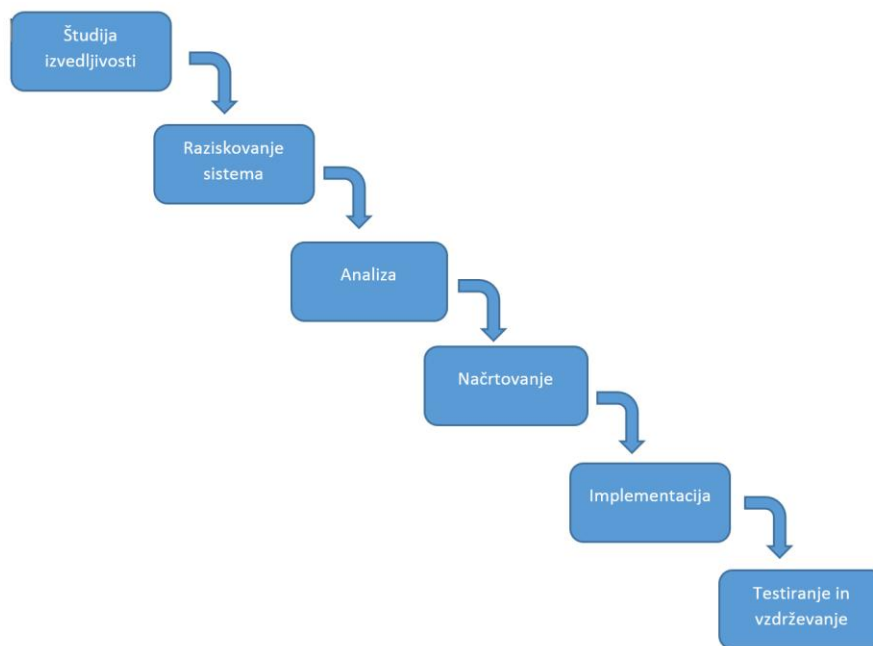
V tej točki, znotraj življenjskega cikla razvoja aplikacije (v nadaljevanju SDLC) modela, je naš cilj izvesti analizo celotne infrastrukture, da bomo lahko načrtovali implementacijo končne rešitve. Učinkovit sistem povezuje samo tehnološko arhitekturo z našimi cilji implementacije sistema, ki bo stroškovno učinkovit, tehnološko ustrezen, prilagodljiv in bo hkrati podpiral tisti poslovni proces, kjer se je po spremembah pokazala potreba (Shelly, Cashman & Rosenblatt, 2008).

Da bi v celoti lahko ugotovili, kakšno je naše okolje in katere tehnologije so nam na voljo, bomo skozi posamezne sklope spoznali celotno infrastrukturo, ki nam je v izbrani organizaciji trenutno na voljo.

Tradicionalne metodologije uporabljajo zaporeden pristop, ki ga predstavlja zaporedje faz. Vsaka naslednja faza se lahko začne šele takrat, ko uspešno zaključimo prejšnjo fazo. Slabost takšnega pristopa je predvsem dolgotrajnost, visoka cena in omejene možnosti spreminjanja. Pristop je primeren, kadar so programske zahteve dobro definirane.

V tem pristopu izdelamo več modelov, med drugim tudi model procesov in model podatkov. Življenjski cikel takšnega modela predstavimo s pomočjo slapovnega diagrama, ki obsega 6 korakov, prikazanih na spodnjem modelu (Gradišar, Jaklič, Damij & Baloh, 2005, str. 270).

Slika 5: Življenjski cikel informacijskega sistema



Vir: A. Gradišar, J. Jaklič, T. Damij & P. Baloh, 2005, *Osnovne poslovne informatike*, str. 270.

Skozi posamezne korake izvedemo preverjanje obstoječega sistema, raziščemo nove zahteve in probleme, ki bi se lahko pojavili in pripravimo možne alternative rešitev ter stroškov izvedbe. Za vsako rešitev moramo predvideti, kakšne bodo omejitve pri izdelavi, naj si bodo to tehnološke omejitve, omejitve okolja, finančne ali zakonske omejitve. Predlagane rešitve morajo biti izvedljive s tehnološkega, ekonomskega, socialnega in organizacijskega vidika. Izbrali bomo tisto rešitev, ki jo bomo ocenili kot najboljšo, poročilo pa predamo vodstvu, v kolikor je predlagana rešitev sprejemljiva. V primeru odobritve lahko preidemo na naslednji korak znotraj modela. V koraku raziskovanja sistema pripravimo funkcionalno specifikacijo sistema in analizo zahtev uporabnika. Opredelimo vrste podatkov in količino podatkov ter probleme obstoječega sistema. Skozi intervjuje z vodstvom šole se preveri izvedljivost in ustreznost rešitve. Analiza sistema je namenjena izdelavi logičnega modela podatkov in procesov sistema, ki jih bomo kasneje uporabili pri načrtovanju. Znotraj načrtovanja bomo logični model pretvorili v načrt končnega sistema. Pomagamo si z diagrami, kjer bomo izvedli analizo vhodnih in izhodnih podatkov.

Znotraj implementacije se bomo lotili izgradnje sistema. V kolikor imamo opravka s programiranjem, je v tej fazi čas za pripravo končne rešitve. Izvedemo tudi nakup in namestitev dodatne programske opreme, v kolikor bi bilo to potrebno. Ob končni implementaciji je potrebno izobraziti tudi uporabnike.

V zadnji fazi testiranja in vzdrževanja zagotavljamo kontinuirano delovanje sistema. Sproti odpravljamo napake, ki se pojavijo in prilagajamo sistem novim zahtevam.

2.1 Strežniško okolje

Na samem začetku, ko se začne načrtovanje projekta in posledično upoštevanje zahtev uporabnika, v našem primeru vodstva šole, moramo izvesti analizo strežniškega okolja, kjer si lahko pomagamo s seznamom preverjanj, ki jih je v sklopu dobre prakse dobro preveriti. Preden sprejmemo končno odločitev, ki bo vplivala tudi na naše trenutno in hkrati končno stanje informacijskega okolja, izvedemo (Shelly, Cashman & Rosenblatt, 2008):

- Pregled celotnih stroškov lastništva (v nadaljevanju TCO), kjer nas zanima vsota stroškov, ki izhajajo iz razvoja, uporabe in vzdrževanja našega informacijskega okolja.
- Razširljivost informacijskega okolja, kjer nas zanima predvsem, kakšne so možnosti širitve našega informacijskega sistema na podlagi trenutne tehnološke infrastrukture.
- Integracija spletne strani, kjer je potrebno zagotoviti ustrezno programsko podporo in hkrati strojno opremo, na kateri bomo omogočili delovanje spletne strani.
- Preverjanje delov sistema, ki so postali zastareli in bi jih bilo potrebno nadgraditi s stališča strojne opreme in programske opreme.
- Zmožnost obdelave podatkov, kjer nas zanima predvsem zmožljivost sistema ob obremenitvah.
- Varnostna vprašanja, ki so vezana na spremembe in vpeljavo nove rešitve v obstoječe informacijsko okolje.

2.1.1 Načrtovanje virov podjetja

Mnogo podjetij danes uporablja programe, ki pomagajo pri načrtovanju virov podjetja (v nadaljevanju ERP). Cilj uporabe ERP sistema je vzpostavitev širše strategije podjetja, ki bo omogočala boljše izkoriščanje informacijskih virov. ERP sistemi določajo neko v naprej zahtevano tehnološko arhitekturo, standarde za shranjevanje podatkov, obdelavo podatkov, omrežno podporo in uporabniški vmesnik. Zaradi v naprej predpisane oz. bolje rečeno zahtevane strojne in programske opreme, pogosto le ti predstavljajo osnovo, ki omogoča enostavno integracijo in hkrati sloni na primeru dobre prakse, ki jo na podlagi izkušenj in obstoječih modelov podjetij, pripravijo velika podjetja, kot so recimo podjetje Microsoft, SAP in ostali veliki ponudniki na tem področju (Shelly, Cashman & Rosenblatt, 2008).

V naši izbrani organizaciji sistem ERP ni integriran. Za uspešno komuniciranje na eni strani med vodstvom šole in zaposlenimi kot tudi na drugi strani med dijaki in starši, je na voljo sistem eAsistent, ki se izvaja na strežnikih zunaj šolskega okolja in predstavlja enega izmed sistemov obveščanja, kot tudi sistemov komunikacije med vsemi udeleženci. Zaradi narave dela ni bilo želje in izkazane potrebe po vpeljavi ERP sistema v šolsko okolje, saj so tudi povezave, ki so vezane na dobavitelje omejene z zakoni in postopki, kjer takšen sistem ne bi imel vpliva na odločanje oz. ne bi omogočal izboljšanja pri izvedbi dnevnih opravil.

2.1.2 Pregled celotnih stroškov lastništva

V ospredju odločanja so pogosto ravno stroški. V povezavi z izbiro rešitve se moramo vprašati ali je izdelava aplikacije res boljša od nakupa rešitve s prilagoditvijo. V kolikor imamo v uporabi kakšen poseben programski paket ali dodatek se seveda vprašamo ali je ta potreben in ali je bil res najboljša izbira v preteklosti, je morda na voljo kakšen zunanji ponudnik, ki bi nam rešitev lahko na podlagi več izkušenj izdelal ceneje in v krajšem roku, ali se je spremenila zakonodaja, ki bi morda vplivala na izbor rešitve in še bi lahko naštevali.

Izbira neke rešitve lahko pomembno vpliva na končne stroške, ki bodo nastali zaradi implementacije nove rešitve. Organizacija Gartner pripravlja teste in pregled rešitev, kjer si velja vzeti čas in malce preveriti njihove analize trenutnih rešitev, ki bi nam lahko prišle prav.

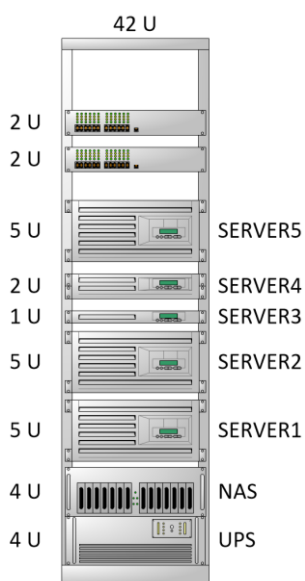
Skozi pregled programske opreme, ki sledi v točki 2.3 si bomo ogledali, katere rešitve so že implementirane v samem informacijske okolju in katere so tiste, ki bi lahko pomenile dodatni strošek organizaciji.

2.1.3 Razširljivost informacijskega okolja

Razširljivost informacijskega sistema predstavlja zmožnost, da naš obstoječi informacijski sistem razširimo glede na nove potrebe in zahteve kot posledico sprememb obstoječih poslovnih procesov ali uvajanja novih poslovnih procesov. Razširljiv sistem je pomemben predvsem, kadar gre za dinamično spreminjajoče se informacijsko okolje ali organizacijo, ki se širi.

Pogledali si bomo, kako je znotraj naše organizacije poskrbljeno za zmožnost širitve na področju strežniškega okolja. Na naslednji sliki lahko vidimo prikazano strežniško omaro v izbranem okolju.

Slika 6: Prikaz strežniške omare



Na prejšnji sliki lahko vidimo, da gre za postavitev, v kolikor gledamo od spodaj navzgor po komponentah:

- Brezprekinitveno napajanje (v nadaljevanju UPS),
- omrežno diskovno polje (v nadaljevanju NAS),
- strežnik 1 (virtualizacija),
- strežnik 2 (virtualizacija),
- strežnik 3 (sekundarni domenski strežnik),
- strežnik 4,
- strežnik 5.

Oznake 1 U do 5 U predstavljajo višino posameznega strežnika. V povezavi z višino je običajno povezan prostor oz. število možnih diskov za shranjevanje podatkov in posledično tudi prostor za vgradnjo dodatnih komponent, kot so omrežne kartice.

Vsak izmed strežnikov ima neko dodeljeno vlogo. S stališča analize je sicer pomembno, kakšne so vloge strežnikov, ki so bili zapisani, vendar zaradi zaupnosti povemo le toliko, da strežnik 3 predstavlja alternativni strežnik, ki skrbi za dinamično dodeljevanje naslovov IP (v nadaljevanju DHCP), sistem domenskih imen (v nadaljevanju DNS), aktivni imenik (v nadaljevanju AD) in ostale funkcije, ki so nameščene na enem izmed virtualiziranih strežnikov. Strežnik 4 opravlja funkcijo nosilca različnih mrežnih aplikacij in dodeljevanja licenc uporabnikom, medtem ko strežnik 5 skrbi za zajem in shranjevanje video vsebin kamer.

Za naše potrebe analize in tudi implementacije rešitve nam preostane le omrežno diskovno polje skupaj s strežnikom 1 in strežnikom 2. Posvetili se bomo načinu povezave in delovanju le teh, saj so najpomembnejši za končno implementacijo rešitve.

Kot je prikazano na shemi strežniške omare, so omrežno diskovno polje in strežnika 1 in 2 v strežniški omari skupaj. Med seboj so povezani preko štirih 1 Gbit/s povezav, ki omogočajo visoke hitrosti prenosa in izmenjave podatkov. Omrežno diskovno polje je razdeljeno v tri hitrostne razrede, in sicer:

- Trajno shranjevanje podatkov na najhitrejših diskih s 15.000 vrtljaji na minuto, kjer se podatki odpirajo pogosto in mora biti odzivnost visoka.
- Trajno shranjevanje podatkov na srednji razred diskov z 10.000 vrtljaji na minuto, kjer se podatki pogosto osvežujejo in spreminjajo.
- Trajno shranjevanje podatkov, kjer gre za shranjevanje arhivov in dokumentov z redkimi dostopi ter zato zadoščajo diski s 7200 vrtljaji na minuto.

Omrežno diskovno polje je nosilec shranjevanja vseh podatkov in hkrati virtualiziranih sistemov. Kot je opredelil Killmeyer Tudor (2001, str. 101 - 138), predstavljajo varnostne kopije ključno odgovornost organizacije, ki se nanaša na kritične poslovne procese. Varnostne kopije so zahtevane, saj le tako lahko poskrbimo za obnovitev virtualiziranih sistemov, programske rešitve, podatkovne baze ali druge podatke. Da je ustrezno poskrbljeno za varnostno shranjevanje, se najpomembnejši podatki shranjujejo na dodatno omrežno diskovno polje. Virtualizirani sistemi so shranjeni na NAS napravi. Zagon le-teh se izvede preko enega izmed strežnikov 1 ali 2. Kateri strežnik bo zagnal virtualno sliko niti ni pomembno, saj sta oba strežnika enakih lastnosti, torej imata enako procesno enoto, količino pomnilnika, enako velikost in število diskov itn. Med strežnikoma se izvaja samodejno preklapljanje, ki samodejno zazna izpad enega strežnika in poskrbi za prehod na drugega v nekaj sekundah. Programska oprema VMware ESXi je rešitev, ki je nameščena kot osnova virtualizaciji na obeh strežnikih.

Programska oprema VMware ESXi omogoča, da lahko vzpostavimo različne virtualne strežnike, torej Microsoft Windows, Linux, OpenBSD ali druge. V konkretnem primeru so vsi virtualizirani strežniki Microsoft Server 2008 R2 ali Microsoft Server 2012 R2.

Tako je poskrbljeno, da bo sistem dostopen večino časa, da bi pa tako ostalo tudi po izpadu omrežnega diskovnega polja v primeru napake ali težav, se virtualizirani sistemi vsako noč varnostno kopirajo na manjši NAS omrežni sistem.

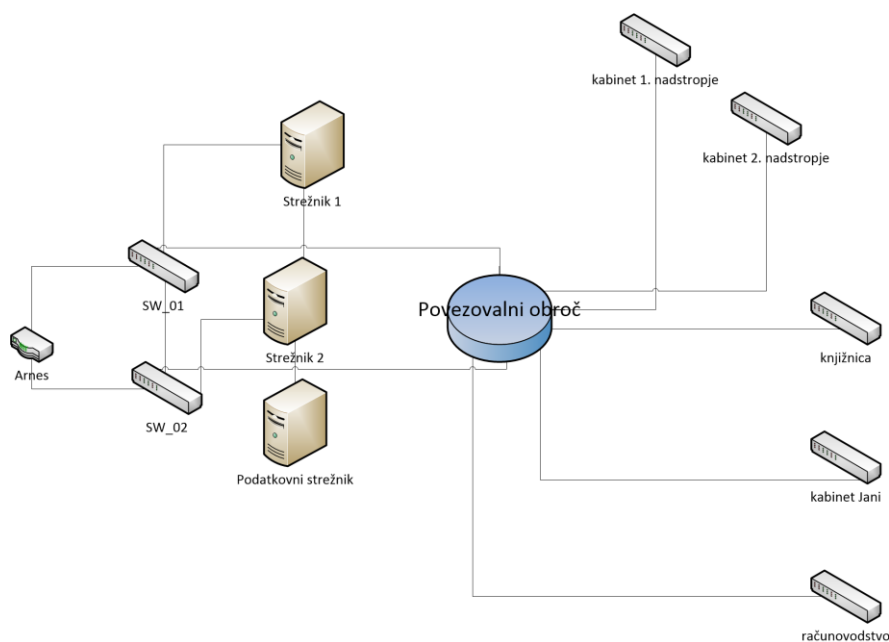
2.2 Krajevno omrežje

Krajevno omrežje nam omogoča, da znotraj prostora, kot je nekaj povezanih prostorov, hiša ali večje poslopje, med seboj povežemo strojno opremo, programsko opremo, omogočimo dostopnost do podatkovnih baz ali ostalih virov, kar nam zmanjša stroške in poveča večjo fleksibilnost uporabnikov. Pri načrtovanju krajevnega omrežja se moramo odločiti za model in koncept, upoštevajoč tako ISO/OSI model, omrežna orodja za načrtovanje, omrežno topologijo, licence, brezžično omrežje itn. Vprašati se moramo tudi, kako bomo zagotavljali ustrezno varnost našega omrežja, zmožnost povezave ustreznega števila naprav in uporabnikov ter prekinitve, ki bi se lahko zgodile med delovanjem (Shelly, Cashman & Rosenblatt, 2008).

Poznamo orodja, ki nam omogočajo načrtovanje krajevnega omrežja. Omrežje lahko s pomočjo sheme predvidimo, kot je to mogoče s programom Microsoft Visio ali na drugi strani izdelamo manj ali bolj podroben model, ki bo prikazoval naše krajevno omrežje. Primer takega orodja je brezplačno orodje GNS3 ali na drugi strani Cisco Packet Tracer, ki je vezan na Cisco omrežno opremo.

V našem primeru je bilo uporabljeno orodje Microsoft Visio, saj je znotraj naše organizacije orodje na voljo v sklopu dostopnih licenc in hkrati je samo omrežje kompleksno, kar pomeni, da bi bilo potrebno veliko dodatnega dela za izris in celotno konfiguracijo za prikaz delovanja.

Slika 7: Poenostavljena shema omrežja



Kot lahko vidimo na zgornji sliki, sta pri vhodnem signalu dve glavni stikali, ki sta med seboj povezani in tako zagotavljata redundanco v primeru izpada enega stikala. Povezovalni obroč je namenjen povezavi stikal po krajevnih vozliščih, kjer so stikala primarno povezana do glavnih stikal preko optičnih vlaken in s tem tvorijo hrbtenico omrežja, zaradi zanesljivosti pa so povezana med seboj preko redundance preko bakrenega medija.

Ker gre za množico vozlišč, ki jih omrežimo z velikim številom različnih topologij, kjer ima vsaka svoje prednosti in slabosti, si na kratko pogledimo značilnosti najpogostejših.

2.2.1 Topologije omrežij

Pri odločanju za ustrezno topologijo se moramo opreti na več različnih dejavnikov. Vedno nas zanima, koliko skokov bo potrebnih, da pridemo do cilja. Manj skokov običajno pomeni krajšo pot in hkrati preprostejšo konfiguracijo omrežnih naprav. Razmisliti moramo tudi o alternativnih poteh, saj se lahko zgodi, da bo ena izmed naprav v omrežju prenehala delovati. To lahko vodi v nedelovanje dela omrežja, kar seveda ni primerno. Rešitev se lahko skriva v redundantnih povezavah ali v alternativnih povezavah. Zelo pomemben dejavnik je tudi število vozlišč, kjer govorimo o številu pristopnih točk in število lokalnih vozlišč (Vidmar, 2002, str. 667 - 739).

V našem konkretnem imamo glavno vozlišče, v katerega so speljane povezave iz krajevnih vozlišč in sedem krajevnih vozlišč, ki s pomočjo dodatnih stikal omogočajo zadostno število vstopnih točk za končne uporabnike.

Najpogosteje uporabljene topologije so (Vidmar, 2002, str. 667 - 739):

- Topologija vodila (bus): Naprave na omrežju si delijo isti fizični medij.

- Zvezdna topologija (star): Omogoča nam enostavno dodajanje novih postaj, saj so naprave povezave v skupno centralno vozlišče. Slabost je izpad centralnega vozlišča.
- Topologija obroča (ring): Vozlišča nanizamo enega za drugim v obroč. V primeru izpada pride do motenj delovanja le pri lokalnih uporabnikih izpadlega vozlišča.
- Drevesna topologija (tree): Drevesno topologijo si lahko predstavljamo kot več nivojsko zvezdo.
- Polna topologija: Zagotavlja neposredne povezave med vsemi pari vozlišč. Usmerjevalni postopki so zelo zahtevni, saj število možnih poti med vozlišči hitro narašča.
- Splošna topologija: V praksi je najpogostejša, saj vsebuje poljubno število povezav popolne topologije, ki še zagotavlja povezanost omrežja.
- Topologija s skupinskim medijem: Potrebujemo le oddajnik in sprejemnik med dvema napravama. Gre za zelo preprost način izmenjave podatkov med napravama.

Na podlagi kratkega vpogleda v topologije in pred tem zapisanega vidimo, da je topologija v izbrani organizaciji v osnovi krog, ki ima dodatne povezave zaradi alternativnih poti v omrežju, kar pomeni, da govorimo o splošni topologiji omrežja. Primarne povezave so izvedene preko optičnih vlaken, alternativne pa preko med seboj povezanih bakrenih medijev, ki predstavljajo oklopljeno sukano parico (v nadaljevanju STP), ki ima dodatno zaščitno pred vplivi in motnjami iz okolja. Zaradi načina povezav je potrebno vzpostaviti tudi protokol STP saj le tako lahko zagotovimo, da alternativne povezave ne bi predstavljale sočasnega oddajanja v omrežju, kar bi vodilo v poplavo podatkov in posledično nedelujoče omrežje, zaradi preobremenitve stikal.

2.2.2 Varnost krajevnega omrežja

Osnovna zaščita omrežja s pomočjo požarnega zidu nikakor ni dovolj. Znotraj omrežja je potrebno poskrbeti za zaščito, ki bo onemogočala izvajanje nelegalnih storitev. To pomeni, da moramo zagotoviti, da se znotraj našega sistema ne izvajajo storitve in dostopi, ki niso bili predvideni in dovoljeni (Vidmar, 2002).

Pri izdelavi varnostnega načrta, kjer bo v uporabi izmenjava podatkov v obliki spletne aplikacije, morajo sodelovati specialisti informatiki, management in posamezni predstavniki zaposlenih. Ko je kreiran ustrezen tim, moramo izvesti sledeče korake (Prešern, 2006, str. 111 - 117):

- Prepoznati vse možne nevarnosti, kjer se upoštevajo tako naravne nesreče kot človeški dejavniki,
- izdelati varnostni načrt, vključno z načeli in postopki za izvedbo načel,
- varnostna politika, ki predstavlja krovni dokument in podaja osnovna navodila za e-varnost,
- postopki ali navodila za varnost, kjer se v praksi izvede preverjanje načel.

Preden se bomo lotili konfiguracije sprememb požarnega zidu, bomo povečanje varnosti zagotovili tudi z ločevanjem omrežij po različnih segmentih, pravimo jim navidezna krajevna omrežja (v nadaljevanju VLAN). S pomočjo le-teh bomo zagotovili, da se bodo preko istih medijev prenašali podatki, ki spadajo v različna omrežja. S tem bomo poskrbeli za zaščito in preprečili, da bi nepooblaščen uporabnik videl ostala omrežja in s tem podatke, ki se nahajajo znotraj omrežji.

S tem namenom so znotraj našega omrežja s pomočjo VLAN-ov medsebojni ločeni različni segmenti:

- VLAN XX: MANAGEMENT: Upravljanje celotnega omrežja in priklop strežnikov ter ostalih naprav.
- VLAN XX: UPRAVA: Predstavlja uporabnike, kot so ravnateljica, pomočnik ravnateljice, tajništvo, računovodstvo, itn.
- VLAN XX: UČITELJ: Povezovanje naprav in računalnikov, ki jih uporabljajo učitelji pri svojem delu.
- VLAN XX: PEDAGOŠKI: Povezovanje naprav in računalnikov, ki jih pri svojem delu uporabljajo dijaki šole.
- VLAN XX: KABINET: Povezovanje naprav in računalnikov, ki so v kabinetih zaposlenih.
- VLAN XX: EDUROAM: Povezovanje naprav v brezžično omrežje, ki je na voljo zaposlenim, dijakom šole ter obiskovalcem, ki imajo svoje dostopne podatke za povezavo v Eduroam omrežje.
- VLAN XX: INTERNET: Ločeno podomrežje, kjer se izvaja prenos podatkov med glavnim vozliščem in vstopno točko interneta.
- VLAN XX: DMZ: Del omrežja, kjer je bil v uporabi spletni strežnik. Zaradi varnosti ločen v svoje podomrežje.
- VLAN XX: IP_TEL: Ločeno omrežje za povezovanje IP telefonskih naprav.
- VLAN XX: VIDEO: Ločeno omrežje, za povezavo IP kamer po celotni šoli.

Zaradi same varnosti konkretnih naslovov IP in številke VLAN-ov ne bomo podajali. Da so omrežja ustrezno ločena in se ne vidijo med seboj, je v uporabi požarni zid SonicWall. Leta skrbi tudi za nadzor in beleženje prometa ter omogoča, da se dohodne in izhodne povezave ustrezno filtrirajo, glede na podana pravila in nastavitve.

Kadar govorimo o varnosti, se moramo zavedati tudi pričakovane letne izgube kot posledice neželenih izgub podatkov, vdorov ali napak uporabnikov. Izračun vrednosti označimo z ALE, kar predstavlja pričakovano vrednost izgub (Bojanc, Jerman-Blažič & Tekavčič, 2014, str. 30 - 68):

$$ALE(I) = \sum_{i=1}^n Li(I)Pi(I) \quad (1)$$

Kot je razvidno iz enačbe (1), je število vseh dogodkov mogoče izračunati kot zmnožkov vseh verjetnosti za dogodek in letne izgube v nekem obdobju.

n = število možnih dogodkov.

i = vsak možen dogodek znotraj enega leta.

I = možna investicija v informacijsko varnost ali določeno skupino. Investicija I lahko vpliva na verjetnost, da se bo dogodek zgodil v danem letu.

$P_i(I)$ = verjetnost za dogodek i v obdobju enega leta ob izvedeni investiciji I

$L_i(I)$ = letna izguba v denarni enoti, ki bi prizadela podjetje, če bi se zgodil dogodek.

Predpostavimo, da omrežja ne bi ločili po VLAN-ih, kar bi pomenilo, da bi virus znotraj okolja lahko nemoteno potoval po celotnem krajevnem omrežju. V takem primeru bi glede na zgornjo enačbo izvedli izračun: nedelovanje spletne strani in vseh ostalih komponent krajevnega omrežja bi predstavljalo 10.000 € izgube in bi se pripetila 1 krat na 2 leti (0,5 verjetnost za okužbo). ALE bi v danem primeru znašal $10.000 \text{ €} \times 0,5 = 5.000 \text{ €}$. Požarni

zid, ki zna delati z VLAN-i in hkrati omogoča filtriranje prometa ter nadzor nad okužbami, nas v enkratnem znesku stane manj, cena pa je odvisna od funkcionalnosti in zmožnosti podpore glede na število uporabnikov, izbire rešitve in tudi dodatnih možnosti konfiguracije.

Kot je zapisal Harkins (2013), ne smemo pozabiti tudi na človeški dejavnik, kjer lahko zaposleni zavedno ali nezavedno vplivajo na delovanje informacijskega sistema, izgubo podatkov zaradi spreminjanja, premikanja ali celo namernega brisanja. Fizične poškodbe s strani zaposlenih so preprečene s pomočjo kovinskih omaric, ki so zaklenjene, na drugi strani pa je preprečen tudi dostop do strežniške sobe nepooblaščenim osebam. S pomočjo ločenih omrežij je preprečeno prehajanje uporabnikov med omrežji in s tem dostopnost in omejevanje podatkov, ki je rešeno tudi s pomočjo aktivnega imenika in ustreznih dostopnih pravic.

2.3 Programska oprema in licence

Računalniški programi predstavljajo zaporedja programskih ukazov, lahko bi rekli tudi stavkov, ki omogočajo izvajanje nekih, v naprej predvidenih, operacij. Glede na probleme, ki jih rešujejo, poznamo tri vrste programov (Gradišar, Jaklič, Damij & Baloh, 2005, str. 79 - 85):

- Namenski ali uporabniški programi,
- programska orodja za gradnjo namenskih programov,
- sistemski programi.

2.3.1 Namenski ali uporabniški programi

Namenski ali uporabniški so tisti programi, ki rešujejo specifične probleme oz. so v pomoč in zabavo njihovim uporabnikom. V to kategorijo sodijo programi, ki nam omogočajo pisanje, delo in urejanje preglednic, programi za urejanje in izdelavo predstavitev itn. Programi so razviti z namenom, da se neko rutinsko ponavljanje dela pohitri in hkrati uporabniku omogoči lažje ter hitrejše delo z njimi. Programi lahko rešujejo nek specifičen problem ali so bolj splošni. Ker so programi napisani za večje število uporabnikov, je cena na posameznega uporabnika nižja, kar pa na drugi strani vodi v standardizacijo rešitev, ki morajo biti prilagojene množicam in ne posameznikom ter njihovim specifičnim zahtevam (Gradišar, Jaklič, Damij & Baloh, 2005).

2.3.2 Programska orodja za izgradnjo namenskih programov

Danes nam je na voljo kup orodij, ki jih zna uporabiti vsak, ki ima nekaj znanja iz področja informatike. V praksi bi to pomenilo, da si posameznik, ki ima takšno znanje, zna izdelati preprost program, ki ga potrebuje pri svojem delu. V takšen nabor bi ponovno sodila orodja za upravljanje in delo s preglednicami, programi za delo z manjšimi podatkovnimi bazami, obdelovanje fotografij itn. Takšnim orodjem zato pravimo tudi orodja za končne uporabnike. V kolikor so orodja kompleksnejša, potem bo kljub vsemu potrebno za pomoč prositi informatika, ki nam bo pomagal pri izdelavi rešitve. Med takšna orodja sodijo

sistemi za upravljanje s podatkovnimi bazami, sistemi za računalniško podprt sistemski inženiring itn. (Gradišar, Jaklič, Damij & Baloh, 2005).

2.3.3 Sistemski programi

Sistemski programi so namenjeni krmiljenju, nadzorovanju in usklajevanju računalniškega sistema, da ta deluje pravilno in ga uporabnik lahko uporablja. Sistemske programe pišejo programerji, ki dobro poznajo delovanje računalniškega sistema. Sem sodijo vsi operacijski sistemi in servisni programi. Operacijski sistem nadzoruje izvajanje uporabniških programov in skrbi za dodeljevanje računalniških virov. Primer takega vira je upravljanje s prostorom na trdem disku. V osebnih računalnikih je danes v uporabi najpogosteje operacijski sistem Microsoft Windows, ki je plačljiv. Njegova alternativa je vedno pogosteje izbrana brezplačna rešitev, ki temelji na Linux osnovi.

2.3.4 Programska oprema in licence v organizaciji

Kot smo že omenili, je v izbrani organizaciji na voljo sistem dveh strežnikov in omrežnega diskovnega polja, kjer gre za podporo virtualizaciji z enotno platformo VMware in možnostjo redundance. Programska oprema VMware ESXi izboljša izkoriščenost strojne opreme, saj omogoča sočasno delovanje več različnih virtualiziranih sistemov in hkrati je upravljanje na enem mestu hitrejše in enostavnejše (VMware - Virtualization, 2015).

Uporabljena programska oprema ni brezplačna. V naši organizaciji je bila kupljena licenca za uporabo na dveh strežnikih skupaj z omrežnim diskovnim poljem, kar pomeni, da ob implementaciji naše končne rešitve, licence ne bo potrebno nadgraditi ali kupiti in s tega vidika ne pričakujemo dodatnih stroškov.

Na drugi strani so v uporabi virtualizirani strežniki, ki za svoje delovanje uporabljajo operacijski sistem Windows Server 2008 R2 ali Windows Server 2012 R2. Izbira platforme je v preteklosti izvirala iz že uporabljenih rešitev in funkcionalnosti, ki so bile vezane na stari sistem pred posodobitvijo v šolskem letu 2011/2012. Ker gre za izobraževalno organizacijo, je nakup licenc Windows možen po pogodbi z Microsoft, in sicer gre za pogodbo Microsoft School Agreement. Cena posamezne licence operacijskega sistema server tako predstavlja strošek, ki se plačuje letno. Govorimo torej o najemu letne licence, ki za Windows Server standard predstavlja vrednost 54,90 €/letno z vključenim davkom na dodano vrednost. V kolikor bi uvedba nove rešitve predvidela tudi nov virtualizirani strežnik, moramo ceno najema upoštevati v ceni končne rešitve.

Pomembno je tudi povedati, da trenutno ni implementirana nobena rešitev, ki bi zahtevala uporabo podatkovne baze. Na voljo imamo brezplačno rešitev MySQL ali na drugi strani Microsoft rešitev, glede na dejstvo, da so vsi programi od Microsofta. Rešitev bi bila v tem primeru plačljiva, in sicer govorimo o najemu programske opreme SQL Server Standard, ki bi nas stala dodatnih 95,16 € letno (Cenik licenc po pogodbi School Agreement, 2015).

Šola ima tudi program Microsoft Dreamspark, kjer je na voljo programska oprema podjetja Microsoft za potrebe učenja in uporabe doma tako za dijake kot tudi za zaposlene. V okviru sistema Dreamspark nam je na voljo programsko orodje Visual Studio, ki je namenjeno programiranju aplikacij in spletnih rešitev. Programsko okolje lahko uporabimo

pri izdelavi in podpori našega problema, ki ga bomo poskušali v tej nalogi rešiti. Na voljo so nam različni programski jeziki okolja .NET.

V tem delu velja omeniti tudi, da je v sklopu storitev, ki jih šoli zagotavlja Microsoft na voljo tudi oblačni paket Office 365, ki ponuja:

- Spletno pošto Web Exchange,
- shranjevanje podatkov v oblak OneDrive,
- urejanje dokumentov Word, Excel, Power Point,
- uporabe SharePoint in
- Skype za poslovno uporabo.

Ker se je pred uvedbo te možnosti že uporabljala Google rešitev, ki omogoča upravljanje z lastno domeno v povezavi z elektronsko pošto, Microsoft rešitev ni povezana z lastno domeno ssts.si, vendar je v uporabi začasna domena, ki jo nudi Microsoft na svojem naslovu onmicrosoft.com.

2.4 Obstoječe rešitve

Ob pregledu obstoječega stanja lahko kaj hitro ugotovimo, da je krajevna infrastruktura prenovljena in težav pri dodajanju nove storitve uporabnikom ne bo. Tudi strojna oprema je dovolj zmogljiva, da lahko vključimo nove rešitve, ki bodo omogočale izvedbo informatizacije poslovnega procesa izobraževanj zaposlenih.

Največji izziv je pravzaprav izbira prave tehnologije glede na obstoječe stanje, ki bo stroškovno sprejemljivo, a hkrati strošek potrebne strojne opreme ali programske opreme ne bo visok tako v fazi implementacije, kot kasneje skozi daljše obdobje, kjer bo potrebno sistem nadgrajevati in vzdrževati.

Ker trenutno ni nobene implementirane rešitve, kot bi bil intranet ali katera druga storitev, je sama zasnova in iskanje ustrezne rešitve enostavnejše. Kljub vsemu, se je potrebno zavedati, da bo potrebno kasneje na obstoječem sistemu pripraviti tudi rešitev, ki bo omogočala nadzor napak in težav v strokovnih učilnicah, s čimer se bo zagotavljalo obveščanje administratorja o težavah ali na drugi strani o poškodbah, ki so namerno ali nenamerno nastale ob uporabi opreme.

2.5 Ocena stanja in možnosti prehoda

Na podlagi izvedene analize tehnološke arhitekture bomo v nadaljevanju podali ključna področja prenove in informatizacije.

2.5.1 Ključni dejavniki uspeha in analiza informacijskih potreb

Kratica SISP predstavlja »Strategic information system planning«, ki predstavlja proces identifikacije računalniško prilagojenih aplikacij, ki nam bodo pomagale pri izvajanju poslovnih opravil in doseganju poslovnih ciljev. SISP predstavlja tudi določanje

podatkovnih baz in sistemov za podporo aplikacijam. Ključni dejavniki, ki nam bodo zagotovili uspešno implementacijo, so tako identifikacija ključnih dejavnikov in strateških aplikacij ter uskladitev poslovnih potreb znotraj organizacije (Lederer & Sethi, 1996, str. 48).

Skozi analizo uspeha v slovenskem strateškem načrtovanju informatike je bilo ugotovljeno, da so ključni dejavniki uspeha odvisni predvsem od sodelovanja vrhnjega in srednjega managementa. Pomembno je tudi, da sta informatika in njen razvoj povezana in usmerjena v skupni cilj z managementom, kar bo skupaj pomagalo k uspešnosti in konkurenčni prednosti (Hovelja, Rožanec & Rupnik, 2010).

V okviru analize informacijskih potreb ter pregleda obstoječega stanja je bilo pomembno tako zbiranje gradiva kot informacij o tem, kaj je trenutno v uporabi znotraj našega informacijskega sistema. Vodstvo šole je podalo željo, da bi se poslovni proces, ki je konkretno vezan na izobraževanje zaposlenih v organizaciji, podprlo s pomočjo informacijskega sistema. Skozi razgovor so bile predstavljene in opredeljene temeljne značilnosti, ki bi jih bilo potrebno pokriti z novo rešitvijo. Glede na analizo in razgovore, ki so bili opravljeni, bomo podali podrobnejša razvojna izhodišča.

Z naraščanjem števila dijakov in s tem tudi števila zaposlenih, je bila dosežena stopnja, kjer vodenje izobraževanj zaposlenih preko orodja Microsoft Excel ni več učinkovito in časovno sprejemljivo. Potreba po vključevanju dokumentov v sistem, pregleden in z ustreznim izpisom je pripeljala vodstvo do točke, kjer je potrebno najti novo rešitev. Potrebna je uvedba nove funkcionalnosti, ki hkrati ne bi vplivala na trenutno delo in sočasni razvoj nove rešitve. Nova rešitev bi morala odpraviti težave, vezane na delo z orodjem Excel.

Težave dela z orodjem Microsoft Excel bi lahko strnili v nekaj točkah:

- Nepreglednost po podatkih za zaposlenega v tekočem letu ali v preteklih letih,
- nezmožnost dodajanja medoperacijsko prenosljive datoteke (v nadaljevanju PDF) ali DOC (Microsoft Word) oblike potrdil o izvedbi izobraževanja,
- nezmožnost sočasnega urejanja in pregledovanja dokumenta,
- zaposlenim je vpogled v dokument onemogočen, saj bi sicer videli podatke o vseh zaposlenih, česar si ne želimo,
- zahteva po preglednejših podatkih in zmožnost priprave poročila.

Zavedati se je potrebno, da zahteve in želje vodstva ne pomenijo nujno uspešnega projekta. Verjetnost, da bodo projekti propadli že v sami fazi razvoja je velika predvsem pri projektih, ki se najbolj razlikujejo od idealnega projekta z najnižjim možnim tveganjem. Na drugi strani je mnogo takšnih projektov, ki se sicer prebijejo skozi fazo razvoja, vendar pri uvajanju v prakso ne dosežejo pričakovanih koristi. Vzrok za neuspeh je lahko pomanjkljiva komunikacija med uporabniki in razvijalci sistema in hkrati slabo nadzorovanje poteka napredka in razvoja sistema (Gradišar, 2003).

V članku Implementing BPM systems (Reijers, 2006) je avtor navedel, da je pri uvajanju poslovnih procesov (v nadaljevanju BPM) ali sorodnih sistemov pomembno, da upoštevamo seznam področij, ki jih preverimo znotraj postopka ocen stanja in zahtev: Tehnologija, management podjetja, človeški viri in procesi.

Vse omenjeno bomo v skladu z zahtevami tudi preverili v sklopu celotne naloge in iz tega izpeljali končno rešitev.

2.5.2 Trenutno tehnološko okolje

Analiza tehnološke arhitekture je pokazala, da je bilo krajevno omrežje v celoti prenovljeno in omogoča širitev, v kolikor bi se izkazala potreba po dodajanju krajevnih vozlišč ali dodatnih stikal znotraj omrežja. Strojna oprema je sodobna in podpira nove vidike in pristope, kot je virtualizacija računalnikov ali programov. Sama varnost omrežja je na ustreznem nivoju, saj omogoča sledenje uporabniku glede na uporabniško ime ali glede na njegov naslov IP. Podatki o dogodkih se shranjujejo in so ustrezno ločeni od dela omrežja, kjer so vanj povezani ostali uporabniki. Dodatna varnost je zagotovljena s pomočjo virtualnih omrežij in požarnim zidom, ki omrežja med seboj ločuje in zagotavlja ustrezen nadzor nad prometom in izvajanjem aplikacij.

Znotraj organizacije je v uporabi več aplikativnih rešitev, ki so od različnih proizvajalcev. Primeri takih rešitev so:

- Špica, za nadzor prihoda in odhoda zaposlenih,
- Intellinet, za nadzor in prikaz video vsebin na zaslonih,
- Dreamspark, ki deluje kot oblachna storitev,
- računovodski program SAOP,
- itn.

Nekateri ključni procesi, kot je tudi proces izobraževanj zaposlenih, so podprti s pomočjo splošnih programov za množice uporabnikov, kot je npr. Microsoft Excel. Načini vnosa podatkov so različni, preglednice se ne nahajajo na skupni lokaciji, četudi tehnološka arhitektura to omogoča. Podatkovna baza, ki bi reševala shranjevanje različnih podatkov na skupnem mestu z enostavnim sistemom varnostnega kopiranja, ni v uporabi. Posledica zapisanega so težave pri upravljanju, arhiviranju, sočasnih dostopih in upravljanju pravic. Takšen sistem dela lahko vodi tudi v stanje, ko podatki med seboj niso konsistentni.

Ker trenutno stanje ne omogoča učinkovitega vodenja procesa izobraževanj, je bila izražena zahteva vodstva, da se pripravi rešitev, ki bi omogočala učinkovitejše delo.

Ustrezen nivo informatizacije bo na podlagi analize lahko zagotovljen zgolj z dodajanjem nove informacijske rešitve in prenova informacijskega okolja v večjem obsegu ne bo potrebna.

2.5.3 Poslovno modeliranje ključnih procesov

V preteklih poglavjih smo si ogledali osnovne značilnosti našega informacijskega okolja, kar je tudi ključno izhodišče, za načrtovanje in prenova informatizacije. Izbira ustrezne rešitve mora temeljiti na pravilnem pristopu podpore poslovnega procesa, oblikovanja ustrezne rešitve, programiranja in končne implementacije rešitve. Obstaja mnogo različnih načinov, kako implementirati končno rešitev. V kolikor je rešitev v večjem delu pokrita z neko rešitvijo, ki je splošno razširjena, moramo razmisliti o nakupu rešitve, ki bo podprla naš poslovni proces. Tudi v kolikor rešitev ni splošna je bila morda podobna rešitev že pripravljena in bi jo lahko kupili ali po potrebi preoblikovali za naše potrebe. Če ugotovimo, da podobne rešitve na trgu ni na voljo, nam ne preostane nič drugega, kot da se lotimo izdelave lastne rešitve, ki pa jo lahko naredimo sami ali najamemo zunanje podjetje za izdelavo. Najem zunanjega podjetja oz. programerja lahko predstavlja izbiro, kjer bo

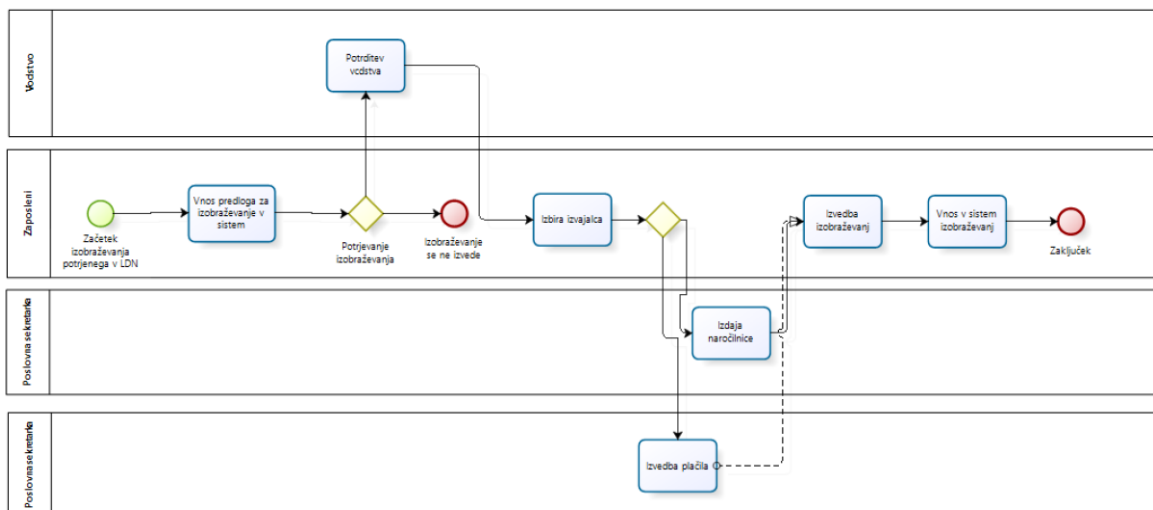
strokovnjak tega področja pripravil ustrezno rešitev glede na naše zahteve. Specializirana oseba lahko pripravi končno rešitev ali nam izvede tudi implementacijo skupaj z izobraževanjem zaposlenih (Satzinger, Jackson & Burd, 2009).

Ključni proces, ki ga je v sklopu te naloge potrebno informacijsko podpreti, predstavlja proces izobraževanj zaposlenih. S pomočjo analize procesa, ki smo ga izvedli, smo naredili posnetek trenutnega stanja. Izvedli smo poslovno modeliranje, ki je bilo pripravljeno grafično s pomočjo orodja BizAgi.

Postopek modeliranja in prenove poslovnega procesa zahteva, da preverimo dejansko stanje in izvedemo posnetek trenutnega stanja. Na podlagi modela izdelamo model procesa kot je oz. »AS-IS«. Opredeliti moramo izhodišča za izboljšave in na podlagi pripravljenega modela izvesti analiziranje trenutne situacije, ki nas bo vodila v izboljšanje procesov. Pripravimo model, kot naj bo oz. »TO-BE« in pripravimo končno načrtovano stanje (Kovačič & Boksilj Vukšić, 2005, str. 181). Analizo trenutnega stanja smo pripravili na podlagi pogovorov in trenutne dokumentacije, ki nam je bila na voljo, upoštevajoč zapisana pravila izvajanja aktivnosti vključno z vsemi aktivnostmi, ki so del našega procesa.

Na podlagi izdelanega modela in tehnološkega okolja je bila izdelana optimizacija in informatizacija procesa. S pomočjo simulacije, ki nam jo nudi orodje BizAgi, smo izvedli analizo in prišli do ozkega grla obstoječega poslovnega procesa. Na podlagi dobljenih rezultatov smo izvedli optimizacijo, ki bo poskrbela za hitrejši dostop do podatkov, možnost hitrejšega ter lažjega odločanja vodstva. Hkrati bodo imeli zaposleni vpogled v trenutno stanje izobraževanj ter možnost enotnega vmesnika za vpogled in vodenje statistike izobraževanj.

Slika 8: Poslovni proces izobraževanj TO-BE



Prikazani poslovni proces predstavlja poenostavljen sistem izobraževanja in hkrati vključuje ključno spremembo, in sicer pripravljeno rešitev, ki bo informatizirala celoten proces izobraževanj in olajšala nadzor nad izobraževanji.

2.5.4 Predlagana prenova s pomočjo informatizacije

Eden izmed primarnih ciljev informacijskega okolja in tehnologije je podpirati napore pri managementu in pospeševati organizacijsko znanje. Boljši dostop do informacij ne prinaša koristi, kot bi si želeli, dokler se informacije ne uporabijo pri samem doseganju zastavljenih ciljev in uspešnosti organizacije (Dimovski, Penger & Škerlavaj, 2007, str. 86).

Na podlagi analize trenutnega okolja in na podlagi podanih zahtev, ki jih moramo z aplikativno rešitvijo podati, moramo izdelati programsko rešitev, ki bo:

- Podprla poslovni proces izobraževanj zaposlenih,
- omogočala nadaljnji razvoj in povezovanje z rešitvami, ki jih je še potrebno integrirati,
- uporabnikom prijazna za uporabo in v skladu z rešitvami, ki jih že poznajo.

Ker je tehnološka arhitektura sodobna, težav s samo implementacijo nove rešitve ni pričakovati. Tudi možnosti glede izbire tehnologije so odprte, čeprav je glede na široko uporabo Microsoft tehnologije bližja izbira tehnologije podjetja, ki je že vpeto v sistem in je manj smiselna vpeljava nove tehnologije drugega proizvajalca.

3 ANALIZA ZAHTEV

3.1 Zakonska osnova

Kolektivna pogodba za dejavnost vzgoje in izobraževanja v Republiki Sloveniji (Ur.l. RS, št. 52/94, 49/95, 34/96, 45/95 – popr., 51/98, 28/99, 39/99 – ZMPUPR, 39/00, 56/01, 64/01, 78/01 – popr., 56/02, 43/06 – ZkoIP, 60/08, 79/11, 40/12 in 46/13 v nadaljevanju KPDVI) določa, da se z aktom o sistematizaciji delovnih mest določijo posebni delovni pogoji za sklenitev delovnega razmerja. Opredelitev del delavca je opredeljena preko kolektivne pogodbe, kot tudi preko sistematizacije delovnih mest znotraj organizacije, glede na potrebe in zahteve ravnatelja. Poleg obveznosti in dolžnosti, ki so vezane tako na zaposlenega kot tudi ravnatelja ter druge zaposlene v šoli, je v našem primeru izjemno pomemben 53. člen KPDVI, ki se navezuje na samo strokovno izobraževanje in izpopolnjevanje zaposlenih.

Prvi odstavek 53. člena KPDVI pravi: »Delavci imajo pravico do stalnega strokovnega izobraževanja, izpopolnjevanja in usposabljanja. V ta namen mora zavod delavcem na delovnih mestih za katere se zahteva najmanj srednja izobrazba omogočiti letno najmanj 5 dni strokovnega izobraževanja oziroma 15 dni na vsaka 3 leta, ostalim delavcem pa najmanj 2 dni letno oziroma 6 dni na vsaka tri leta.« Zahtevane stopnje izobrazbe učiteljev v srednjem in poklicnem izobraževanju so opredeljene v Pravilniku o izobrazbi učiteljev in drugih strokovnih delavcev v poklicnem in strokovnem izobraževanju (Ur.l. RS št. 48/11 in 92/12, v nadaljevanju PUDSD).

Stroške, ki so vezani na strokovno izpopolnjevanje, izobraževanje in usposabljanje vključno s pripadajočimi nadomestili, mora plačati zavod. V sklopu stroškov pripadajo delavcu tudi povračila, kot so:

- Potni stroški,
- kotizacije in
- stroški bivanja.

Delavec se je dolžan izobraževati, če ga je zavod napotil na izobraževanje, ki je vezano na pridobitev javno veljavne izobrazbe. Program strokovnega izpopolnjevanja in izobraževanja, ki je potrebno znotraj posameznega zavoda, določi ravnatelj po predhodnem mnenju strokovnega organa. Ravnatelj je tudi oseba, ki skrbi za potrjevanje sklepov o posamičnih izpopolnjevanjih in izobraževanjih.

Obvezni del izobraževanj zaposlenih je naveden tudi v Zakonu o organizaciji in financiranju vzgoje in izobraževanja (Ur.l. RS 47/15: ZOFVI-NPB19, v nadaljevanju ZOFVI), ki je zapisan v uradnem listu RS št. 16/07 in opredeljuje temeljne zahteve, ki so osnova za izobraževanje strokovnih delavcev v vzgojno-izobraževalnih ustanovah. Sprva si bomo pogledali, kaj pravi zakonska osnova, preverili del o izobraževanju, ki je zapisan v javno dostopnem dokumentu LDN in hkrati preko Pravilnika o napredovanjih preverili, kakšne so osnove za napredovanje, ki so vezane tudi na osnovne zahteve po izobraževanju strokovnih delavcev.

V 105. členu ZOFVI vidimo, da je za strokovne delavce v vrtcih in šolah podano strokovno izobraževanje in izpopolnjevanje za vse zaposlene. Na podlagi izobraževanj, ki so nujna, kasneje delavci lahko napredujejo v ustrezne nazive in plačilne razrede. Pogoje,

načine in postopke strokovnega izobraževanja in usposabljanja ter napredovanja v nazive, določi minister.

Delo strokovnega delavca je opredeljeno v 119. členu ZOFVI, kjer je jasno zapisano tudi, da je delo vezano na LDN šole, ki je potrjen s strani sveta šola in predložen na Ministrstvo za izobraževanje, znanost in šport.

Strokovni delavci lahko na podlagi PNZVIN napredujejo v nazive, ki se glede na doseganje le-teh razvrščajo v mentorja, svetnika in svetovalca.

V posamezne nazive lahko napredujejo strokovni delavci, ravnatelji oz. direktorji, ki izpolnjujejo z zakonom in drugimi predpisi določene pogoje za strokovne delavce, ravnatelje oz. direktorje. V skladu z zapisanim lahko v posamezen naziv napredujejo strokovni delavci, ravnatelji oz. direktorji, ki opravljajo vzgojno-izobraževalno delo oz. svetovalno delo ali razvojno delo. Pridobljeni nazivi mentor, svetovalac in svetnik so trajni.

Minister za šolstvo je tisti, ki je na podlagi PNZVIN pristojen za preverjanje in potrjevanje napredovanj. Pridobitev naziva strokovnega delavca, ravnatelja iz direktorja je na podlagi PNZVIN mogoč:

- S 1. decembrom, v kolikor je bil predlog za napredovanje vložen do vključno zadnjega dne novembra,
- s 1. marcem, v kolikor je bil predlog za napredovanje vložen do vključno zadnjega dne februarja in
- s 1. julijem, v kolikor je bil predlog za napredovanje vložen do vključno zadnjega junija.

Ravnatelj je tisti, ki predlaga napredovanje strokovnega delavca v naziv, soglasje pa mora podati tudi strokovni delavec sam. Napredovanje ravnatelja ali direktorja predlaga svet šole, vrtca oz. zavoda v soglasju z ravnateljem.

V kolikor strokovni delavec izpolnjuje pogoje za napredovanje v naziv, vendar le-tega ne stori ravnatelj, lahko strokovni delavec sam vloži vso potrebno dokumentacijo skupaj s predlogom za napredovanje.

V primeru, da obrazcu, skupaj z dokazili o ustreznem strokovnem izpopolnjevanju delavca, niso priloženi ustrezni dokumenti, kot so ocena uspešnosti delavca ali mnenja o predlogu za napredovanje, bo ravnatelj zavoda pozvan k dopolnitvi z ustrezno dokumentacijo. V primeru, da se le-ta v zakonsko predpisanem času, ki je podan z zahtevkom s strani ministrstva ne odzove, se pozove svet šole oz. vrtca.

V naziv mentor lahko napreduje strokovni delavec, ravnatelj oz. direktor, ki ima:

- Najmanj 4 leta delovne dobe, ki pa ni nujno vezana zgolj na šolski prostor,
- opravlja svoje delo uspešno,
- je uspešno opravil programe dodatnih izpopolnjevanj in usposabljanj v vzgoji in izobraževanju, ki doprinesejo najmanj 4 točke,
- je opravil dodatna strokovna dela različnih področij, ki prinesejo najmanj 4 točke.

Napredovanje v naziv mentor je mogoče tudi, če strokovni delavec dopolni 25 let delovne dobe v vzgoji in izobraževanju.

V naziv svetovalac lahko napreduje strokovni delavec, ravnatelj oz. direktor, ki:

- Je predhodno imel naziv mentor vsaj 4 leta,

- opravlja svoje delo uspešno,
- je uspešno opravil programe dodatnih izpopolnjevanj in usposabljanj v vzgoji in izobraževanju, ki prinesejo najmanj 5 točk,
- je opravil dodatna strokovna dela različnih področij, ki prinesejo najmanj 18 točk, kjer mora biti najmanj 9 izmed teh 18 točk pridobljenih preko potrdil vrednih vsaj 2 točki ali več.

Ne glede na prej zapisano, je mogoče v naziv svetovalca napredovati tudi pod določenimi pogoji, in sicer če strokovni delavec ima:

- Najmanj 4 leta delovne dobe,
- opravlja svoje delo nadpovprečno uspešno,
- je opravil dodatna strokovna dela različnih področij, ki prinesejo najmanj 18 točk pridobljenih preko potrdil, vrednih vsaj 3 točke ali več.

V naziv svetnika lahko napreduje strokovni delavec, ravnatelj oz. direktor, ki ima:

- Naziv svetovalca najmanj 5 let,
- je uspešen pri svojem delu,
- je uspešno opravil programe dodatnih izpopolnjevanj in usposabljanj v vzgoji in izobraževanju, ki prinesejo najmanj 7 točk,
- je opravil dodatna strokovna dela različnih področij, ki prinesejo najmanj 38 točk, kjer mora biti najmanj 18 izmed teh 38 točk pridobljenih preko potrdil, vrednih vsaj 3 točke ali več.

Ne glede na prej zapisano, je mogoče v naziv svetnika napredovati tudi pod pogoji, in sicer, če ima strokovni delavec:

- Najmanj 8 let delovne dobe,
- opravlja svoje delo nadpovprečno uspešno,
- je opravil dodatna strokovna dela različnih področij, ki prinesejo najmanj 38 točk pridobljenih preko potrdil, vrednih vsaj 3 točke ali več.

V 19. členu pravilnika o napredovanju zaposlenih v vzgojno-izobraževalnih ustanovah je opredeljeno tudi dodatno strokovno izpopolnjevanje in uveljavljanje nadaljnjih izobraževanj, ki so vezana na diplomu, specializacijo, magisterij ali doktorat znanosti, ki ne predstavlja pogoja za zasedbo delovnega mesta in se vrednoti:

- Diploma 4 točke,
- specializacija 6 točk,
- magisterij 8 točk;
- doktorat znanosti 16 točk.

Če se strokovni delavec dodatno izpopolnjuje, vendar usposabljanje ni opredeljeno kot program v skladu s pravilnikom, ki ureja nadaljnje izpopolnjevanje in usposabljanje v vzgoji in izobraževanju, se le-to vrednoti po sledečem točkovanju:

- 8 do 15 ur izobraževanj prinese 0,5 točke,
- 16 do 23 ur izobraževanj prinese 1 točko,
- 24 ur ali več prinese 1,5 točke.

V primeru, da je izobraževanje daljše od 24 ur, se lahko na predlog Programskega sveta za nadaljnje izobraževanje in izpopolnjevanje zaposlenih v vzgoji in izobraževanju na podlagi sklepa ministra ovrednoti z več točkami, vendar je potrebno priložiti:

- Potrdilo o udeležbi,

- predmetnik programa, ki se ga je strokovni delavec udeležil skupaj s pripisom ur izobraževanj, opisom tem, predavatelji in metodami dela,
- dokazilo o aktivni udeležbi.

Pridobivanje točk strokovnega delavca je vezano tudi na dodatna strokovna dela, ki so opredeljena v 20. členu pravilnika PNZVIN.

Strokovna ocena dela strokovnega delavca je izvedena s strani ravnatelja oz. direktorja zavoda.

3.2 Omejitve

Kadar je govora o omejitvah, je tukaj na prvem mestu vprašanje, kakšni predpisi veljajo na področju zbiranja osebnih podatkov in ustvarjanja zbirk osebnih podatkov. V kolikor si na tem mestu ogledamo Zakon o varstvu osebnih podatkov (Ur.l. RS, št. 94/07, v nadaljevanju ZVOP) hitro naletimo na 7. člen, ki govori o izjemah uporabe tega zakona. Vidimo lahko, da 4. točka tega člena govori o številu zaposlenih, kjer je meja za upravljanje z osebnimi podatki postavljena pri številki 50 zaposlenih. V našem primeru govorimo o 53 strokovnih delavcih oz. skupno 60 zaposlenih v trenutku obdelave podatkov.

Na našo srečo je znotraj istega člena 5. točka 7. člena ZVOP namenjena ravno izjemam, kjer je zapisano, da se upravljalcem zbirk osebnih podatkov, ki jih vodijo v javnem sektorju, notarji, odvetniki, detektivi, izvršitelji, izvajalci zasebnega varovanja, zasebni zdravstveni delavci, izvajalci zdravstvenih storitev ter upravljavci osebnih podatkov, ki vodijo zbirke, ki vsebujejo občutljive osebne podatke in je obdelava teh podatkov del registrirane dejavnosti, ni potrebno dodatno opredeljevanje ali določanje varnostnih protokolov, za uporabo in obdelavo podatkov.

Šola kot taka se ukvarja z obdelavo podatkov dijakov in zaposlenih, zato je ena izmed tistih organizacij, ki podatke za osebno obdelavo lahko hrani v zbirkah podatkov, ki so ustrezno zavarovane in niso na voljo nepooblaščenim osebam.

Zavarovanje osebnih podatkov, ki so občutljive narave npr. EMŠO, davčna številka, naslov, podatki o starših itn. morajo biti posebej označeni oz. zavarovani, da nepooblaščenim osebam dostop ni omogočen.

Pri prenosu občutljivih podatkov, kjer je izmenjava podatkov izvedena preko uporabe telekomunikacijskih omrežij, se za varni načini prenosa smatra ustrezno zavarovane podatke, če se uporabijo ustrezne kriptografske metode ali elektronski podpis.

Na drugi strani lahko govorimo o omejitvah tudi, ko je govora o izdelavi prve različice aplikacije. Kot je bilo zapisano, je končna ideja izdelave portala za zaposlene, kjer bi na eni strani vodili izobraževanja zaposlenih, na drugi pa bi šlo za vodenje stanja učilnic in tudi ostalih aktivnosti, kot je portal intranet. Tudi pri samem izobraževanju je začetna ideja podprtje in informatizacija pregleda izobraževanj, kjer se ob dodatnih zahtevah portal lahko nadgradi in dopolni z dodatnimi funkcionalnostmi. Podatkovni model bomo zato poskusili pripraviti v osnovno zahtevani obliki, z možnostjo dodajanja ustreznih tabel ob širitvi.

3.3 Analiza poslovnih procesov

V hitro spreminjajočem se poslovnem svetu oz. v poslovnem okolju podjetja, predstavlja prenova poslovanja in hkrati neprestano prilagajanje programskih rešitev, edino stalnico. Spremembe, ne vplivajo le na samo poslovanje, temveč tudi na informacijsko podporo poslovanju. Učinkovito prilagajanje programskih rešitev zahteva neprestano spreminjanje in prilagajanje poslovnim pravilom. Vsako poslovno pravilo mora biti spremljano na vseh ravneh uporabe skozi celoten razvojni cikel. Zavedati se moramo, da noben pristop ali informacijsko orodje ni v naprej predpisana edina pot ali prava pot. Tudi pretirana analiza obstoječih procesov, neprestano dopolnjevanje in izpopolnjevanje, kot tudi prenašanje celotne odgovornosti na informatiko, pogosto ne vodi do končnega cilja. Kot je ugotovljeno v strokovni literaturi, so uspešni projekti tisti projekti prenove in informatizacije poslovanja, kjer je stalno prisotna vodilna in usmerjevalna vloga managementa. Izhodišče mora biti jasen in nedvoumen poslovni model, ki nam služi kot osnova za končno rešitev. Management mora biti aktivno vključen v proces prenove, saj se le tako lahko premosti ali odpravi tradicionalni prepad med poslovanjem in informatiko (Kovačič & Boksilj Vukšić, 2005).

V poglavju 1.3 smo si že ogledali poslovni proces, ki je vezan na potek izobraževanj zaposlenih. Naš cilj je poenostaviti in pohitriti proces izobraževanj ter ga podpreti s pomočjo izdelave informacijske rešitve.

V poglavju 2.5.3 smo predstavili model, kot si ga želimo imeti. Proces je bil pregledan skupaj z vodstvom šole, da že v samem začetku ne bi naleteli na ovire pri izdelavi rešitve, ki ne bi podpirala končnega cilja ali morda celo popolnoma zgrešenega pristopa pri izdelavi in pripravi končne rešitve.

Naš cilj, ki ga želimo doseči je, da bomo s pomočjo razvoja nove spletne aplikacije podprli proces izobraževanj, le tega naredili bolj preglednega vodstvu kot tudi končnim uporabnikom. Rešitev mora biti enostavna, saj je povprečna starost kadrov 47 let. Same izkušnje pri vpeljavi novih rešitev so znotraj organizacije vedno naletele na odpor, saj predstavljajo novo storitev, ki jo je potrebno osvojiti in se naučiti uporabe, hkrati jo uporabniki lahko razumejo napačno kot dodaten nadzor.

Skozi informatizacijo izobraževanj zaposlenih, ostalih poslovnih procesov ne bomo spreminjali ali vplivali na njih. Ob pregledu ostalih poslovnih procesov, ki potekajo v organizaciji, vidimo da so med seboj do neke mere povezani, a hkrati sprememba izobraževanja zaposlenih ne bo vplivala na temeljne procese. Vpliv izobraževanj je bil, kot bo tudi v prihodnje, na osnovni izobraževalni proces, kjer gre predvsem za odsotnost učitelja v času izobraževanja, kar predstavlja zahtevo po nadomeščanju ur v razredu in ostalih dejavnostih. Na drugi strani bo učitelj z novimi znanji pridobil sodobna znanja in pristope, ki jih bo lahko predajal naprej dijakom. S tem bo izboljšal svoje delo, skupno znanje dijakov in pripomogel k boljšemu pouku.

4 IZDELAVA USTREZNE REŠITVE

Pregled trenutne literature nam ob prebiranju pokaže, da je pogled na različnost pristopov in hkrati izbiro tehnologije za prenovu informacijske arhitekture, zelo raznovrsten in se med avtorji razlikuje. Končna rešitev prenove predstavlja rešitev, ki jo lahko ocenimo na

podlagi ocene kvalitete. Dejavniki, ki so izpostavljeni ob ocenjevanju kakovosti, so na primer: čas

do vzpostavitve delovanja novega sistema, stroški življenjskega cikla našega produkta, enostavnost dostopa, zmožnost dodajanja novih storitev in funkcionalnosti, hitrost delovanja aplikacije, nadzor nad funkcijami, zanesljivost delovanja aplikacije, zanesljivost prikazanih podatkov, enostavnost vzdrževanja, skalabilnost, uporabnost s strani zaposlenih in še bi lahko naštevali (Barry, Conboy, Lang, Wojtkowski & Wojtkowski, 2009, str. 939).

Preden se lotimo izdelave projekta, si moramo zastaviti nekaj vprašanj:

- Kakšen je želeni izhod podatkov?
- Kakšne vrste uporabnikov imamo v podjetju, ki bodo pripravljeno rešitev uporabljali?
- Katere podatke bomo vanj vključili?
- Ali bodo izhodni podatki natisnjeni, prikazani na zaslonu, morda celo oboje?
- Katere naprave bodo uporabljene za dostopanje do podatkov?
- Ali obstaja varnostno tveganje zlorabe podatkov?

Preden se lotimo izdelave, si zastavimo vprašanja in seveda nanj poskusimo odgovoriti. Da bo končni rezultat čim bližje temu, kar je bilo podano kot zahteva vodstva po rešitvi in bo hkrati možno implementirati znotraj obstoječega tehnološkega okolja. Že v naprej vemo, da bodo uporabniki zaradi ločenih omrežij dostopali do portala preko internega omrežja. Varnostno tveganje se tako zniža, saj dostopnosti preko internetne povezave iz zunanjega okolja ne bo. Tudi dostopanje znotraj lokalnega omrežja bo omejeno na kabinete in zbornico. Dostop bo torej omejen z VLAN-i. Uporabniki v teh omrežjih pametnih naprav ne morejo priključiti in dostopati do internega omrežja. Dostop do interneta je mogoč preko Eduroam omrežja, ki pa je ločeno od internega dela.

Uporabniki si želijo preprost vmesnik, kjer ne bo prikazanih preveč podatkov sočasno. V primeru prijave uporabnika bodo le-ti prejeli zgolj svoje podatke in drugih ne bodo videli. Podatki vseh uporabnikov bodo prikazani le administratorjem sistema. Ker večina uporabnikov računalnike uporablja za svoje delo kot delovni pripomoček in niso večji uporabe zahtevnejših aplikacij, je vsekakor potrebno imeti v mislih rešitev, ki bo enostavna in ne bo zahtevala dolgih izobraževanj.

Ker gre za spletno aplikacijo, bodo podatki prikazani na zaslonu računalnika, vsekakor pa velja razmisliti tudi o zmožnosti tiskanja prilagojene oblike, saj bi bilo mogoče natisnjeno obliko vstaviti v osebno mapo zaposlenega.

Varnostno tveganje je vedno prisotno, naša naloga je, da ga minimiziramo in poskrbimo, da bi bila zloraba takoj odkrita. S pomočjo požarnega zidu bomo poskrbeli za ustrezne pravice, ki bodo omejevale dostopnost, aktivni imenik nam bo služil za avtentikacijo uporabnika in hkrati bomo uporabili sistem beleženja dostopov do portala.

4.1 Načrtovanje projekta

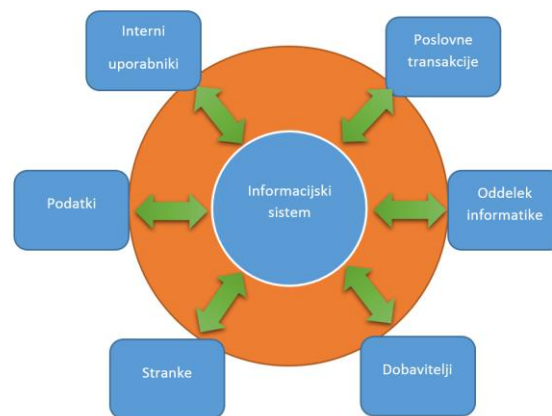
Končna rešitev vključuje tako vizualno podobo, ki jo bo predstavljala rešitev kot tudi ustrezen fizični model, ki bo integriran v našo rešitev. Uporabniški vmesnik oz. User Interface (v nadaljevanju UI) opisuje način komunikacije uporabnika preko računalnika z našim sistemom. Uporabniški vmesnik predstavlja povezavo programske rešitve, podpore

strojne opreme, vhodno izhodne naprave, programske funkcionalnosti, izhodne podatke in ostale možnosti, ki se odvijajo preko dvosmerne komunikacije.

Pri razvoju starejših rešitev so analitiki sprva pripravili vizualno podobo, ki je vsebovala tako tiskanje na papir kot tudi izhoda podatkov programa, da so predstavili rezultate. Uporabniški vmesnik pogosto vsebuje kontrole, ki omogočajo nadziranje procesov preko zaslona naprave, ki izvajajo komunikacijo s sistemom. Takšen pristop je bil učinkovit s starejšimi sistemi, ki so podatke zgolj pretvarjali iz ene oblike v drugo.

S prehodom iz centralizirane obdelave podatkov v dinamično, se je fokus usmeril na končnega uporabnika. Informatiki so postali neke vrste oskrbovalci informacijske tehnologije in ne več skrbniki informacij. Naslednja slika prikazuje sodobni pristop in usmerjenost informacijskega sistema.

Slika 9: Sodobni, uporabniško usmerjeni model



Vir: G. B. Shelly, T. J. Cashman, H. J. Rosenblatt, System Analysis and Design (7th ed.), 2008, str. 298.

Deset najpomembnejših dejavnikov, ki jih moramo upoštevati pri izdelavi grafičnega vmesnika za končnega uporabnika (Shelly, Cashman & Rosenblatt, 2008):

- Perspektiva uporabnika,
- namestitvev,
- skladnost s podanimi funkcijami,
- navodila za uporabnika,
- nadzor nad sistemom,
- povratne informacije,
- odvisnost med programi,
- zmožnosti sistema,
- pomoč uporabniku in
- uporabnost programske rešitve.

Da bi bila rešitev narejena po zahtevah vodstva in uporabnika, moramo natanko vedeti, kaj želimo doseči. Za doseg ciljev moramo poznati poslovne funkcije in vse vpletene v njih, saj bomo le tako ustrezno pridobili vse potrebne informacije, ki jih bomo upoštevali v naši rešitvi. Vizualizacija je pomemben vidik, ki ga uporabniki pričakujejo od programske

rešitve. Če se malce zazremo v zgodovino, bomo hitro ugotovili, da je popularnost sistema Microsoft Windows močno narasla v času, ko je le ta prešel iz diskovnega operacijskega sistema (v nadaljevanju DOS) okolja v grafično okolje. Grafični vmesnik je enostavnejši in se hitreje prikupi uporabniku. Uporabniške izkušnje so pomembne pri vpeljavi nove programske rešitve. Predznanje uporabnikov je lahko ovira pri uvajanju nove rešitve, saj jih lahko pridobivanje novega znanja odvrne od uporabe novosti. Enostavnost rešitve je pomembna, saj bomo le tako uspešni pri uvajanju nove rešitve. Da bi bila rešitev ustrezna za končnega uporabnika, se moramo poskusiti vživeti v njegovo vlogo.


Razmišljamo kot uporabnik, saj nam bo to pomagalo pri razvoju ustrezne rešitve. S pomočjo prototipa si lahko pomagamo pri izdelavi končne rešitve. V vmesni fazi lahko izdelamo model, ki je zgolj vizualno blizu končnemu, vendar skozi analizo in testiranja tega spreminjamo in približamo končnemu uporabniku. Stalna komunikacija z uporabniki nam bo dala potrebne povratne informacije, kje vidijo težave v uporabi, nepreglednost ali slabšo funkcionalnost, kot so jo morda pričakovali.

Pomembno je, da izdelamo uporabniški vmesnik, ki bo atraktiven, enostaven za uporabo in hkrati učinkovit. Kadar izdelujemo nov uporabniški vmesnik, poskušamo upoštevati nekaj osnovnih usmeritev (Shelly, Cashman & Rosenblatt, 2008):

- Bistvo naj bo usmerjeno v osnovne parametre,
- naredimo vmesnik, ki je enostaven za uporabo in učenje,
- delo naj bo učinkovito,
- uporabniku naj bo pomoč na dosegu,
- zmanjšamo možnosti napačnega vnosa podatkov,
- uporabniku omogočimo povratno informacijo,
- ustvarimo atraktiven vmesnik in stil in
- uporabimo dobro znane izraze in slike za ikone.

Osnovni dizajn uporabniškega vmesnika bomo izrisali s pomočjo programa Microsoft Expression Design 4, ki je namenjen izdelavi programskega vmesnika, spletne aplikacije itn. vendar brez dodatne funkcionalnosti.

Slika 10: Grafični vmesnik TO-BE spletne aplikacije

Portal za zaposlene				 Srednja šola tehniških strok Šiška www.ssts.si		
Zasposleni	Učilnice	Obvestila	Šolsko leto 2014/2015			Prijava
ZAP. ŠT.	IME IN PRIIMEK	TIP IZOBRAŽEVANJA	ŠT. UR/DNI	POROČILO	POTRDILO	ŠT. TOČK
1.	Peter Kребelj	strokovno	3 dni	poročilo.doc	ni vnosa	1,5
		strokovno	8 ur	poročilo.doc	ni vnosa	0,5
		strokovno	8 ur	poročilo.doc	ni vnosa	0,5
2.	Janez Novak	strokovno	2 dni	poročilo.doc	ni vnosa	1
		pedagoško	8 ur	poročilo.doc	ni vnosa	0,5
		strokovno	8 ur	poročilo.doc	ni vnosa	0,5
3.	Jožica Novak	strokovno	2 dni	poročilo.doc	ni vnosa	1
		pedagoško	8 ur	poročilo.doc	ni vnosa	0,5
		strokovno	8 ur	poročilo.doc	ni vnosa	0,5
4.	Polde Novak					

Slika nam prikazuje osnovni prikaz ob prijavi uporabnika in je zgolj začetno izhodišče za oblikovanje.

Pogled nad vsemi uporabniki je predviden le za administratorja in ne za vse zaposlene.

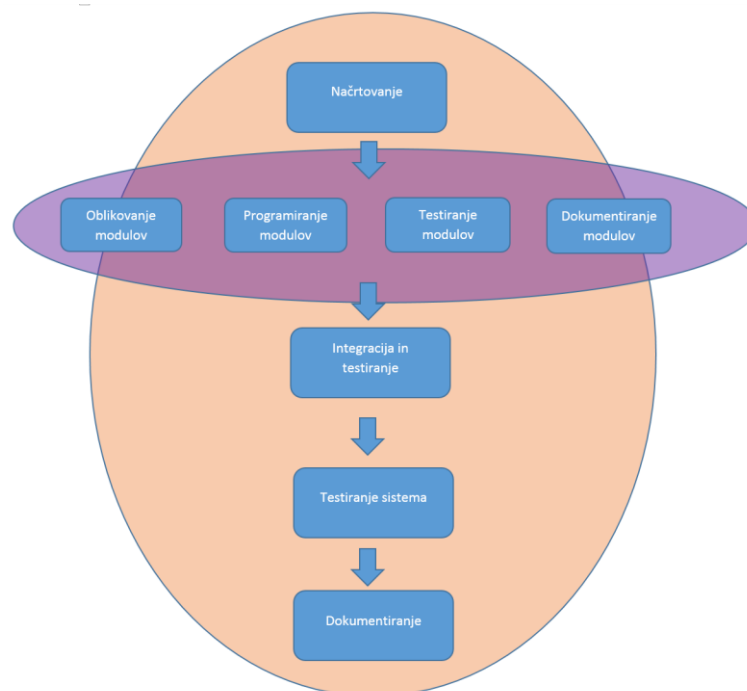
Če se malce vrnemo na točke, ki nam podajajo neke smernice pri razvoju in jih apliciramo na našo rešitev bi lahko rekli sledeče:

1. Bistvo naj bo usmerjeno v osnovne parametre: znotraj izdelave bomo poskusili zajeti le tisto, kar poslovni proces izobraževanja od nas zahteva.
2. Naredimo vmesnik, ki je enostaven za uporabo in učenje: spletni vmesnik bo vseboval minimalistično število gumbov in ostalih elementov, ki zahtevajo klike in pomikanje znotraj strani. S tem bomo zmanjšali verjetnost za napake in težave pri uporabi sistema.
3. Delo naj bo učinkovito: končno oceno učinkovitosti bodo podali končni uporabniki.
4. Uporabniku naj bo pomoč na dosegu: znotraj organizacije so na voljo trije administratorji, kjer dva predstavljata prvo stopnjo pomoči, tretji pa je namenjen reševanju večjih težav. Vseskozi so administratorji sistema na voljo končnim uporabnikom.
5. Zmanjšamo možnosti napačnega vnosa podatkov: zmanjševanje napak bomo omejili že pri vnosnih poljih, ki jih bomo uporabili pri branju in prenašanju podatkov v podatkovno bazo.
6. Uporabniku omogočimo povratno informacijo: v sam vmesnik lahko dodamo hitro povezavo za pošiljanje elektronske pošte, vendar bi to vodilo v pošiljanje zahtevkov po pomoči in ne ravno pošiljanju povratnih informacij.
7. Ustvarimo atraktiven vmesnik in stil: atraktivno ni nujno tudi najbolje, saj se uporabnik v množici interaktivnih elementov lahko preprosto izgubi. Ker so ciljna publika tudi starejši v tem delu ne bomo pretiravali.

8. Uporabimo dobro znane izraze in slike za ikone: uporaba znanih ikon in simbolov bo uporabnikom olajšala delo in poenostavila razumevanje.

Pri razvoju aplikacije naj si gre za strukturirano ali objektno orientirano načrtovanje, bomo pretvorili oblikovanje aplikacije v delujočo aplikacijo. Pri načrtovanju nove aplikacije moramo biti še posebej pazljivi, saj bodo morda kasnejše želje težje implementirane. Ko imamo pripravljeno celotno strategijo izdelave aplikacije, moramo oblikovati vsak posamezen modul, napisati ustrezno kodo zanj, testirati vsak modul in dokumentirati programsko kodo. Programski modul predstavlja povezane dele programa, ki so organizirani v manjše enote, ki jih je lažje razumeti in tudi vzdrževati. Sledi testiranje celotnega sistema, ki ga postavimo v končno okolje.

Slika 11: Osnovni koraki v razvoju aplikacije



Vir: G. B. Shelly, T. J. Cashman, H. J. Rosenblatt, *System Analysis and Design* (7th ed.), 2008, str. 453.

Izdelave projekta in programiranja končne rešitve se bomo lotili na podlagi analize projekta in zakonske osnove, končna rešitev pa bo morala podpirati naš prenovljeni poslovni proces izobraževanj zaposlenih.

4.2 Izbira ustrezne tehnologije

Odločanje predstavlja proces izbiranja med različnimi danimi možnostmi. Na voljo nam je več alternativnih možnosti, ki dosegajo enak končni cilj.

Današnje okoliščine sprejemanja odločitev managerjev zahtevajo poslovne odločitve, kjer se okolje hitro spreminja. Razlogi za spremembe so (Turban, Sharda & Delen, 2011):

- Hiter razvoj računalniške in informacijske tehnologije vodi v večje število alternativnih možnosti, ki so nam na izbiro,
- večja tekmovalnost med proizvajalci kot tudi podjetji pomeni večje stroške v primeru napačne odločitve,
- negotovost prihodnjega poslovanja, ki je posledica globalizacije, potrošništva in vladne intervencije ter
- hitrejša odločitve kot posledica hitrih sprememb.

Da bi se izognili napačni odločitvi, morajo managerji stremeti k novim pristopom, novim tehnikam in tehnologijam pri odločanju med odločitvenim procesom. Sistematičen odločitveni proces predstavlja sosledje aktivnosti, ki ga v grobem delimo v tri faze (Kovačič, Jaklič, Indihar Štemberger & Groznik, 2004):

- Faza priprave,
- faza oblikovanja ter
- faza izbire.

Fazo priprave smo že izvedli, saj smo naredili pregled in analizo problema, ki ga bomo poskušali rešiti. V fazi oblikovanja določimo množico alternativnih rešitev in izberemo enega ali več kriterijev, ki nam bo služil kot osnova za ocenjevanje možne rešitve problema. Na podlagi modela v fazi izbire ocenimo alternativne rešitve in izberemo najboljšo oz. dovolj dobro rešitev. Pogosto iskanje ali implementacija najboljše rešitve lahko vodi v prezahtevno finančno, časovno ali kakšno drugo slepo ulico, zato je včasih dovolj primerna rešitev, četudi to ne pomeni popolne rešitve.

Izbiro ustrezne rešitve lahko podpremo s pomočjo:

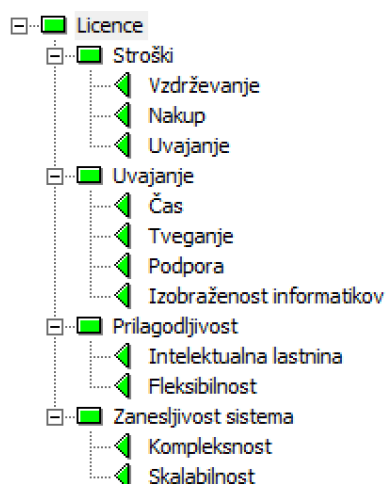
- Modelov za napovedovanje,
- modelov vrst,
- modelov za optimizacijo,
- odločitvenih tabel in dreves,
- vodenja projektov in
- vodenja zalog.

Pri napovedovanju si želimo, da bi glede na znane podatke o pojavu v preteklosti lahko izvedli napovedovanje o pojavu v prihodnosti. Pri samem napovedovanju se lahko opremo tudi na analizo finančnega vidika in izberemo rešitev, ki bo finančno sprejemljiva.

V našem primeru bomo s pomočjo več parametrskega modela podali najpomembnejše kriterije pri izbiri, ovrednotili uteži za posamezna področja in analizirali dobljeno končno rešitev. Sprva se bomo ustavili pri izbiri med licenčno in odprtokodno rešitvijo. Kot smo omenili, temelji naš informacijski sistem v izbrani organizaciji na Microsoft platformi, tako da bomo glede na preference posameznih lastnosti preverili, kakšna bi bila programska rešitev na podlagi modela.

Sprva si pogledimo odločitvene dejavnike, ki bodo vplivali na končno rešitev in programsko izbiro na podlagi vnesenih kriterijev. Uporabljeni program je brezplačno orodje Dexi, kjer sprva opredelimo parametre z ustreznimi atributi, spremenljivkami in kriteriji.

Slika 12: Izbira parametrov za odločanje



Na podlagi modela vnesemo različne variante za odprtokodno in licenčno programsko opremo.

Slika 13: Variante izbire

Varianta	Licenčna	Odpertokodna
Vzdrževanje	Srednje	Srednje
Nakup	Srednja cena	Nizka cena
Uvajanje	Hitro	Dolgo
Čas	Kratek	Dolg
Tveganje	Nizko	Visoko
Podpora	Dobra	Slaba
Izobraženost informatikov	Srednja	Srednja
Intelektualna lastnina	Da	Da
Fleksibilnost	Visoka	Srednja
Kompleksnost	Nizka	Srednja
Skalabilnost	Visoko	Srednje

Ob tem, ko smo vnesli ustrezne pogoje za posamezno vrednost znotraj modela in posamezne variante, izvedemo vrednotenje, ki nam na podlagi vrednosti in variant izbire najprimernejšo rešitev. S pomočjo izdelave poročila dobimo končni pogled, ki ga na podlagi vnesenih vrednosti in izračunanih uteži dobimo kot končni rezultat.

Slika 14: Prikaz povprečnih uteži kriterijev

Kriterij	Lokalne	Globalne	Lok.norm.	Glob.norm.
Licence				
Stroški	22	22	22	22
Vzdrževanje	36	8	36	8
Nakup	36	8	36	8
Uvajanje	27	6	27	6
Uvajanje	26	26	26	26
Čas	46	12	46	12
Tveganje	6	2	6	2
Podpora	44	11	44	11
Izobraženost informatikov	4	1	4	1
Prilagodljivost	33	33	33	33
Intelektualna lastnina	67	22	57	19
Fleksibilnost	33	11	43	14
Zanesljivost sistema	19	19	19	19
Kompleksnost	0	0	0	0
Skalabilnost	100	19	100	19

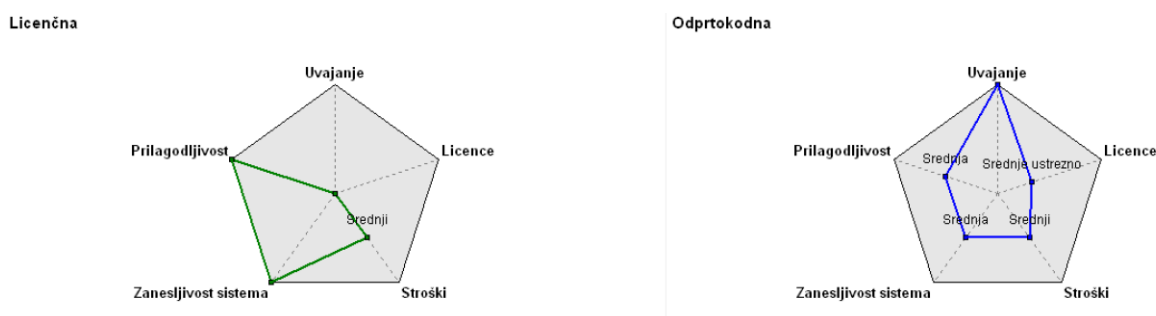
Glede na prikazane vrednosti lahko vidimo, da je v našem primeru glede na podane uteži in končna pravila za licence, najpomembnejši kriterij prilagodljivost, sledijo uvajanja in pa stroški. Zadnji po pomembnosti, glede na ustvarjena pravila, se izkaže kriterij zanesljivost sistema. Vidimo lahko tudi, da so različno pomembni kriteriji znotraj združenih področij, vendar nas glede na končno oceno le-ti ne bodo podrobneje zanimali.

Slika 15: Rezultati vrednotenja z orodjem Dexi

Kriterij	Licenčna	Odpptokodna
Licence	<i>Zelo ustrezno</i>	Srednje ustrezno
Stroški	Srednji	Srednji
Vzdrževanje	Srednje	Srednje
Nakup	Srednja cena	<i>Nizka cena</i>
Uvajanje	<i>Hitro</i>	<i>Dolgo</i>
Uvajanje	<i>Kratko</i>	<i>Dolgo</i>
Čas	<i>Kratek</i>	<i>Dolg</i>
Tveganje	<i>Nizko</i>	<i>Visoko</i>
Podpora	<i>Dobra</i>	<i>Slaba</i>
Izobraženost informatikov	Srednja	Srednja
Prilagodljivost	<i>Visoka</i>	Srednja
Intelektualna lastnina	<i>Da</i>	<i>Da</i>
Fleksibilnost	<i>Visoka</i>	Srednja
Zanesljivost sistema	<i>Visoka</i>	Srednja
Kompleksnost	<i>Nizka</i>	Srednja
Skalabilnost	<i>Visoko</i>	Srednje

Ob pregledu vrednotenja vidimo, da bo naša odločitev temeljila na licenčni programski opremi. Izdelamo lahko tudi oba grafikona, ki nam bosta glede na izbrane kriterije izdelala končni model in prikazala izbiro na podlagi vnesenih vrednosti. Predlagana izbira s strani orodja Dexi je označena z zeleno črto, ki nam podaja tudi našo končno izbiro ob upoštevanju vseh kriterijev.

Slika 16: Prikaz grafa, kot rezultat vrednotenja



Na voljo nam je tudi izdelava poročila, ki nam izdelava drevo kriterijev, prikaže vse zaloge vrednosti, prikaže tabele odločitvenih pravil, kjer imamo podane tako odločitvena pravila kot tudi uteži, ki smo jih uporabili pri odločitvenih pravilih.

Odločitev, vezana na programsko rešitev, je torej znana in se bomo odločili za licenčno programsko opremo. Še vedno ostaja odprto vprašanje ali gremo v lastno izdelavo (make it) ali v nakup neke rešitve (buy it). Lastna rešitev predstavlja vprašanje znanja kadrov, zmožnosti ustreznega razvoja, potrebnega časa, potrebnih financ itn. Na drugi strani imamo možnost nakupa programske opreme, ki morda ne pokriva rešitev v celoti (npr. Microsoft Sharepoint), kjer so mogoče tudi prilagoditve do neke mere ali izdelava rešitve po naših željah in potrebah preko zunanega ponudnika.

Četudi razmišljamo o najemu zunanega podjetja, se moramo zavedati, da bo nekdo od zaposlenih pri komunikaciji moral sodelovati, podajati pojasnila, testirati delovanje, namestiti in zagotoviti delovanje znotraj naše infrastrukture itn. V vsakem primeru najem podjetja za izdelavo ne vodi v nekaj tedensko ali mesečno delo, kjer nam ne bo potrebno sodelovati.

Odločitev lahko sprejmemo preko upoštevanja vhodnih elementov in stroškov, ki so z njimi povezani ter tako izvedemo analizo modela stroškov, ki bi jih imeli pri lastni izdelavi ali izdelavi preko zunanega podjetja.

Kenneth (2012) je v svojem dokumentu zapisal, da je potrebno ob odločitvi ali gremo v lasten razvoj ali gremo v nakup presoditi sledeče temeljne dejavnike:

- Temeljni procesi: Predstavljajo temeljne procese, ki vplivajo na strategijo organizacije.
- Pokritost storitve: Zanima nas kakovost pokrivanja in želja pri vpeljavi nove rešitve.
- Življenjski cikel: Kakšna bo fleksibilnost, vzdrževanje in razširljivost čez celoten življenjski cikel.
- TCO: Vsi stroški povezani z namestitvijo, uporabo in vzdrževanjem aplikacije.
- Integracija: Boljša kot je integracija, večji bo vpliv na jedrne poslovne procese.
- Čas: Koliko časa bo potrebnega za razvoj končne rešitve.
- Standardi: Upoštevanje standardov kot tudi zakonov pri izdelavi aplikacije, še posebej v kolikor gre za storitev, ki se bo širila med organizacijami ali drugimi trgi.
- Nadzor nad storitvijo: Ali imamo možnost popravkov in dodajanja novih funkcionalnosti znotraj pripravljene rešitve.
- Nadgradnje oz. posodobitve: Kakšna so časovna obdobja posodabljanja in odpravljanja napak.

Ponovno se poslužimo orodja Dexi, kjer bomo vnesli vse opisane kriterije in podali vrednosti, ki bodo predstavljale naš odločitveni model.

Slika 17: Model naredi ali kupi



Sprva smo ustvarili odločitve s parametri, dodamo še vrednosti, ki ustrezajo našim zahtevam.

Slika 18: Izbira naredi ali kupi z vrednostmi

Varianta	lasten razvoj	zunanji razvoj
podpora temeljnim procesom	Da	Da
pokritost storitve	v celoti pokrita	deloma pokrita
integracija storitve	v celoti pokrita	deloma pokrita
čas	srednji	srednji
življenski cikel	srednji	srednji
TCO	srednji stroški	srednji stroški
standardi	deloma pokrita	v celoti pokrita
nadgradnje	v celoti pokrita	deloma pokrita
nadzor nad storitvijo	dober nadzor	slab nadzor

S pomočjo poročila preverimo končno rešitev, ki nastane na podlagi modela. Preverimo model uteženih vrednosti, kjer bomo videli, kaj je tisto, kar je pri izbiri najpomembnejše glede na prikazane uteži.

Slika 19: Povprečne uteži za naredi ali kupi

Kriterij	Lokalne	Globalne	Lok.norm.	Glob.norm.
naredi ali kupi				
Storitev	28	28	34	34
podpora temeljnim procesom	63	17	45	15
pokritost storitve	37	10	55	18
Uvajanje rešitve	10	10	9	9
integracija storitve	80	8	84	8
čas	20	2	16	1
Lastništvo	20	20	18	18
življenski cikel	86	17	82	15
TCO	14	3	18	3
Nadzor nad storitvijo	43	43	39	39
standardi	18	8	22	8
nadgradnje	20	8	23	9
nadzor nad storitvijo	62	26	55	21

V končnem poročilu dobimo prikazan tudi rezultat vrednotenja, kjer vidimo, da sta odločitvi precej podobni, a vseeno vidimo, da prevladuje izbira lastnega razvoja.

Slika 20: Prikaz rezultatov vrednotenja za naredi ali kupi

Kriterij	lasten razvoj	zunanji razvoj
naredi ali kupi	deloma pokrita	slabo pokrita
Storitev	v celoti pokrita	deloma pokrita
podpora temeljnim procesom	Da	Da
pokritost storitve	v celoti pokrita	deloma pokrita
Uvajanje rešitve	kratek	srednji
integracija storitve	v celoti pokrita	deloma pokrita
čas	srednji	srednji
Lastništvo	srednji	srednji
življenski cikel	srednji	srednji
TCO	srednji stroški	srednji stroški
Nadzor nad storitvijo	dober nadzor	srednji nadzor
standardi	deloma pokrita	v celoti pokrita
nadgradnje	v celoti pokrita	deloma pokrita
nadzor nad storitvijo	dober nadzor	slab nadzor

Oglejmo si še končna grafa, ki nastaneta kot rezultat analize in hkrati izbire boljše možnosti, kjer sicer ena ne prevladuje tako očitno kot druga, a je vseeno razvidno, da je lastna izdelava boljša izbira.

Slika 21: Lasten razvoj ali zunanji razvoj odločitvenega grafa

lasten razvoj



zunanji razvoj



Postopek določanja ustrezne rešitve bi lahko izdelali tudi preko modela uteženih vrednosti, kjer bi posamezni lastnosti pripisali utež in pripadajočo vrednost. Na tak način bi dobili številčno ovrednoteno lastno izdelavo in možnost nakupa, ki bi nam pomagala pri končni izbiri na podlagi končne ocene, ki bi jo izračunali.

Končno rešitev bomo, glede na analizo okolja in prikaz z orodjem za izbiranje, izvedli s pomočjo sledečih rešitev:

- Postavitev virtualnega strežnika Windows Server 2012 R2 na platformi VMware ESXi z imenom SQL,
- namestitev programskega okolja Microsoft SQL Server 2014 na strežnik SQL,
- priprava podatkovnega modela s pomočjo orodja SAP Power Designer,
- konfiguracija obstoječega aktivnega imenika na primarnem domenskem strežniku,
- konfiguracija požarnega zidu za zagotavljanje ustrezne varnosti in zmožnost dostopa do novega sistema,
- postavitev podpore spletni strani (v nadaljevanju IIS) na Microsoft strežniku na enem izmed obstoječih fizičnih strežnikov in
- izdelava spletne aplikacije s pomočjo C# .NET tehnologije v orodju Visual Studio 2015.

Začetni razvoj bo potekal v ekvivalenten okolju na strežniku doma, kjer bodo uporabljeni enaki virtualizirani sistemi z enako programsko rešitvijo, saj bo delo tako potekalo vseskozi in bo mogoče tudi testiranje izven končnega produkcijskega okolja.

4.3 Izdelava modela

Podatkovni model predstavlja glavni del aplikacije. Določa nam vpletene akcije in pravila, ki ponazarjajo potek poslovnih procesov v obliki, ki je razumljiva tako ljudem kot tudi aplikacijam. Brez podatkovnih modelov ne bi bili zmožni avtomatizirati procesov, ki predstavljajo bistvene procese znotraj organizacije. Slabo zgrajeni podatkovni modeli vodijo v težave, četudi so aplikacije zgrajene na njih izvrstne. Slaba odzivnost, napačni rezultati poizvedb, neprilagodljiva pravila delovanja in nekonsistenčnost podatkov, je le nekaj tistih najbolj tipičnih. Konceptualni in logični model predstavljata način razmišljanja informatikov in hkrati predstavljajo razumevanje in dojetje poteka poslovnega procesa.

Podatkovni model predstavlja nabor simbolov in besedila, ki skupaj natančno opisujeta podmnožico informacij v različnih oblikah. Simboli in besedilo morajo biti podani v ustrezni obliki, ki jo ponazarja poenoten jezik modeliranja (v nadaljevanju UML). Podatkovni model predstavlja bistven del v načrtovanju končne aplikacije. Komunikacija in natančnost sta ključna znotraj samega modela. Tradicionalno so bili podatkovni modeli zgrajeni skozi analizo in razvojne stopnje projekta, da je bilo zagotovljeno pravilno razumevanje zahtev znotraj programske rešitve in preden je bila ustvarjena končna podatkovna baza. Za uporabo je potrebno razumeti (Hoberman, Burbank & Bradley, 2009, str. 13 - 53):

- Celotno sliko obstoječe aplikacije: podatkovni model predstavlja preprosto, a hkrati natančno sliko konceptov obstoječe aplikacije.
- Izvesti analizo vpliva razvoja: podatkovni model lahko vključuje koncepte in interakcije, ki bodo vplivali na nadaljnji razvoj nove aplikacije.
- Poslovno okolje: predhodno je potrebno poznati in razumeti kako poslovni proces deluje, preden se lahko lotimo izdelave aplikacije, ki bo tak poslovni proces podpirala.

- Izobraževanje članov tima: v primeru vstopa novega člana v tim, zaradi potrebe po pospešitvi dela ali nadomestitev drugega člana, podatkovni model pomaga pri učinkoviti predstavitvi koncepta in dela.

Oblikovanje modela podatkov predstavlja modeliranje podatkov. Model podatkov predstavimo abstraktno, kar pomeni da gre za poenostavljen pogled, a vendar razumljivo predstavitev podatkov, ki predstavljajo delo z vidika potrebnih podatkov. Zaporedje postopkov, s katerimi opredelimo entitete, predstavlja modeliranje. Posamezni entiteti dodelimo ustrezne attribute in jih med seboj povežemo. Kadar govorimo o entitetah, najpogosteje govorimo o entitetno relacijskem modelu oz. E-R modelu. Namen modela je osnova za izgradnjo podatkovne baze. Zagotoviti mora dovolj podrobne informacije in dogajanja v podjetju. Znotraj procesa modeliranja podatkov identificiramo povezave med podatkovnimi elementi. Modeliranje nam poda končno sliko podatkovnih elementov, ki jih bomo kasneje upoštevali v postopku zasnove. Podatkovni modeli so uporabljeni za predstavitev konceptualnega in zunanjšega nivoja podatkovne baze. Podatkovni modeli nam ne predstavljajo fizičnega nivoja. V modelu entitet in povezav (v nadaljevanju E-R model) uporabljamo termine, kot so (Hoberman, 2009):

- Entiteta: Entiteta predstavlja objekt opazovanja. Identificiramo jih tako, da jim določimo njihovo ime, opišemo lastnosti in vrednosti. Primer entitete v našem primeru bi bil zaposleni.
- Atribut: Atributi predstavljajo lastnosti oz. attribute znotraj entitete. Atribut predstavlja podatek, ki ga uporabnik ne more več členiti naprej na manjše enote oz. bi bila členitev nesmiselna. Atributi na primeru zaposlenega so: ime, priimek, EMŠO, davčna itn.
- Relacija: Relacija je predstavljena s tabelo, kjer imamo tudi attribute, ki so predstavljeni s stolpci. Relacijska podatkovna baza predstavlja eno ali več tabel. Tabela vsebuje zapise za 0, 1 ali n-vrstic oz. n-teric relacije.
- Ključ: Ključ predstavlja podatkovni element, katerega namen je ločiti vrednost pojava enega elementa od vrednosti pojava drugega elementa. Ker je vrednosti potrebno ločevati, so le-te enolične, kar pomeni, da podvajanj ne dovolimo. Poznamo primarne, sekundarne, kandidatne, zunanje in sestavljene ključe.

Postopek priprave podatkovne baze si bomo ogledali na spodnji sliki.

Slika 22: Postopek priprave podatkovne baze



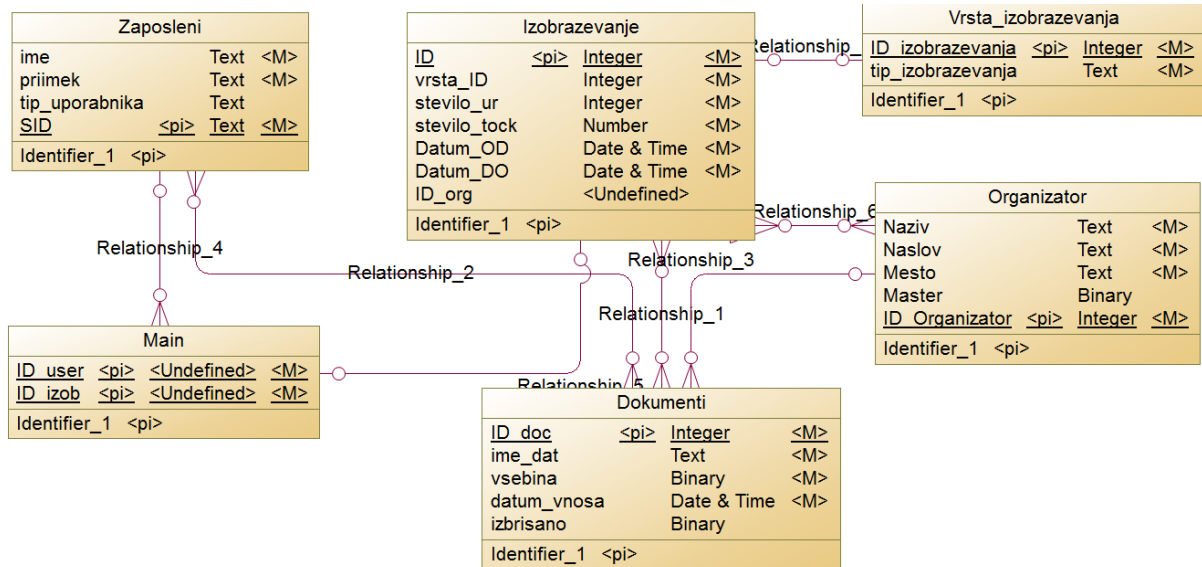
Vir: S. Hoberman, 2009, Data modeling made simple, str. 44.

Na osnovi idejne zasnove bomo pripravili končni E-R diagram. Sledila bo izdelava relacijskega modela in na osnovi te bomo pripravili podatkovno bazo, na kateri bo slonela naša končna rešitev. E-R model predstavlja visoko nivojski abstraktni model za razliko od abstraktnega, ki predstavlja konkreten model podatkovne baze.

V pomoč pri izdelavi modela nam bo program SAP Power Designer, ki nam omogoča izris E-R modela, ki ga lahko pretvorimo v ostale modele, kot so npr. logični model,

konceptualni model, fizični model itn. Na voljo nam je tudi orodje, ki preveri naš model in nas opozori na napake ali možne nepredvidene težave na podlagi načrtanega modela.

Slika 23: E-R model v programu SAP Power Designer

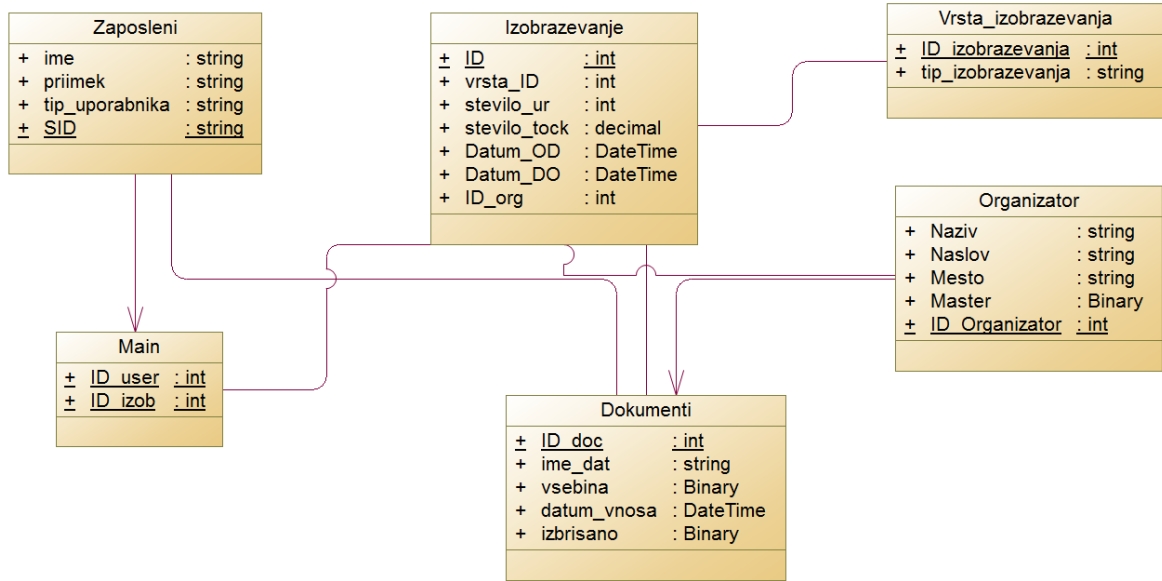


Na zgornji sliki je prikazan E-R model, ki nam bo služil kot osnova za izdelavo preostalih modelov in na samem koncu izdelavo podatkovne baze. V osnovi imamo pet entitet, šesta predstavlja povezovalno entiteto. Osnovna entiteta je zaposleni, ki ima osnovne atribute, kot so ime, priimek, tip uporabnika in povezovalni ID (v nadaljevanju SID), kar predstavlja primarni ključ, na podlagi katerega bomo spremljali posameznega zaposlenega. SID je sočasno namenjen tudi povezovanju z aktivnim imenikom, kjer bomo preverjali prijavne podatke za dostop do aplikacije.

Entiteta izobraževanje predstavlja podatke o konkretnem izobraževanju, kjer bomo preko enoličnega podatka ID vodili vsa izobraževanja, ostali atributi, kot so vrsta_ID, stevilo_ur, stevilo_tock, Datum_OD, datum_Do in ID_org pa bodo služili shranjevanju dodatnih lastnosti o izobraževanju. Entiteta je povezana tudi z entiteto Dokumenti, kjer bomo shranjevali dokumente in na drugi strani entiteta Organizator, kjer bomo vodili podatke o izvajalcu izobraževanja. Vrsta izobraževanja je namenjena vodenju podatka o tem ali gre za strokovno ali pedagoško izobraževanje. Četudi vemo, da bosta tudi v prihodnje le ti dve možnosti smo entiteto ločili, kar nam omogoči dodajanje novih vrednosti, če bi se potrebe po tem izkazale.

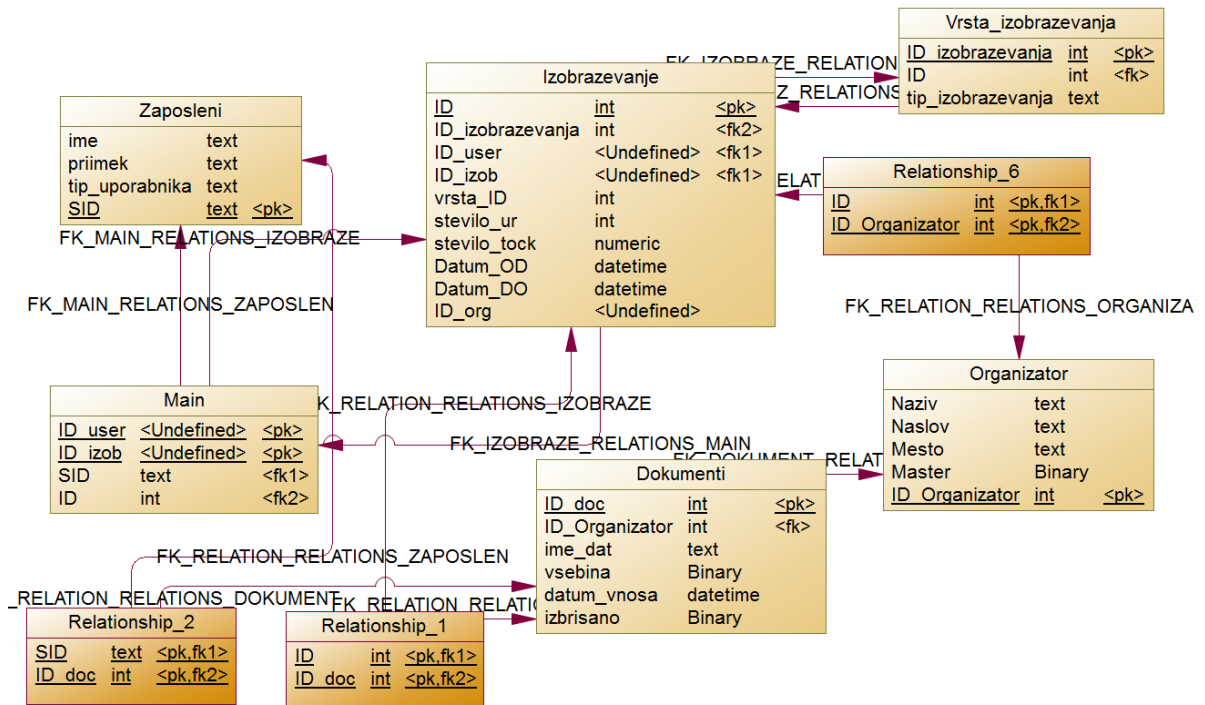
Na podlagi izdelanega modela lahko pripravimo tudi objektno usmerjen model, kjer sem za prikaz izbral jezik C# v okolju .NET.

Slika 24: Objektno usmerjen model C#



Za samo razumevanje in lažjo pripravo podatkovne baze lahko pripravimo tudi konceptualni model za podatkovno bazo.

Slika 25: Konceptualni model



Prikazani model nam bo služil kot osnova za izgradnjo, kar pa ne pomeni, da ničesar ne bi smeli spremeniti ob sami izgradnji in ugotavljanju pravilnosti ter uporabnosti modela.

4.4 Izdelava rešitve

4.4.1 Konfiguracija strežniškega okolja

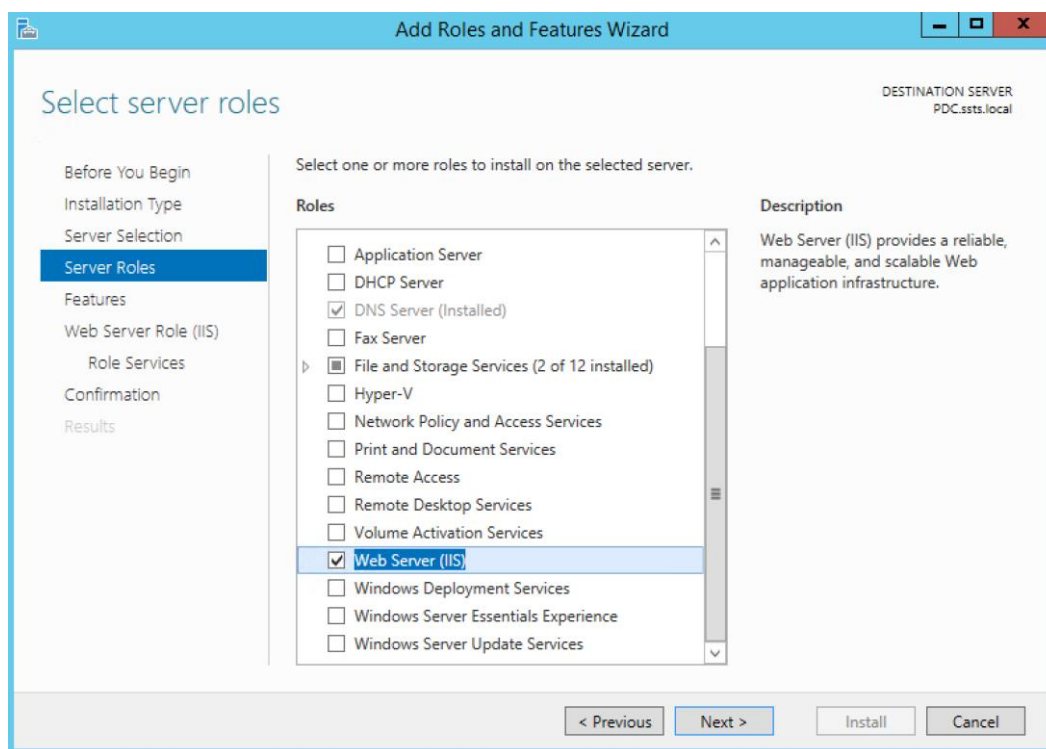
Kot smo že omenili, bomo sprva pripravili ustrezno okolje, kjer bomo poganjali našo končno rešitev. Glede na posamezne sklope konfiguracije, ki jo moramo opraviti pred izvedbo implementacije končne rešitve, si bomo pogledali po korakih glede na vrsto spremembe, kaj in kako bomo izvedli.

4.4.1.1 Aktivacija podpori IIS

Spletna stran za svoje delovanje potrebuje ustrezen spletni strežnik. V ta namen lahko uporabimo brezplačne rešitve, ki so nam na voljo z namestitvijo paketa, kot je npr. Xampp, Wamp in podobne, ki lahko poganjajo Apache ali druge storitve v sklopu podpore spletni strani. V Microsoft okolju imamo na voljo spletni strežnik IIS, ki ga lahko namestimo kot komponento. Na samem strežniku moramo namestiti vtičnik za podporo .NET 3.5 ter IIS.

Obe možnosti bomo našli znotraj upravljanja strežnika, kjer moramo izvesti namestitev obeh ključnih komponent.

Slika 26: Namestitev spletnega strežnika IIS



Pod vlogami strežnika izberemo IIS, ki nam bo omogočal poganjanje spletnih strani v okolju .NET. V naslednjem koraku pod dodatnimi parametri obkljukamo še možnost .NET Framework 3.5.

Microsoftov spletni strežnik IIS predstavlja paleto internetnih procesov, ki jih ponuja preko dodatnih funkcionalnosti. Ker je namestitev enostavna in hitra, je eden izmed popularnejših na svetu. IIS trenutno podpira protokole FTP, SMTP, NNTP in HTTP/HTTPS. Trenutno je v uporabi različica IIS 8. Prednost nove arhitekture je v tem, da se izvajajo zgolj tisti procesi, ki jih zahtevamo. Na voljo so nam tudi možnosti za razširjanje funkcionalnosti in nameščanje dodatnih modulov. Najpogostejši moduli, ki so nam na voljo so:

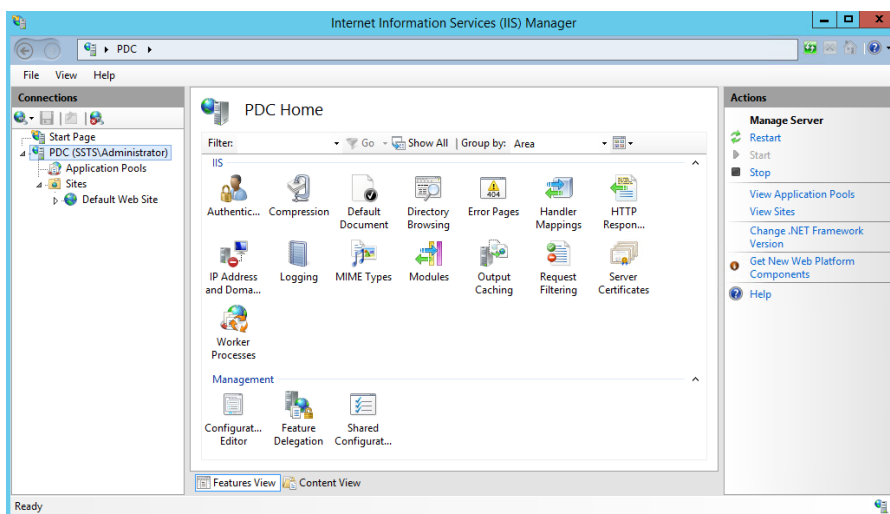
- Moduli za spletne strani (v nadaljevanju http),
- varnostni moduli,
- vsebinski moduli,
- kompresijski moduli,
- pomnilniški moduli in
- dnevniški moduli.

Razširjanje v IIS 8 je mogoče preko programskih vmesnikov. Razvijalci imajo večji nadzor nad zahtevanimi procesi ter možnost, da razvijejo svoje module s pomočjo .NET ogrodja. Vse nastavitve strežnika so shranjene v obliki strukturiranih dokumentov (v nadaljevanju XML), namesto do zadnje različice uporabljene zbirke podatkov. Vsaka navidezna spletna mapa vsebuje svojo datoteko web.config, ki privzame začetne nastavitve. Konfiguracija je vezana na ustrezne pravice, ki so ločene od administratorskih, kar je prednost za manj izkušene administratorje. Potrebna je tudi namestitev .NET podpore (Priloga 2, Slika 1), ki se nahaja znotraj dodatnih možnosti strežnika.

Če smo pri namestitvi izbrali tudi orodja za IIS, nam vmesnik samodejno predlaga tudi namestitev .NET Framework 4.5, ki predstavlja zadnjo različico .NET okolja.

Ob zaključku namestitve lahko preverimo nadzorno ploščo spletne storitve ter funkcionalnosti, ki so nam na voljo. Že ob začetni namestitvi vloge strežnika smo izbrali namestitev vtičnikov za zagotavljanje varnosti, ustreznih orodij za nadzor ter beleženje prometa, kar nam bo prišlo prav pri nadzoru in vpogledu v dogajanje znotraj dostopanja do spletne strani.

Slika 27: Upravljanje spletnega strežnika



Na koncu izvedemo še preverjanje ali storitev deluje do tega trenutka pravilno, in sicer preko brskalnika, kjer lahko uporabimo kar privzeti Internet Explorer z odprtjem lokalnega gostitelja (Priloga 2, Slika 2), kjer se nam odpre začetna stran IIS.

Vidimo, da nam storitev deluje pravilno. Kasneje bomo uredili tudi nastavitve v požarnem zidu, da bomo lahko dostopali do spletne strani iz drugih sistemov znotraj omrežja ter konfiguracijo požarnega zidu, kjer bomo odprli vrata 80 za dostop iz različnih VLAN okolij.

4.4.1.2 Postavitev strežnika SQL

Sprva bomo postavili virtualizirani strežnik z imenom SQL, kjer bomo poganjali Microsoft Server 2012 R2. Za delovanje strežnika je dobro upoštevati priporočila, ki so vezana na zahteve po strojni konfiguraciji:

- 64 bitno arhitekturo strojne opreme,
- 64 bitni operacijski sistem Windows Server 2008 R2 ali novejši,
- 4 GB delovnega pomnilnika,
- 2 GHz procesor ali hitrejši,
- dovolj prostora na trdem disku za namestitev, odvisno od izbire namestitvenih paketov,
- Microsoft .NET Framework 3.5 in
- priporočljivo je imeti posodobljen operacijski sistem.

V našem primeru smo izbrali namestitev virtualnega sistema, kjer smo dodeli:

- 4 GB začetnega dinamičnega pomnilnika,
- deljenje 1 Xeon procesorja,
- 100 GB disk za namestitev z dodatnim diskom za transakcijske datoteke ter shranjevanje podatkov,
- omrežni vmesnik z dodelitvijo ustreznega VLAN segmenta in
- operacijski sistem Windows Server 2012 R2, ki je privzeto 64 biten z dodatno funkcionalnostjo .NET Framework 3.5.

V primeru, da bi bile začetne nastavitve premalo, ni nobenih skrbi, saj lahko virtualnemu računalniku dodelimo dodatne zmogljivosti glede na potrebe. Namestitev je izvedena na diske hitrosti 15.000 vrtljajev na minuto, saj je zaradi večjega števila branj in pisanj v podatkovno bazo potrebna višja hitrost, četudi število uporabnikov še vedno ni tako veliko in ni potrebe po ekstremni zmogljivosti.

4.4.1.3 Konfiguracija požarnega zidu

Požarni zid je, kot je bilo že omenjeno, fizična rešitev s pred nameščenim operacijskim sistemom, temelječim na Linux platformi z integrirano rešitvijo in funkcionalnostjo požarnega zidu.

Konfiguracijo izvedemo preko povezave na spletni vmesnik, kjer uporabimo varno spletno povezavo (v nadaljevanju HTTPS). Naslov IP je dodeljen iz strežniškega podomrežja, torej je v uporabi naslov IP 192.168.0.x. Zadnje vrednosti zaradi varnosti ne bomo podali.

Uporabniki bodo dostopali do spletne strani, ki se bo tudi nahajala v strežniškem omrežju, torej moramo odpreti dostop do naše spletne strani iz VLAN omrežij, kjer bodo uporabniki imeli dostop do nove spletne aplikacije. Glede na začetno analize tehnološke infrastrukture, bomo to izvedli za uporabnike VLAN omrežij učitelji, kabineti in uprava.

Pri dodajanju pravil si lahko pomagamo s prikazano matriko, ki nam olajša razumevanje in konfiguracijo, saj vedno preverimo iz katerega omrežja v katero drugo omrežje želimo spuščati omrežni promet. V primeru, kjer bomo želeli prepuščati omrežni promet za vrata 80 do spletnega strežnika na nekem naslovu IP, bomo izbrali npr. Upravo in nato LAN. Predstavljamo si lahko kot preusmeritev Uprava → LAN. Kratica LAN v našem primeru predstavlja celotno lokalno omrežje, četudi je v resnici celotna ločena na manjša podomrežja z različnimi naslovi IP.

Ob izbiri ustrezne translacije, ki jo želimo izvesti, se nam odprejo vsa pravila znotraj nekega omrežja.

Slika 28: Urejanje pravil v požarnem zidu

Firewall /

Access Rules

Restore Defaults...

Access Rules (UPRAVA > LAN)

View Style: All Rules Matrix Drop-down Boxes View IP Version:

Add... Delete

#	From	>	To	Priority	Source	Destination
<input type="checkbox"/> 1	UPRAVA	>	LAN	1	Any	manager
<input type="checkbox"/> 2	UPRAVA	>	LAN	2	Any	Licensing
<input type="checkbox"/> 3	UPRAVA	>	LAN	3	Any	PDC Private
<input type="checkbox"/> 4	UPRAVA	>	LAN	4	Any	IP kamere
<input type="checkbox"/> 5	UPRAVA	>	LAN	5	Any	PDC Private
<input type="checkbox"/> 6	UPRAVA	>	LAN	6	Any	Licensing
<input type="checkbox"/> 7	UPRAVA	>	LAN	7	Any	pdv Private
<input type="checkbox"/> 8	UPRAVA	>	LAN	8	Any	Licensing
<input type="checkbox"/> 9	UPRAVA	>	LAN	9	Any	pdv Private
<input type="checkbox"/> 10	UPRAVA	>	LAN	10	Any	Licensing
<input type="checkbox"/> 11	UPRAVA	>	LAN	11	Any	pdv Private
<input type="checkbox"/> 12	UPRAVA	>	LAN	12	Any	X0:V61 IP
<input type="checkbox"/> 13	UPRAVA	>	LAN	13	Any	manager
<input type="checkbox"/> 14	UPRAVA	>	LAN	14	Any	X0:V60 Subnet
<input type="checkbox"/> 15	UPRAVA	>	LAN	15	Any	X0:V61 Subnet
<input type="checkbox"/> 16	UPRAVA	>	LAN	16	Any	IP_KAMERE Subnets
<input type="checkbox"/> 17	UPRAVA	>	LAN	17	Any	Any

Add... Delete

Na zgornji sliki je zajet le prvi del zaslona, saj so na drugem vidna natančna pravila z vrati in končno destinacijo, kar bi lahko predstavljalo preveč podrobne podatke in potencialno nevarnost znotraj omrežja.

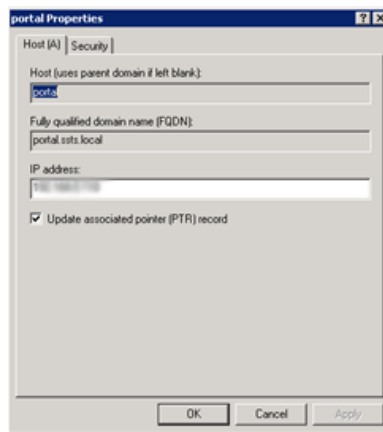
S klikom na gumb Add dodamo novo pravilo, kjer bomo omogočili dostop iz omrežja Uprava do destinacije, ki predstavlja naš IIS strežnik na katerem se nahaja naša spletna stran.

Zaradi večje varnosti znotraj pravila omogočimo tudi beleženje vseh dogodkov, saj bomo le tako našli tudi težave ali poskuse zlorabe, v kolikor bi do le-teh prišlo.

Poleg konfiguracije požarnega zidu v omrežju je potrebno nastaviti tudi požarni zid sistema Windows Server 2012 R2. Da bi omogočili dostop do vrat 80, kjer se bo nahajala naša spletna aplikacija, pod napredne možnosti vnesemo lastno pravilo dostopa, kjer bo postavljen IIS strežnik za vrata 80.

S tem konfiguracija okolja še ni zaključena, saj je potrebno dodati v interni DNS strežnik tudi informacijo o tem, kje se spletna stran nahaja.

Slika 29: Vnos spletne strani v korensko območje DNS



To storimo z dodajanjem novega zapisa znotraj korenskega območja DNS, in sicer ustvarimo nov zapis tipa A, ki predstavlja preslikavo imena IPv4 oz. AAAA, ki predstavlja preslikavo imena IPv6. V našem primeru vnesemo ime gostitelja, ki bo predstavljal naslov za povezavo v krajevnem omrežju in naslov IP, kjer se naša spletna stran nahaja. Zaradi zagotavljanja varnosti je naslov strežnika prikrit oz. zamegljen.

4.4.2 Izdelava programske rešitve

Analizo in modele imamo do tega trenutka pripravljene, tako da se lahko lotimo izdelave končne rešitve. Pričeli bomo z izdelavo modela podatkovne baze, ki bo slonela na pripravljene konceptualni shemi. Da pa bi lahko model ustvarili, bo potrebno namestiti strežnik za delo s podatkovnimi bazami (v nadaljevanju SQL), saj le-tega v organizaciji še nimamo postavljenega.

4.4.2.1 Podatkovna baza SQL

Začetna namestitev bo slonela na postavitvi samostojnega strežnika SQL brez redundance, katero bi sicer lahko ustvarili s postavitvijo dveh strežnikov v gruči. Ker je sistem redundanten že sam po sebi, potrebe po dvojni namestitvi ni. Varnostno kopijo bomo avtomatizirali, s čimer bomo zagotovili shranjevanje podatkov na NAS napravo.

Podatkovni model smo izdelali in nam predstavlja abstrakten pogled na naš problem, logične definicije podatkovnih struktur, podatkovnih operatorjev itn. Model bomo prenesli preko implementacije v končno rešitev, ki bo predstavljala fizično rešitev na našem sistemu (Date, 2011).

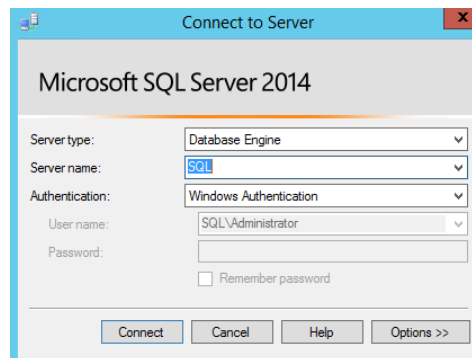
Izvedba postavitve se prične s pripravo strežnika Windows Server 2012 R2 na katerega bomo namestili SQL Server 2014. Ker gre za namestitev na virtualizirani sistem moramo biti pozorni na to, da namestimo tudi paket gonilnikov in ostalih orodij okolja VMware, ki so potrebna za pravilno delovanje virtualnega sistema.

Zaženemo namestitveni paket Microsoft SQL 2014, ki ga prejmemo na namestitvenem mediju ali v našem primeru v obliki ISO slike, ki jo prenesemo preko uradne strani Microsoft Volume Licensing Service Center.

Kot smo omenili v začetku tega poglavja, bomo izvedli namestitev samostojne postavitve (Priloga 3, Slika 1) strežnika SQL.

Sledi nekaj osnovnih nastavitvev kot so vnos licenčne kode, nastavitve globalnih pravil, posodobitve, izbira namestitvene konfiguracije itn. Za samo namestitev smo izbrali strežnik SQL ter orodja za upravljanje konfiguracije. Pomemben del je nastavitve pravic (Priloga 3, Slika 2), ki nam bodo omogočale dostopanje do strežnika znotraj orodja SQL Server 2014 Management Studio ali na drugi strani dostopanje do same baze preko spletne aplikacije. V pravice dodamo tudi domenskega uporabnika, ki mu omogočimo dostop do naše podatkovne baze in hkrati bomo z njim izvajali delovanje spletne strani na IIS strežniku. Ker gre za specifičnega uporabnika, le-tega v zgornji sliki zaradi varnosti nismo prikazali.

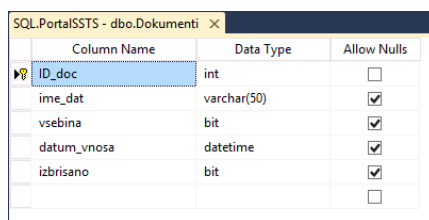
Slika 30: SQL Server Management Studio



Ob zaključku namestitve lahko uporabimo zgoraj prikazano orodje SQL Server Management Studio za povezavo do našega SQL strežnika, kjer bomo ustvarili nova podatkovno bazo za našo aplikacijo. Za izdelavo podatkovne baze se bomo oprli na konceptualni model, ki smo ga predhodno že pripravili. S klikom na New Database kreiramo novo podatkovno bazo PortalSSTS (Priloga 3, Slika 3). Poimenovanje ustvarjene baze lahko podamo poljubno, je pa smotrno upoštevati priporočila, ki so enaka tistim običajnim, kot je npr. poimenovanje s celotno besedno zvezo, izogibamo se presledkom v imenu, šumniki niso priporočljivi itn. Znotraj novo ustvarjene podatkovne baze kreiramo ustrezne tabele z atributi, kot smo jih predhodno načrtovali.

Ob kliku na izbrano tabelo izberemo urejanje oz. oblikovanje tabele, kjer vnesemo ustrezne attribute ter podatkovne tipe, ki jih bomo potrebovali. Prikaz tabele z atributi in podatkovnimi tipi je prikazan na naslednji sliki.

Slika 31: Prikaz tabele Dokumenti



Column Name	Data Type	Allow Nulls
ID_doc	int	<input type="checkbox"/>
ime_dat	varchar(50)	<input checked="" type="checkbox"/>
vsebina	bit	<input checked="" type="checkbox"/>
datum_vnosa	datetime	<input checked="" type="checkbox"/>
izbrisano	bit	<input checked="" type="checkbox"/>

V programskem okolju Visual Studio, ob povezavi na našo bazo, preverimo samo bazo (Priloga 3, Slika 4), kjer lahko tudi ustrezno uredimo povezave in številčnost.

S tem smo postavitev in konfiguracijo podatkovne baze zaključili in se lahko lotimo konfiguracije aktivnega imenika.

4.4.2.2 Konfiguracija Aktivnega imenika (AD)

Urejanje pravic uporabnikov bi lahko nadzirali preko lastnosti v aplikaciji. Pravice uporabnika bodo vplivale na možnosti, ki bodo uporabniku na voljo. Ker pa bi morali pravice nastavljati vsakemu uporabniku posebej, lahko to uredimo z integracijo pravic v aplikacijo, kjer določimo možnosti skupin uporabnikov, nastavimo ustrezna imena skupin ob postavitvi IIS strežnika in s tem rešimo problem določanja pravic. Tak način določanja pravic zahteva od nas konfiguracijo dveh dodatnih varnostnih skupin znotraj aktivnega imenika, in sicer bomo ustvarili Zaposleni Admins (Priloga 4, Slika 1) za administratorje aplikacije ter Zaposleni Users (Priloga 4, Slika 2) za uporabnike aplikacije. Ime skupine lahko določimo sami glede na želje in preference, vendar moramo paziti, da bomo pravice v aktivnem imeniku in lastnostnih aplikacije IIS strežnika nastavili enako.

V podanem primeru je pomembna samo skupina Zaposleni Admins. Ostale skupine so potrebne zaradi pravic uporabnika v domenskem okolju in pravic, ki so vezane na delo uporabnika.

Na podlagi izbire ustrezne skupine, kateri bo uporabnik pripadal, bo uporabnik dobil ustrezne pravice znotraj spletne aplikacije. Pravice so tako hitreje in lažje nadzirane.

4.4.2.3 Konfiguracija IIS

Konfiguracija IIS ni zahtevna s stališča vklopa storitve, bo pa potrebno prilagoditi nekaj nastavitvev, ki jih bomo morali vključiti za delovanje naše ASP aplikacije. Na samem začetku omogočimo storitev strežnika IIS znotraj upravljanja in nadzora vlog in funkcionalnosti Windows strežnika. Ob pogledu na osnovni vmesnik (Priloga 5, Slika 1) se nam prikaže privzeta spletna stran ter nastavitve, ki jih v povezavi z uporabo strežnika lahko uporabimo.

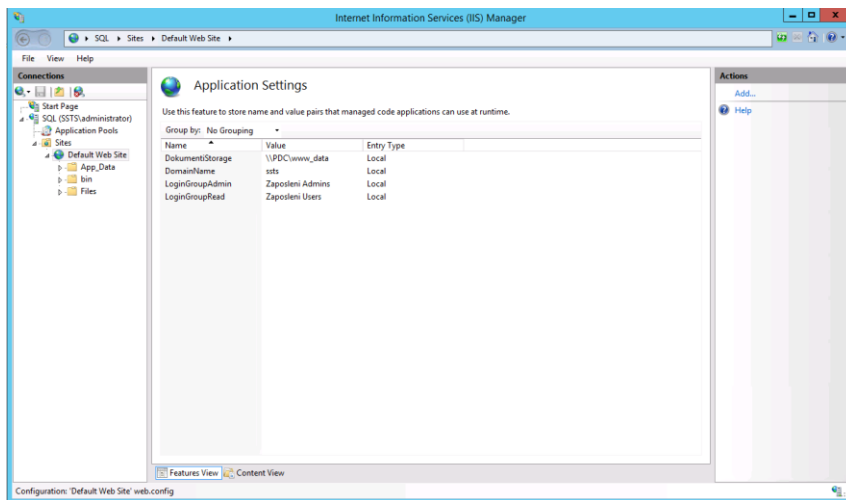
Kot smo omenili v poglavju postavitve IIS v razdelku 4.4.1.1, moramo določiti ustrezne nastavitve, preko katerih bo aplikacija pridobila ustrezne parametre za branje uporabnikov, shranjevanje dokumentov in domensko okolje.

V programski kodi bomo določili nastavitve, ki bodo preko sledečih elementov prejeli ustrezne lastnosti:

- DokumentStorage: pot za shranjevanje podatkov.
- DomainName: ime domenskega okolja.
- LoginGroupAdmins: ime varnostne skupine v aktivnem imeniku za določanje pravic administratorju.
- LoginGroupRead: ime varnostne skupine v aktivnem imeniku za določanje pravic uporabnikom.

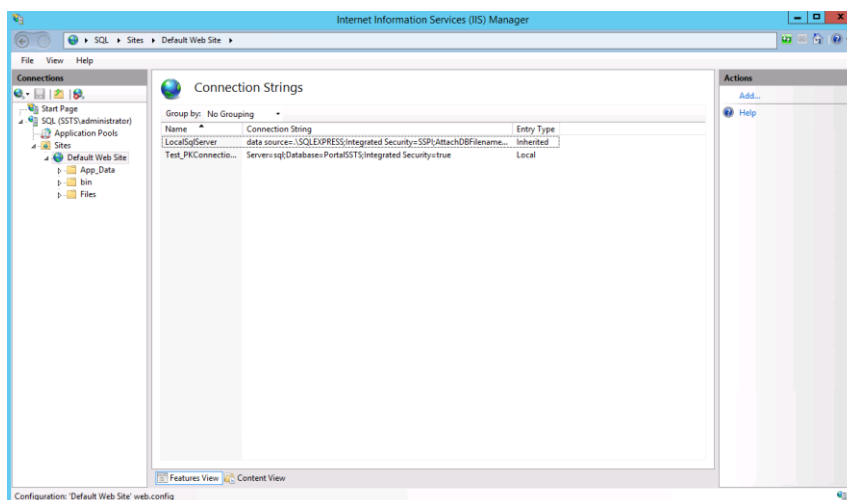
Nastavitve, ki so nujno potrebne za pravilno delovanje aplikacije, določimo preko zavihka nastavitve znotraj IIS pod Application settings. Polja lahko vnesemo ročno v naprej ali jih zgolj spremenimo, ko prenesemo spletno stran znotraj ustrezne mape IIS strežnika.

Slika 32: Določanje parametrov za lastnosti aplikacije znotraj IIS



V sami aplikaciji je določena tudi pot za dostop do strežnika SQL. Ker se lahko pot spreminja, lahko le-to popravimo preko nastavitve IIS strežnika pod zavihkom Connection Settings. Nastavitve lahko preverimo s pomočjo aplikacije ali z malce iznajdljivosti. V kolikor ustvarimo prazen dokument tipa .UDL. Novo ustvarjeni dokument nam omogoči dostop do orodja, kjer lahko vnesemo podatke imena strežnika SQL, ime podatkovne baze in ustrezne pravice ter testiramo povezavo.

Slika 33: Nastavitev povezave do podatkovne baze

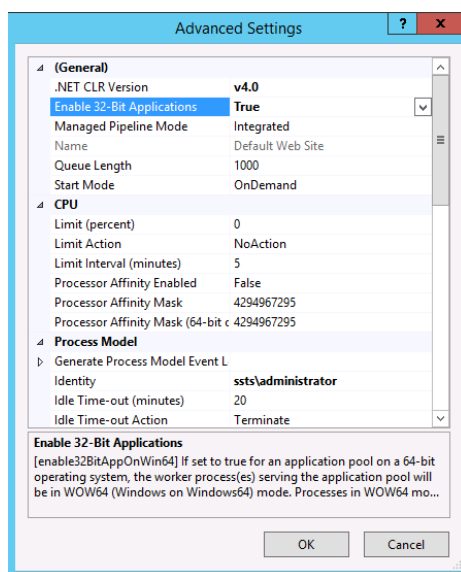


Priporočljivo je, da uspešnost povezave preverimo s prej opisanim postopkom ali kako drugače, saj bomo sicer naleteli kasneje ob testiranju na napako, ki bo povezana z neuspešno povezavo, ki pa bo precej manj razumljiva, kot v prej opisanem postopku. Seveda je na voljo več različnih možnosti za testiranje, vendar je prikazana ena izmed bolj preprostih in uspešnih.

Sledi še konfiguracija privzete spletne strani oz. dodatne (Priloga 5, Slika 2), če uporabljamo strežnik za serviranje različnih spletnih rešitev. Pomembno je, da nastavimo pot do strani z ustreznimi pravicami ter naslov, kjer bo spletna stran dosegljiva.

Ena izmed zelo pomembnih konfiguracij je nastavitev podpore zaganjanju ASP aplikacije, ki je 32 bitna. Razlog se skriva v tem, da je strežnik v različici Windows Server 2012 R2 privzeto 64 biten.

Slika 34: Nastavitev podpore 32 bitni aplikaciji



V nastavitvah moramo omogočiti tudi brskanje po datotekah (Data Browsing) in omogočiti preverjanje uporabnikov preko Windows avtentikacije. S tem je končan postopek priprave strežnikov in programske opreme.

4.4.2.4 Izdelava spletne aplikacije

Za izdelavo spletne aplikacije lahko uporabimo različna orodja. Orodja so lahko brezplačna ali plačljiva, odvisno od načina dela, njihove funkcionalnosti in tudi načina dela (Krebelj, 2015).

Izdelava spletne strani se začne s pisanjem označevalnega jezika (v nadaljevanju HTML) in prekrivnih slogov (v nadaljevanju CSS). Kot je zapisal Lloyd (2011), se sprva lotimo izdelave ogrodja spletne strani. Ogrodje nam predstavlja osnovne elemente HTML, ki nam omogočajo pripravo dokumenta HTML in njegov pravilen prikaz.

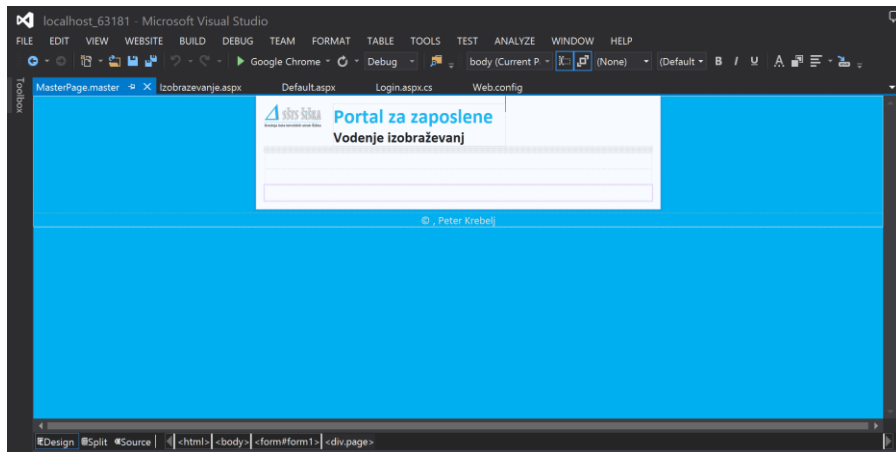
Celotna infrastruktura temelji na Microsoft rešitvah. V procesu analize smo se odločili, da bomo spletno aplikacijo postavili na strežnik IIS in bo le ta zgrajena v okolju .NET. Visual Studio nam ponuja več različic, ki so na voljo uporabnikom. Ena izmed njih je Express for Web, ki je namenjena izdelavi spletne strani v okolju .NET in je brezplačna. Na voljo so nam tudi različne distribucije, ki so namenjene podjetjem in organizacijam, kot so Ultimate, Premium, Test Pro, Pro, MSDN Platforms in na drugi strani tudi oblačne rešitve, kjer nam je na voljo brezplačna uporaba osnovnega paketa (Martin & Banks, 2014).

ASP.NET strani se generirajo dinamično na zahtevo, ko se izvede dostop do spletne strani. Konkretnije bi lahko rekli, da se stran generira takrat, ko se izvede zahtevek za prenos in prikaz spletne vsebine, ki se nahaja na spletnem strežniku. Dinamično generiranje spletnih strani ni značilno zgolj za ASP.NET, temveč gre tukaj tudi za upravljanje spletnih procesov, nadzor spletnega vmesnika, upravljanje globalnega global.aspx dokumenta in še bi lahko naštevali. ASP.NET HTTP proces poskrbi, da se spletni objekt, ki se pošlje kot odgovor na zahtevo ustrezno generira (Esposito, 2011, str. 169 – 217).

Za razvoj naše rešitve bomo uporabili Visual Studio 2013 Pro, ki nam je na voljo v sklopu paketa Dreamspark. Za samo oblikovanje nam ponuja Visual Studio vse potrebno, lahko pa si pomagamo tudi z orodjem Adobe Dreamviewer.

Začnemo z izdelavo MasterPage.aspx, saj nam bo ta osnovna oblika služila za osnovo s stališča podobe in osnovne konfiguracije skozi vse dodane podstrani. Koda za oblikovanje strani HTML se v Visual Studiu za osnovno postavitev generira samodejno, kar nam olajša predvsem prvi korak, in sicer pisanje ter izdelavo ogrodja. V vseh naslednjih korakih se nam koda za posamezne elemente generira ob dodajanju gradnikov, vseeno pa moramo dele kode popraviti tudi ročno.

Slika 35: Visual Studio MasterPage



Kot lahko vidimo, je začetna stran precej prazna, ker ne vsebuje nobenih posebnih elementov, saj se bo jedro strani vseskozi spreminjalo glede na podstrani.

Preverjanje se je ob obisku strani preveri, ali smo uspešno prijavljeni. V primeru, da nismo prijavljeni, smo preusmerjeni na stran za prijavo (Priloga 6, Slika 1), kjer se izvede preverjanje vnesenega uporabniškega imena in gesla.

Na spodnji sliki je prikazana funkcija, ki poskrbi za prijavo uporabnika. Sprva preverimo, ali sta v obe polji, pod uporabniško ime in geslo, vpisana podatka. V kolikor le teh nismo izpisali, nam bo funkcija vrnila vrednost false. Če so podatki uspešno vpisani, bomo izvedli preverjanje uporabnika v aktivnem imeniku. Podamo pot za dostop do lokacije protokola za dostop do imenikov na osnovni arhitekture odjemalec-strežnik (v nadaljevanju LDAP) in obvezno podamo ime domene, v kateri preverjamo vpisano uporabniško ime in geslo.

Slika 36: Preverjanje uporabnika v domenskem okolju

```
public bool AuthenticateGroup(string userName, string password, string domain, string group)
{
    if (userName == "" || password == "")
    {
        return false;
    }

    try
    {
        DirectoryEntry entry = new DirectoryEntry("LDAP://" + "SSTS", userName, password);
        DirectorySearcher mySearcher = new DirectorySearcher(entry);
        mySearcher.Filter = "(objectClass=user)(cn=" + userName + ")(sAMAccountName=" + userName + ")))";
        SearchResult result = mySearcher.FindOne();

        return IsInGroup(userName, group);
    }
    catch (DirectoryServicesCOMException)
    {
        return false;
    }
}
```

V kolikor bi spletno aplikacijo prenesli v neko drugo okolje, bi morali nastavitve domene popraviti v sami kodi, saj je v naprej predvideno domensko okolje SSTS, torej okolje naše organizacije.

Prikaz podatkov, ki jih vidi uporabnik, omejimo glede na njegove pravice (Priloga 6, Slika 2). Preverjanje se je izvedeno glede na to ali je uporabnik sploh prijavljen in če je uspešno prijavljen, nas zanima ali je zgolj uporabnik ali je administrator.

V primeru uspešne prijave nam izpiše uporabniško ime uporabnika, ki je prikazano na prikazu strani. Poleg imena izpišemo tudi vrsto uporabnika, torej Uporabnik ali Administrator. Običajni uporabnik dobi prikazana le zavihka privzete strani ter zavihka izobraževanj, medtem ko administrator dobi vpogled nad vsemi ostalimi elementi in zavihki znotraj spletne aplikacije.

Prikazovanje podatkov o izvedbi izobraževanj, kot tudi vseh ostalih podatkov na strani, npr. izobraževanja, organizator, zaposleni so izvedeni preko GridView (Priloga 6, Slika 3), ki nam omogoča enostaven prikaz zelenih podatkov. Izvedemo le povezavo na ustrezno bazo in izberemo ustrezno tabelo z ustreznimi polji, ki jih želimo prikazati. Na takšen način izvajamo prikaz podatkov iz same baze.

Način določanja prikazanih polj izvedemo preko določanja zelenih prikazanih polj znotraj lastnosti prikaza izobraževanj zaposlenega (Priloga 6, Slika 4).

Zavihek izobraževanj je namenjen upravljanju izobraževanj uporabnikov. Običajni uporabnik vidi vnesena izobraževanja in jih lahko potrdi kot opravljena. Administrator je tisti, ki bo izobraževanje vnesel v sam sistem. Vnos lahko izbrišemo ali uredimo, v kolikor bi se izkazalo, da so podatki napačni ali izobraževanje ni bilo izvedeno. Število ur vedno predstavlja cela števila, pridobljene točke pa so lahko cela ali decimalna števila. Pri vnosu moramo biti pozorni na podajanje pravilne oblike datuma in ure. Vnos lahko izvedemo zgolj preko vnosa datuma, vendar to pomeni, da se nam bo samodejno dodala ura v obliki samih ničel. Če želimo imeti natančen podatek o izvedbi, torej datum in uro, moramo biti pozorni tudi pri vnosu. Podatek podamo v obliki DD:MM:YYYY HH:MM:SS, kar predstavlja evropski zapis časa.

Prikaz testnih podatkov je prikazan na spodnji sliki za primer uporabnika z administratorskimi pravicami znotraj spletne aplikacije.

Slika 37: Prikaz vodenja izobraževanj

SSŠS Šiška
Srednja šola tehniških strokov Šiška

Portal za zaposlene
Vodenje izobraževanj

Prijavljen: **PETERK** (Administrator) - [ODJAVA](#)

Moja izobraževanja | Izobraževanje | Zaposleni | Vrsta izobraževanj | Organizator

Izobraževanje

Novo izobraževanje uspešno dodano v bazo

[Dodaj](#)

	Vrsta izobraževanja	stevilo ur	stevilo točk	Datum OD	Datum DO	Organizator	Opravljeno?
Edit Delete	Strokovno	8	0,5	1.7.2015 0:00:00	1.7.2015 0:00:00	CPI	Opravljeno
Edit Delete	Strokovno	24	1,5	18.5.2015 0:00:00	20.5.2015 0:00:00	Microsoft	Opravljeno
Edit Delete	Strokovno	8	0,5	1.8.2015 0:00:00	1.8.2015 0:00:00	CPI	Opravljeno
Edit Delete	Strokovno	16	1,5	1.7.2015 0:00:00	7.7.2015 0:00:00	CPI	Opravljeno
Edit Delete	Strokovno	8	0,5	27.5.2015 0:00:00	27.5.2015 0:00:00	FRI	Označi kot opravljeno

© 2015, Peter Krebelj

Primer dodajanja novega izobraževanja je prikazan na spodnji sliki. Vnosna polja predstavljajo minimalne zahteve, ki jih potrebujemo pri samem vodenju.

Slika 38: Dodajanje izobraževanja

Izobraževanje

[Dodaj](#)

Vrsta izobraževanja:

Število ur:

Število točk:

Datum OD:

Datum DO:

Organizator:

V primeru, da so vneseni vsi podatki v vseh poljih in se ob preverjanju ne zgodi napaka, izvedemo poskus zapisovanja v podatkovno bazo. Preko DataSetTableAdapters izvedemo zapis podatkov, kjer moramo podati vse zahtevane attribute, da bo zapis uspešno izvršen. V poteku zapisa izvedemo tudi pretvarjanje podatkov v ustrezno obliko in s tem omogočimo končni postopek zapisa.

Če pride do napake pri izvedbi zapisa, nas aplikacija opozori z napisom, da je prišlo do napake pri vnosu in besedilo prikaže z rdečo barvo (Priloga 6, Slika 5). Napaka ni bolj podrobna, saj bi zahtevala več preverjanj in obvestil, kar bi na eni strani uporabniku morda

koristilo, da bi videl kaj je vnesel napačno ali ga na drugi strani še bolj zbegalo, ker morda napake ne bi razumel.

Vsako izobraževanje nam prinese končno potrdilo o udeležbi ali certifikat, kar lahko znotraj uspešno opravljenih izobraževanj tudi dodamo kot dokument. S tem je celota vodenja izobraževanja zaključena in imamo na voljo pregled opravljenih izobraževanj po uporabniku.

Slika 39: Administratorski pogled nad izobraževanji

Portal za zaposlene
Vodenje izobraževanj

Prijavljen: **PETERK** (Administrator) - [ODJAVA](#)

Moja izobraževanja | Izobraževanje | Zaposleni | Vrsta izobraževanj | Organizator

Moja uspešno opravljenja izobraževanja

	Vrsta izobraževanja	stevilo ur	stevilo točk	Datum OD	Datum DO	Organizator	Poročilo
Edit	Strokovno	24	1,5	18.5.2015 0:00:00	20.5.2015 0:00:00	Microsoft	580_NTK2015.docx
Edit	Strokovno	8	0,5	1.7.2015 0:00:00	1.7.2015 0:00:00	CPI	Dodaj prilogo
Edit	Strokovno	16	1,5	1.7.2015 0:00:00	7.7.2015 0:00:00	CPI	Dodaj prilogo
Edit	Strokovno	8	0,5	1.8.2015 0:00:00	1.8.2015 0:00:00	CPI	Dodaj prilogo

Administratorski pogled po vseh zaposlenih

Peter Krebelj: Strokovno	8	0,50	01.07.2015 00:00 - 01.07.2015 00:00	CPI	Ni priloge
Peter Krebelj: Strokovno	24	1,50	18.05.2015 00:00 - 20.05.2015 00:00	Microsoft	580_NTK2015.docx
Peter Krebelj: Strokovno	8	0,50	01.08.2015 00:00 - 01.08.2015 00:00	CPI	Ni priloge
Peter Krebelj: Strokovno	16	1,50	01.07.2015 00:00 - 07.07.2015 00:00	CPI	Ni priloge
Janez Novak: Strokovno	8	0,50	01.07.2015 00:00 - 01.07.2015 00:00	CPI	Ni priloge
Janez Novak: Strokovno	24	1,50	18.05.2015 00:00 - 20.05.2015 00:00	Microsoft	Ni priloge
Janez Novak: Strokovno	8	0,50	01.08.2015 00:00 - 01.08.2015 00:00	CPI	Ni priloge

Zamenjaj leto:

© 2015, Peter Krebelj

Na sliki (Priloga 6, Slika 6) je prikazana koda za zapis dokumenta v bazo podatkov v povezavi s samim izobraževanjem. Možnost DokumentiStorage predstavlja lokacijo, ki smo jo morali v IIS strežniku nastaviti zaradi shranjevanja. Sprva se preveri nastavitve poti za shranjevanje dokumentov.

Zapis v bazo se izvede na podlagi povezave ID uporabnika in ID izobraževanja, kar omogoča povezovanje dokumenta in pravi izpis.

Poglejmo si še izpis uporabnika, ki ima zgolj pravice Uporabnik. Uporabnik ima prikazana njegova potrjena izobraževanja, kjer ima možnost dodati priponko o izvedbi.

Slika 40: Prikaz izobraževanj zaposlenega



Portal za zaposlene
Vodenje izobraževanj

Prijavljen: BILKOD (Uporabnik) - ODJAVA

Moja izobraževanja Izobraževanje

Moja uspešno opravljena izobraževanja

	Vrsta izobraževanja	stevilo ur	stevilo tock	Datum OD	Datum DO	Organizator	Poročilo
Edit	Strokovno	24	1,5	18.5.2015 0:00:00	20.5.2015 0:00:00	Microsoft	Dodaj prilogo
Edit	Strokovno	8	0,5	1.7.2015 0:00:00	1.7.2015 0:00:00	CPI	Dodaj prilogo
Edit	Strokovno	8	0,5	1.8.2015 0:00:00	1.8.2015 0:00:00	CPI	Dodaj prilogo

Prikazana rešitev predstavlja prvo različico izdelave spletne aplikacije, kjer se bodo gotovo izkazale kakšne pomanjkljivosti ali težave, ki bodo sledile ob uporabi. Skozi povratne informacije uporabnikov bodo nepredvidene težave in problemi odpravljeni in hkrati bodo zagotovo uvedene tudi kasnejše izboljšave, če se bo izkazalo, da bodo vodstvo in uporabniki rešitev sprejeli. Čas bo edini pravi pokazatelj, ki bo pokazal uspešnost vpeljave in kakovost rešitve.

Preden pa bomo rešitev ponudili končnemu uporabniku, izvedemo še prenos aplikacije na ustrezno lokacijo. Strežnik IIS je že postavljen, vendar nam poleg prikaza začetnega okna IIS ne bo prikazal naše rešitve. Kot so zapisali avtorji (Nagel, Evjen, Glynn, Watson & Skinner, 2013, str. 477 – 484) moramo pred prenosom aplikacije sprva preveriti ali bo povezava do podatkovne baze ostala enaka ali jo bo treba po prenosu spreminjati. Odvisno od načina dela in testiranja pred končno namestitvijo v produkcijsko okolje. V Visual Studio nam je na voljo orodje, ki nam omogoča neposredno povezavo na strežnik in objavo spletne aplikacije ali izdelavo paketa, ki ga bomo sami prenesli v ustrezno mapo na strežniku. Zaradi dela s testnim okoljem doma, bomo v našem primeru ustvarili spletni paket, ki ga le kopiramo v ustrezno mapo strežnika IIS.

4.5 Analiza rešitve

Naša začetna zahteva je bila analiza trenutnega tehnološkega okolja in poslovnih procesov, kjer smo se usmerili v poslovni proces nadaljnjih izobraževanj in izpopolnjevanj zaposlenih. Zaradi neposredne vpletenosti v proces prenove tehnološkega okolja in tudi vzdrževanja tega okolja, je bilo pridobivanje potrebnih informacij praktično neomejeno in hkrati je temu primerno tudi poznavanje obstoječega sistema pri iskanju novih rešitev. Cilj, ki smo si ga zadali, je bila sprememba omenjenega poslovnega procesa in njegovo podprtje s pomočjo izdelave spletne aplikacije.

Da bi uspešno preverili vse dane zahteve, bomo s pomočjo enostavne tabele preverili ali smo zadostili zahtevam znotraj same aplikacije in seveda kaj si o njej mislijo uporabniki, ki bodo oceno uspešnosti videli drugače kot sami snovalci.

Tabela 1: Pregled zahtev in doseženih ciljev aplikacije

Zahteve	Doseženi cilji
Aplikacija zahteva prijavo uporabnika	Uporabnik mora vpisati enolično uporabniško ime in geslo.
Prijava mora biti povezana z aktivnim imenikom	Preverjanje uporabniškega imena in gesla se izvede v Microsoft aktivnem imeniku (AD).
Omogočene morajo biti ločene uporabniške pravice	Integrirana je možnost pogleda skrbnika aplikacije in uporabnika, kjer je omejen prikaz in s tem funkcionalnost aplikacije glede na pravice uporabnika. Določitev pravic je vezana na ustrezno razvrstitev uporabnika znotraj skupine v AD.
Aplikacija naj deluje na različnih operacijskih sistemih	Ker gre za spletno aplikacijo, uporaba ni omejena na operacijski sistem ali na brskalnik, ki ga uporabljamo.
Omogočati mora dodajanje dokumentov	Dodajanje je omogočeno preko poti, ki jo podamo v lastnostih aplikacije znotraj IIS.
Vodstvo poskrbi za vnos izobraževanj	Vnos je vezan na administratorske pravice uporabnika, torej na vodstvo.
Omogoča nam vnašanje pripadajočih točk za izobraževanje	Uporabnik lahko izvede vnos točk glede na opravljeno število ur. Zaradi možnosti vnosa napačnih podatkov je na voljo urejanje vnesenih podatkov.
Omogoča nam izpis izobraževanj tekočega leta	Filter je izveden glede na letnico, privzeto je to leto 2015.
Omogočeno mora biti dodajanje izobraževanj	Administrator je tisti, ki izobraževanje lahko dodaja.
Dostop do aplikacije mora biti omejen glede na omrežja	Sama aplikacija deluje na spletnem strežniku, torej je sistemski skrbnik omrežja tisti, ki ureja in določa zmožnosti dostopa.
V celoti mora podpirati tehnološko infrastrukturo	Na podlagi analize je aplikacija narejena z Microsoft tehnologijo ASP .NET, kar ustreza trenutnemu tehnološkemu okolju.
Podpirati mora spremenjeni poslovni proces izobraževanj zaposlenih	Zmožnost vnosa izobraževanja pred samo izvedbo omogoča lažji nadzor izobraževanj kot tudi spremljanje le teh.

Ob pregledu prej podanih zahtev in rešitev v tabeli (Tabela 1) je razvidno, da so bile vnesene vse zahteve vodstva, ki jih mora aplikacija vsebovati. Skupaj z vodstvom smo izvedli sestanek, kjer sta bili prisotni poslovna sekretarka in ravnateljica, da je bil izveden prikaz delovanja in uporabe aplikacije z vsemi funkcionalnostmi. Ugotovljeno je bilo, da so pokrite zahtevane značilnosti in je aplikacija pripravljena za končne uporabnike, ki jo bodo testirali in podali končno mnenje, kot tudi njihov vidik uporabe ter predloge, ki bi aplikacijo še izboljšali. Sprva je bilo predvideno, da bi izvedli tudi anketo med končnimi uporabniki. Ker je sam proces od analize, do iskanja končne rešitve, kot tudi izdelave

končne aplikacije dolgotrajen proces, ne bi bilo smiselno izvajati ankete med uporabniki, kateri bi aplikacijo prejeli v uporabo, a je morda ne bi niti preverili ali bi le to preložili. Smiselno bi bilo tudi, če bi izbrali manjšo skupino uporabnikov izmed zaposlenih, kateri bi aplikacijo preverili in podali svoj pogled, preden bi v uporabo vključili vse uporabnike. Tako bi bil prehod postopen, saj bi postopoma odkrili napake v delovanju, izučili nekaj uporabnikov, ki bi bili kasneje tudi v pomoč ostalim zaposlenim v postopku spoznavanja in uporabe aplikacije. Ker vsi uporabniki uporabljajo rešitev eAsistent za vodenje procesa poučevanja, bi bila to ena izmed velikih prednosti pri samem uvajanju, saj je koncept dela s spletnimi aplikacijami uporabnikom že dobro poznan.

Izvedba ankete bi bila smiselna po daljšem časovnem obdobju, ko bi uporabniki aplikacijo že dobro spoznali in jo tudi redno uporabljali, saj bi le tako lahko pričakovali iskrene odgovore, ki bi nam vračali sliko zadovoljstva in sprejetja aplikacije. V nasprotnem primeru, bi hitra uvedba aplikacije, kjer bi zahtevali takojšnjo uporabo, lahko vodila v negativni odziv zaposlenih in s tem tudi popačene rezultate, ki bi jih prikazala anketa.

Omenimo tudi prednosti, ki jih takšna aplikacija prinese s stališča celotne organizacije. Ena pomembnejših je gotovo vpogled v zgodovino izobraževanj vseh zaposlenih. V primeru, da se zaposleni odloči za napredovanje, bo tako ob hitrem pregledu ugotovil, katera potrdila mora priložiti. Tako zaposleni kot vodstvo bosta na takšen način hitro preverila ali so bila opravljena vsa potrebna izobraževanja. Aplikacija bo tudi poskrbela, da bo poslovna sekretarka manj obremenjena z vnosom in bo zaposleni tisti, ki bo moral prevzeti del odgovornosti za vnos potrdila o izobraževanju.

Slabost uporabe se bi lahko izkazala v primeru, da uporabniki ne bi vnašali potrdil o izobraževanjih in bi tako kasneje ugotovili, da nimajo podatka o izvedbi. Tudi v kolikor bi prišlo do nedelovanja sistema ali težav s podatkovno bazo in varnostna kopija ne bi obstajala, bi to vodilo v slabo voljo in dodatno delo in s tem nepotrebno porabo časa. Kot je bilo omenjeno skozi nalogo, prinaša aplikacija tudi nekaj dodatnih stroškov, ki nastanejo zaradi najema licence strežnika SQL in strežnika Windows Server 2012 R2.

SKLEP

Problematika in želja vodstva izobraževalne ustanove je bila tista, ki nam je odprla problem magistrske naloge, skozi katero smo se lotili reševanja konkretnega problema znotraj izbrane organizacije. Izobraževanja so za zaposlene v šolskem prostoru nujna, tako z vidika pridobivanja novih znanj kot na drugi strani zahteve po izobraževanju, ki je predpisana in zahtevana. Tretji vidik so napredovanja, kjer je pridobivanje točk iz naslova izobraževanj pomembno predvsem za tiste posameznike, ki lahko v plačilne razrede in nazive še vedno napredujejo. Pridobivanje točk iz naslova napredovanj je zagotovo manj pomembno tistim, ki so v šolstvu dalj časa in jim nova izobraževanja pomenijo zgolj pridobivanje znanja in celostno osebnostno rast.

Med procesom izdelave pričujočega dela smo se seznanili z izbrano ustanovo, ki predstavlja srednje veliko organizacijo, ki se še danes širi. Z večanjem števila zaposlenih narašča tudi zahteva po lažjem vodenju in evidenci izvedenih izobraževanj, ki so za zaposlene nujna. Da bi vodstvu omogočili hitrejši proces vnosa ustreznih podatkov v aplikacijo in s tem tudi boljši pregled nad dogajanjem, ki ga je v takšnem številu težko voditi v Microsoft Excel okolju, je bila izdelana celotna analiza okolja in zahtev, kot tudi pregled zakonske osnove, ki je ob tem pomembna.

Ker je organizacija usmerjena v uporabo Microsoft programske opreme, ki je s strani Ministrstva za izobraževanje, znanost in šport podprta tudi s pogodbo, ki organizaciji nudi dostop do programske opreme po nižji ceni, je bil izbor tehnologije na podlagi trenutnega okolja in ostalih informacij, toliko bolj usmerjen v uporabo Microsoft tehnologije v primeru lastne izdelave. Reševanje problema je omejeno na trenutne rešitve trga, kjer ustrezne aplikacije ni mogoče kupiti in jo je zato potrebno izdelati. S pomočjo odločitvenega modela smo izbrali razvoj lastne rešitve, saj nam le-ta omogoča izdelavo prilagojene rešitve končnemu uporabniku in nam ne podaja drugih nepotrebnih komponent, ki jih običajno vsebujejo splošne programske rešitve.

Skozi proces analize in iskanja vseh zahtev, ki jih mora aplikacija pokrivati, smo prišli do točke, kjer je bilo potrebno vedeti kakšno aplikacijo bomo izdelali, katero tehnologijo bomo uporabili in seveda zagotoviti ustrezno informacijsko okolje, kjer bomo aplikacijo izvajali. Sledila je konfiguracija strežnikov in vseh povezanih komponent, ki so potrebne za pravilno delovanje naše aplikacije. Prvotna postavitev strežnika, baze SQL kot tudi vseh ostalih delov, je bila izvedena na domačem strežniku, ker je bilo zaradi varnosti in tudi postopka razvoja lažje upravljati sistem in ob tem ni bilo potrebno paziti, da ne bi vplivali na delujoče produkcijsko okolje in s tem povzročili nedelovanje ali izpad informacijskega okolja.

Sledila je priprava rešitve v okolju Visual Studio, kjer je uporabljen programski jezik C#.NET. Na spletnem portalu Microsoft Technet so zbrani delčki kode, ki nam pomagajo pri iskanju odgovorov in rešitev, v kolikor se z določenim problemom še nismo srečali in kode za izvajanje določene funkcije ne poznamo. Skozi izgradnjo splošne spletne predloge, ki se v izbranem orodju imenuje MasterPage, smo pripravili osnovno postavitev spletne strani, kateri smo v osnovi izbrali eno izmed barv iz logotipa organizacije, v povezavi z drugimi barvami.

S pomočjo dodajanja posameznih funkcionalnosti, kjer govorimo o dostopanju do podatkovne baze, prikazovanju izbranih zapisov iz podatkovne baze, preverjanju uporabnika v aktivnem imeniku in še bi lahko naštevali, je postopoma nastala preprosta a delujoča aplikacija, ki nam omogoča zahtevana področja, ki so bila v povezavi z vodstvom zapisana v poglavju 4.5.

Uspešnost končne rešitve se v prvi vrsti kaže z odzivi vodstva, kjer je pomembno, da je bilo zadoščeno vsem danim zahtevam. V drugi vrsti bi končno zadovoljstvo preverili tudi med uporabniki. Da bi bili odgovori uporabnikov konstruktivni in utemeljeni, bi uporabnikom morali dati na voljo ustrezen čas, od nekaj mesecev do leta ali dveh, da bi se aplikacije postopoma privadili, jo začeli uporabljati in ob tem podali tudi odgovore, ki bi prikazovali njihov dejanski pogled.

Zaradi napisanega je končna ocena uspešnosti postavljena na pogovor vodstva, ki je bil v samem začetku tudi naročnik analize in izvedbe izdelave aplikacije. V našem primeru se bomo oprli tudi na dejstvo, da se naročnik strinja s tem, da so bile podane zahteve v celoti pokrite in izvedene po začetnem dogovoru. Kasnejšo analizo bi bilo smiselno izvesti v obliki ankete, ki bi jo razdelili med zaposlene in s tem dobili njihov odziv o zadovoljstvu in o uporabnosti pripravljene spletne aplikacije.

Naloga nam skozi izbrani primer prikazuje, kako poteka celoten proces prenove in uvedbe nove rešitve, torej od same zahteve po reševanju nekega poslovnega problema, do analize poslovnega in tehnološkega okolja, iskanja ustrezne rešitve, konfiguracije informacijske opreme, izdelave neke rešitve do končne implementacije. Ocena uspešnosti sloni le na odzivih, ki jih prejmemo s strani uporabnikov, v našem primeru bi rekli s strani vodstva organizacije. Takšna ocena ni popolna, vendar lahko na podlagi pogovora in zadovoljstva vodstva ob predstavitvi končnega rezultata rečemo, da smo bili z izdelavo uspešni, saj nas je pripeljala do ustrezne končne rešitve. Dodatno bi uspešnost lahko preverili preko ankete zaposlenih, vendar bo pravi pokazatelj, ali smo rešitev približali končnemu uporabniku, viden šele skozi daljše časovno obdobje, ko bodo uporabniki rešitev sprejeli za svojo ali pa jo bodo pozabili.

LITERATURA IN VIRI

1. Bajaj, A., & Wrycza, S. (2009). *System Analysis and Design for Advanced Modeling Methods: Best practices*. New York: IGI Global.
2. Barry, C., Conboy, K., Lang, M., Wojtkowski, G., & Wojtkowski, W. (2009). *Information System Development*. New York: Springer Science + Business Media, LLC.
3. Bojanc, R., Jerman-Blažič, B., & Tekavčič, M. (2014). *Informacijska varnost v podjetniškem okolju*. Ljubljana: Ekonomska fakulteta.
4. *Cenik licenc po pogodbi School Agreement*. Najdeno 15. junija 2015 na spletnem naslovu http://www.ena.com/oddelki/microsoft/izd_1389_1402185_najem_po_ees__1__leto
5. Date, C. (2011). *SQL and Relational Theory: How to Write Accurate SQL Code*. Sebastopol: O'Reilly Media, Inc.
6. Dimovski, V., Penger, S., & Škerlavaj, M. (2007). *Organiziranje in odločanje*. Ljubljana: Ekonomska fakulteta.
7. Esposito, D. (2011). *Programming ASP.NET*. Redmond: Microsoft Press
8. Gradišar, M. (2003). *Uvod v informatiko*. Ljubljana: Ekonomska fakulteta
9. Gradišar, M., Jaklič, J., Damij, T., & Baloh, P. (2005). *Osnove poslovne informatike*. Ljubljana: Ekonomska fakulteta.
10. Harkins, M. (2013). *Managing Risk and Information Security*. New York: Technics Publications, LLC.
11. Hoberman, S. (2009). *Data modeling made simple*. Bradley Beach: Technics Publications, LLC.
12. Hoberman, S., Burbank, D., & Bradley, C. (2009). *Data modeling for the business*. New York: Technics Publication, LLC.
13. Hovelja, T., Rožanec, A., & Rupnik, R. (2010). Measuring the success of the strategic information system planning in enterprises in Slovenia. *Management*, 15(2), 25-46.
14. *Katis – katalog programov nadaljnega izobraževanja in usposabljanja strokovnih delavcev*. Najdeno 15. junija 2015 na spletnem naslovu <http://lim1.mss.edus.si/katis/default.aspx>

15. Kenneth, L. S. (2012, 13. februar). A Decision Paradigm For Information Technology Applications. Najdeno 25. junija 2015 na spletnem naslovu <http://www.nevo.com/our-knowledge/whitepapers/BuildVsBuy.pdf>
16. Killmeyer Tudor, J. (2001). *Information Security Architecture*. Florida: CRC Press LCC.
17. Kolektivna pogodba za dejavnost vzgoje in izobraževanja v Republiki Sloveniji. *Uradni list RS* št. 52/94, 49/95, 34/96, 45/95 – popr., 51/98, 28/99, 39/99 – ZMPUPR, 39/00, 56/01, 64/01, 78/01 – popr., 56/02, 43/06 – ZkoIP, 60/08, 79/11, 40/12 in 46/13.
18. Kovačič, A., & Boksilj Vukšič, V. (2005). *Management poslovnih procesov*. Ljubljana: GV založba, d. o. o.
19. Kovačič, A., Groznik, A., & Ribičič, M. (2009). *Temelji elektronskega poslovanja*. Ljubljana: Ekonomska fakulteta v Ljubljani.
20. Kovačič, A., Jaklič, J., Indihar Štemberger, M., & Groznik, A. (2004). *Prenova in informatizacija poslovanja*. Ljubljana: Ekonomska fakulteta v Ljubljani.
21. Krebelj, P. (2015). *HTML in CSS za začetnike*. Ljubljana: Atelje Doria.
22. Lederer, A. L., & Sethi, V. (1996, 1. junij). Key prescriptions for strategic information systems planning. *Journal of Management Information Systems*, 13(1), 35-62.
23. Lloyd, I. (2011). *Build your own website the right way using HTML & CSS*. Collingwood VIC Australia: SitePoint Pty. Ltd.
24. Martin, J., & Banks, R. (2014). *Visual Studio 2013 Cookbook*. Birmingham, UK: Packt Publishing Ltd.
25. Nagel, C., Evjen, B., Glynn, J., Watson, K., & Skinner, M. (2013). *C# 2012 and .NET 4.5*. Indianapolis: John Wiley & Sons, Inc.
26. Pravilnik o napredovanju zaposlenih v vzgoji in izobraževanju v nazive. *Uradni list RS* št. 54/02, 123/08, 44/09 in 18/10.
27. Pravilnik o izobrazbi učiteljev in drugih strokovnih delavcev v poklicnem in strokovnem izobraževanju. *Uradni list RS* št. 48/11 in 92/12.
28. Prešern, S. (2006). *Poslovna informatika in internet za podjetnike in managerje*. Ljubljana: Fleks, d. o. o.
29. Reijers, A. H. (2006). Implementing BPM systems: the role of process orientation. *Business Process Management Journal*, 12(4), 389-409.

30. Satzinger, J., Jackson, R., & Burd, S. (2009). *System analysis & design in a changing world*. Boston, MA: Course Technology
31. Shelly, G. B., Cashman, T. J., & Rosenblatt, H. J. (2008). *System Analysis and Design* (7th ed.). Boston: Thomson Course Technology.
32. Turban, E., Sharda, E. R., & Delen, D. (2011). *Decision support and business intelligence systems*. Oklahoma: Oklahoma State University.
33. Turk, T. (2013). *Razvijanje informacijskih rešitev – sodobni pristopi razvoja*. Ljubljana: Ekonomska fakulteta.
34. *Ustanovitev šole in začetni koraki*. Najdeno 15. junija 2015 na spletnem naslovu www.ssts.si/zgodovina
35. Valacich, J. S., George, J. F., & Hoffer, J. A. (2009). *Essentials of System Analysis and Design*. New Jersey: Pearson Education, Inc.
36. Vidmar, T. (2002). *Informacijsko-komunikacijski sistem*. Ljubljana: Pasadena.
37. *VMware – Virtualization*. Najdeno 15. junija 2015 na spletnem naslovu <http://www.vmware.com/virtualization>
38. Zakon o organizaciji in financiranju vzgoje in izobraževanja. *Uradni list RS* št. 16/07 – uradno prečiščeno besedilo, 36/08, 58/09, 64/09 – popr., 65/09 – popr., 20/11, 40/12 – ZUJF, 57/12 – ZPCP-2D, 47/15: ZOFVI-NPB19.
39. Zakon o varstvu osebnih podatkov. *Uradni list RS* št. 94/07.
40. Zupan, N. (2003). *Organizacija podjetja*. Ljubljana: Ekonomska fakulteta

PRILOGE

KAZALO PRILOG

Priloga 1: Intervju z vodstvom šole.....	1
Priloga 2: Aktivacija podpore IIS.....	6
Priloga 3: Podatkovna baza SQL.....	7
Priloga 4: Konfiguracija aktivnega imenika.....	9
Priloga 5: Konfiguracija IIS	10
Priloga 6: Izdelava spletne aplikacije	11
Priloga 7: Skripta za kreiranje baze SQL	13
Priloga 8: Spletna aplikacija.....	17
Priloga 9: Seznam uporabljenih kratic.....	18

Priloga 1: Intervju z vodstvom šole

1. Kakšna je vaša trenutna funkcija znotraj organizacije SŠTS Šiška?
2. Koliko časa ste že zaposleni v tej organizaciji?
3. Kateri izobraževalni programi potekajo na SŠTS Šiška?
4. Kateri so za vas glavni cilji, ki naj jih v prihodnosti doseže šola?
5. Kakšen je vaš pogled v prihodnost glede razvoja organizacije kot celote?
6. Ali se vam zdi osebni razvoj kot tudi izobraževanje zaposlenih pomembno? Zakaj?
7. Katere so bistvene prednosti neprestanega strokovnega izobraževanja in izpopolnjevanja zaposlenih?
8. Ali menite, da je zakonsko predpisano minimalno število dni izobraževanj zadostno, da lahko strokovni delavci stopajo v koraku s časom?
9. V povezavi s prejšnjim vprašanjem. Kako ocenjujete zmožnost sledenja razvoju tehnike, glede na to, da ste predvsem tehnično usmerjena šola?
10. Učinkovito spremljanje posameznega zaposlenega je potrebno voditi. Kako vi vidite rešitev, da bi se to lahko izvajalo bolj učinkovito?
11. Imate v mislih kakšno konkretno rešitev, kako bi spremljali udeležbo zaposlenih na strokovnih izobraževanjih in izpopolnjevanjih?
12. Kaj je po vašem mnenju ključni dejavnik, da se zaposleni želijo izobraževati?
13. Ali menite, da je strokovno izobraževanje in izpopolnjevanje zaposlenih povezano tudi s pridobivanjem točk za napredovanje?
14. Menite, da bi boljši nadzor nad izobraževanji vodil v boljše rezultate ali bi se zgodilo ravno nasprotno?
15. V kolikor bi potek izobraževanj vodili v elektronski obliki, obstajajo kakšni zadržki povezani z zakonodajo?
16. Kakšna je vaša vizija vodenja izobraževanj zaposlenih v elektronski obliki?
17. Kaj bi pridobili z elektronsko obliko?
18. V kolikor bi takšna aplikacija podprla poslovni proces vodenja izobraževanj in ga s tem izboljšala, bi podprli takšen projekt?

Hvala za vaše sodelovanje.

Poslovna sekretarka

1. Kakšna je vaša trenutna funkcija znotraj organizacije SŠTS Šiška?
Opravljam delo poslovne sekretarke v Srednji šoli tehniških strok Šiška.
2. Koliko časa ste že zaposleni v tej organizaciji?
Skoraj 31 let.
3. Kateri izobraževalni programi potekajo na SŠTS Šiška?
Redno šolanje: SPI Električar in Računalnikar, SSI Tehnik mehatronike in PTI Elektrotehnik in Tehnik računalništva.

4. Kateri so za vas glavni cilji, ki naj jih v prihodnosti doseže šola?
Uspešna vpeljava višješolskega programa inženir mehatronike in stabilen vpis v vse izobraževalne programe.
5. Kakšen je vaš pogled v prihodnost glede razvoja organizacije kot celote?
Lahko bi postali center za tehniško izobraževanje.
6. Ali se vam zdi osebni razvoj kot tudi izobraževanje zaposlenih pomembno? Zakaj?
Da zdi se mi pomembno. V obeh primerih lahko bolje delujemo v okolju. Lažje se prilagajamo in dosežemo zastavljene cilje.
7. Katere so bistvene prednosti neprestanega strokovnega izobraževanja in izpopolnjevanja zaposlenih?
Z izobraževanjem tudi osebnostno zrastemo. Lažje spremljamo napredek in glede na to, da izvajamo zelo atraktivne programe izobraževanja, ki se zelo hitro razvijajo, s sprotnim izobraževanjem lahko sledimo napredku in ga prenašamo na mladino.
8. Ali menite, da je zakonsko predpisano minimalno število dni izobraževanj zadostno, da lahko strokovni delavci stopajo v koraku s časom?
Ne. Mislim da bi moralo biti tega vsaj 15 delovnih dni.
9. V povezavi s prejšnjim vprašanjem. Kako ocenjujete zmožnost sledenja razvoju tehnike, glede na to, da ste predvsem tehnično usmerjena šola?
Zelo težko. Učitelji se morajo izobraževati večinoma v svojem prostem času. Le del (7 dni) jim lahko omogoči šola. Prav tako je to povezano s stroški izobraževanj. Ti so velikokrat previsoki in marsikomu težko dostopni.
10. Učinkovito spremljanje posameznega zaposlenega je potrebno voditi. Kako vi vidite rešitev, da bi se to lahko izvajalo bolj učinkovito?
Že sedaj poskušamo reševati pregled izobraževanja z nekakšnimi samostojnimi Excel-ovimi datotekami, vendar pa žal največkrat zmanjka časa.
11. Imate v mislih kakšno konkretno rešitev, kako bi spremljali udeležbo zaposlenih na strokovnih izobraževanjih in izpopolnjevanjih?
Glede na to, da smo v dobi računalništva in da so vsi zaposleni v večini tudi računalniško pismeni, bi bil dobrodošel portal, kjer bi si vsak zaposleni beležil svoja izobraževanja in dodajal scane potrdil o udeležbi in opravljenem seminarju. Administrator pa bi lahko iz tega potegnil skupaj statistiko glede na pokazane potrebe (kdo, koliko dni, uspešno ali ne ...).
12. Kaj je po vašem mnenju ključni dejavnik, da se zaposleni želijo izobraževati?
Spremljanje novosti, ki bi jih lahko prenašali na mladino in uporabljali pri vsakodnevnem delu.
13. Ali menite, da je strokovno izobraževanje in izpopolnjevanje zaposlenih povezano tudi s pridobivanjem točk za napredovanje?
Pri mlajših definitivno. Pri starejših, ki pa nimajo več toliko možnosti za napredovanje pa je izobraževanje bolj za njihovo dušo in za to, da bi znanje predali mlajšim.
14. Menite, da bi boljši nadzor nad izobraževanji vodil v boljše rezultate ali bi se zgodilo ravno nasprotno?

Pri nekaterih bi to zbudilo odpor do izobraževanja, saj tako pri nekaterih prevladuje prepričanje, da jih nadzorujemo na vsakem koraku. Pri drugih pa bi dejansko bili lahko rezultati boljši, saj bi imeli tudi sami boljši pregled nad izobraževanjem, ki so se ga udeležili in koliko si ga še lahko privoščijo.

15. V kolikor bi potek izobraževanj vodili v elektronski obliki, obstajajo kakšni zadržki povezani z zakonodajo?
Če ta evidenca obstaja le za interno uporabo in ne za zbiranje podatkov navzven mislim, da zadržkov ne bi smelo biti.
16. Kakšna je vaša vizija vodenja izobraževanj zaposlenih v elektronski obliki?
Želja, da bi si vsak posameznik lahko sam vodil evidenco izobraževanja, hkrati pa olajšal delo tudi nam, v vodenju zavoda, z boljšo preglednostjo, kam, kdaj, in koga poslati na posamezno izobraževanje.
17. Kaj bi pridobili z elektronsko obliko vodenja izobraževanj?
Boljši pregled udeležbe na izobraževanju, prihranek časa ob administrativnem delu.
18. V kolikor bi takšna aplikacija podprla poslovni proces vodenja izobraževanj in ga s tem izboljšala, bi podprli takšen projekt?
Absolutno bi jo podprla, saj bi mi prihranila kar nekaj časa. Vsekakor bi trajalo, da bi zaposlene navadili na to, da je potrebno vpisovati izobraževanja v portal ampak dejstvo pa je, da bi po določenem času, ko bi bilo to delo osvojeno, vsem lahko zelo pripomoglo k preglednosti izobraževanj.

Ravnateljica šole

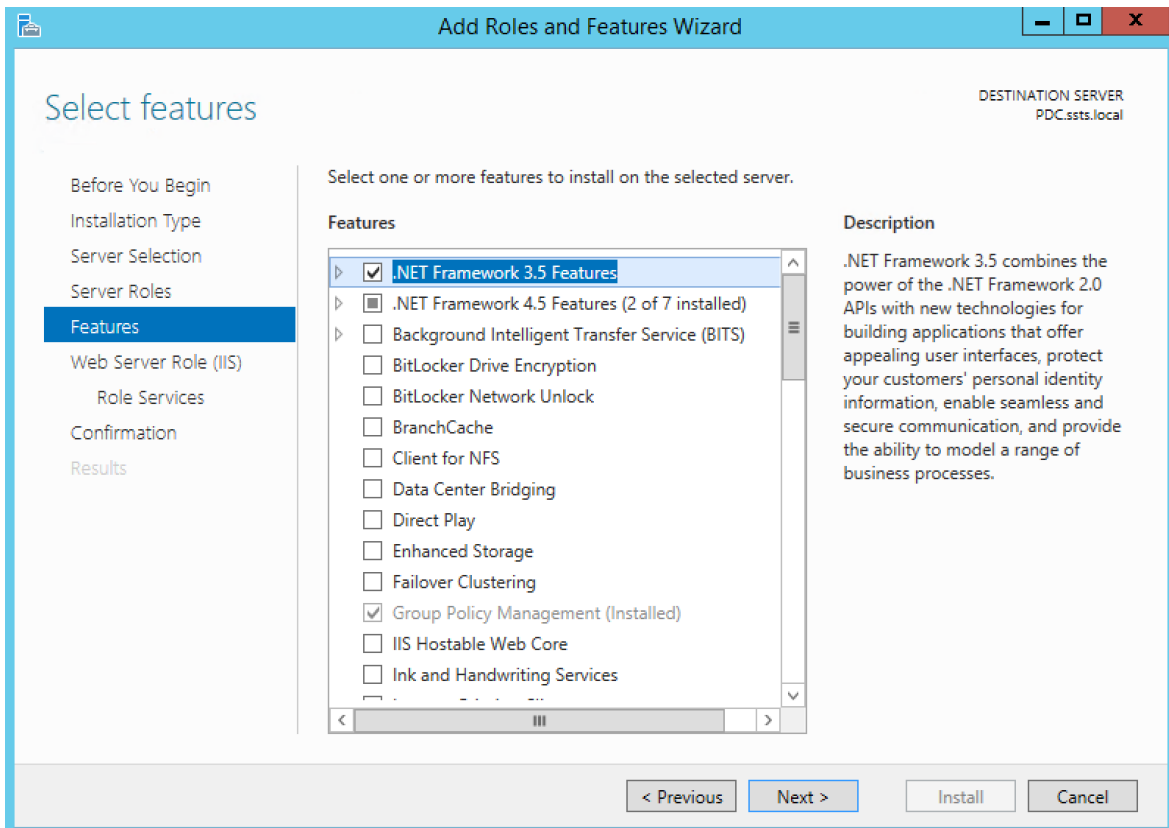
1. Kakšna je vaša trenutna funkcija znotraj organizacije SŠTS Šiška?
Ravnateljica šole.
2. Koliko časa ste že zaposleni v tej organizaciji?
23 let.
3. Kateri izobraževalni programi potekajo na SŠTS Šiška?
SPI električar,
SPI računalnikar,
SSI tehnik mehatronike,
PTI elektrotehnik,
PTI tehnik računalništva.
4. Kateri so za vas glavni cilji, ki naj jih v prihodnosti doseže šola?
Postati in ostati vodilna šola v regiji na področju izobraževanja za potrebe elektrotehnik, računalništva in mehatronike. Regiji zagotavljati dobro usposobljene kadre.
5. Kakšen je vaš pogled v prihodnost glede razvoja organizacije kot celote?
Šola mora biti tudi s pomočjo informacijske tehnologije dobro organizirana, kar pomeni, da procesi tečejo nemoteno, saj s tem omogočajo večjo kakovost strokovnega dela in manj neusklajenosti, podvajanja, pomanjkljivosti in odstopanj od načrtovanega dela.

6. Ali se vam zdi osebni razvoj kot tudi izobraževanje zaposlenih pomembno? Zakaj?
Vsi strokovni delavci oz. vsi zaposleni morajo slediti tako razvoju na strokovnem področju kot tudi razvoju pedagoških znanj. Zelo pomembna je tudi osebna rast in izobraževanje v tej smeri. Šola posveča izobraževanju zaposlenih veliko sredstev in izobraževanje umešča v eno vodilnih prioritiet razvoja organizacije.
7. Katere so bistvene prednosti neprestanega strokovnega izobraževanja in izpopolnjevanja zaposlenih?
Učiteljski poklic naj bi bil poklic za katerega bi se odločali ljudje, ki kažejo stalno nagnjenost k vseživljenjskemu učenju. Večino kadra, zaposlenega na naši šoli, takšne lastnosti ima. Ker pa organizacija rabi »rdečo nit« je zelo pomembno, da so izobraževanja ciljno usmerjena v doseganje kratkoročnih in tudi dolgoročnih ciljev organizacije.
8. Ali menite, da je zakonsko predpisano minimalno število dni izobraževanj zadostno, da lahko strokovni delavci stopajo v koraku s časom?
Zelo pomembno je stalno sledenje razvoju in novostim na področju posamezne stroke. Število dni, ki je predpisano z ZOFVI, zagotovo temu ne zadošča, se pa pričakuje, da posamezni zaposleni sledi novostim ne samo v okviru predpisanega števila dni. Gre namreč za segment priprav na pouk, načrtovanje dela, dela v študijskih skupinah in strokovnih aktivih, ki pa v teh dnevih niso zajeti.
9. V povezavi s prejšnjim vprašanjem. Kako ocenjujete zmožnost sledenja razvoju tehnike, glede na to, da ste predvsem tehnično usmerjena šola?
Veliko možnosti ponuja stalna navezava na gospodarstvo in oblike sodelovanja z realnim okoljem, v katerem se dijaki po končanem izobraževanju izpopolnjujejo.
10. Učinkovito spremljanje posameznega zaposlenega je potrebno voditi. Kako vi vidite rešitev, da bi se to lahko izvajalo bolj učinkovito?
Ko delamo kadrovske načrte, LDN za posamezno leto, karierno načrtovanje za posamezne zaposlene, se vedno poglobimo tudi v področje strokovnega izpopolnjevanja. Z leti načrtnega vodenja se slika oblikuje, vendar je zato potrebnega veliko truda in 'papirologije'. Nekateri zaposleni si tudi sami vodijo evidence strokovnih izobraževanj, kar je organizaciji v veliko pomoč, saj se obe evidenci lahko dobro dopolnjujeta.
11. Imate v mislih kakšno konkretno rešitev, kako bi spremljali udeležbo zaposlenih na strokovnih izobraževanjih in izpopolnjevanjih?
Bolj kot spremljanje, je problem evidentiranje in izdajanje potrdil. Veliko strokovnih usposabljanj se zdaj izgubi oz. pozabi, z načrtnim vodenjem s pomočjo spletne aplikacije pa bi bilo lažje, preglednejše in trajnejše.
12. Kaj je po vašem mnenju ključni dejavnik, da se zaposleni želijo izobraževati?
Narava našega dela je taka, da je razvoj tako hiter, da zaposleni, ki se ne izobražuje, svojega dela ne more kakovostno opravljati. Ko začne zaostajati, se začnejo tudi druge težave. Običajno te zaznamo, vzrok pa spregledamo. Le dobro usposobljen delavec lahko sledi zahtevam poklica.

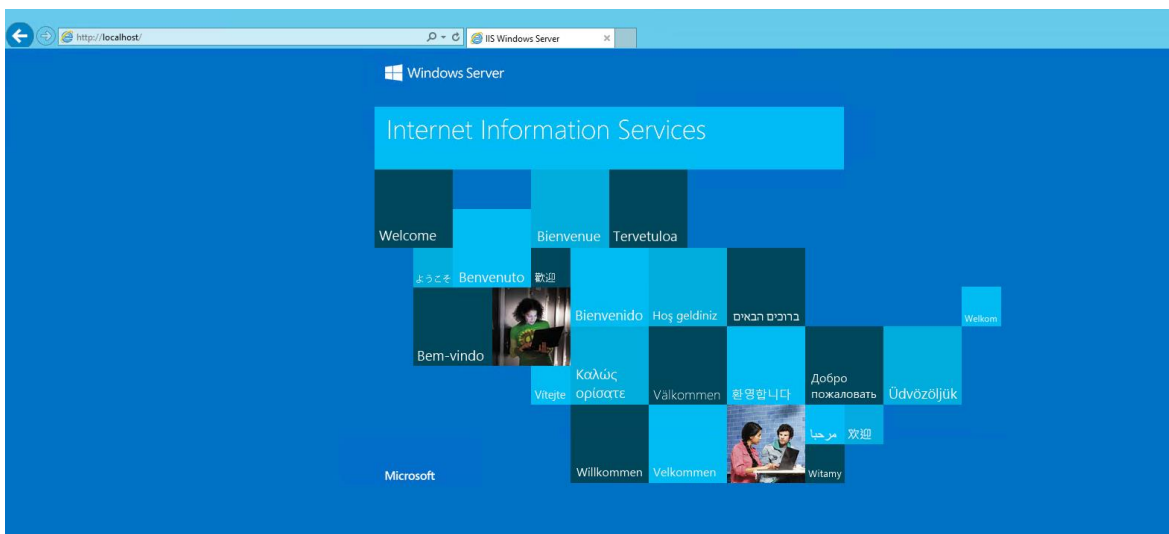
13. Ali menite, da je strokovno izobraževanje in izpopolnjevanje zaposlenih povezano tudi s pridobivanjem točk za napredovanje?
Ko so bila napredovanja »zamrznjena« se na naši šoli želja po izobraževanjih ni zmanjšala, kar potrjuje moje navedbe iz prejšnje točke. Je pa res, da so se mnogi izobraževali na področjih, ki niso prinašala točk. Nekateri so pa bolj pragmatični (bili) – ko so nabrali dovolj točk, je bila za njih stvar praktično zaključena – tu ne gre toliko za izobraževanja kot za druga strokovna dela.
14. Menite, da bi boljši nadzor nad izobraževanji vodil v boljše rezultate ali bi se zgodilo ravno nasprotno?
Uvedbo spletne aplikacije ne bi poimenovala nadzor, saj že danes spremljamo koliko se kdo izobražuje, le da je vse skupaj zamudnejše in nepreglednejše.
15. V kolikor bi potek izobraževanj vodili v elektronski obliki, obstajajo kakšni zadržki povezani z zakonodajo?
Kako se vodijo posamezne evidence, ni predpisano.
16. Kakšna je vaša vizija vodenja izobraževanj zaposlenih v elektronski obliki?
Vodenje izobraževanj v elektronski obliki podpiram.
17. Kaj bi pridobili z elektronsko obliko vodenja izobraževanj?
Z uvedbo vsake novosti je potrebno pogledati kaj pridobijo vsi, ki so vanjo vključeni. Šola bi si omogočila preglednejše delo pri vodenju izobraževanj za zaposlene in lažje načrtovanje, ker pa bi imel v to aplikacijo (upam) vpogled vsak zaposleni za svoj del, bi to pomenilo tudi lažje spremljanje in načrtovanje za vsakega posameznika.
18. V kolikor bi takšna aplikacija podprla poslovni proces vodenja izobraževanj in ga s tem izboljšala, bi podprli takšen projekt?
Poenostavitev dela je vedno dobrodošla, zato takšno aplikacijo podpiram.

Priloga 2: Aktivacija podpora IIS

Slika 1: Namestitev okolja .NET Framework 3.5

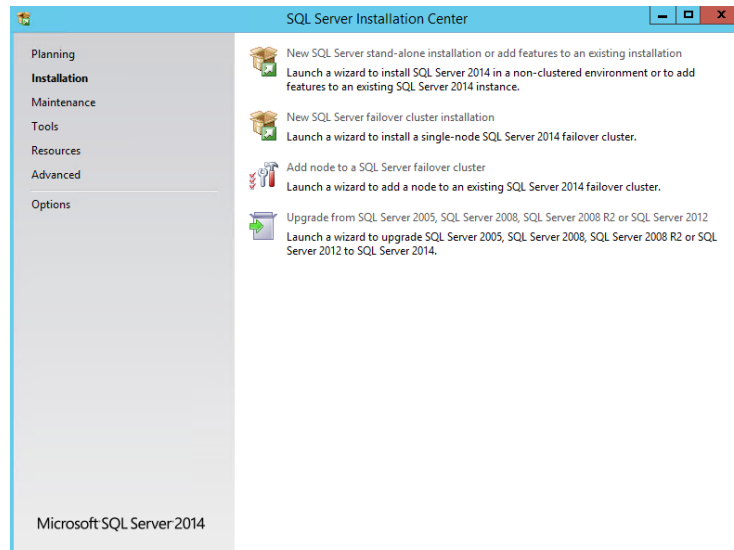


Slika 2: Prikaz lokalnega gostitelja

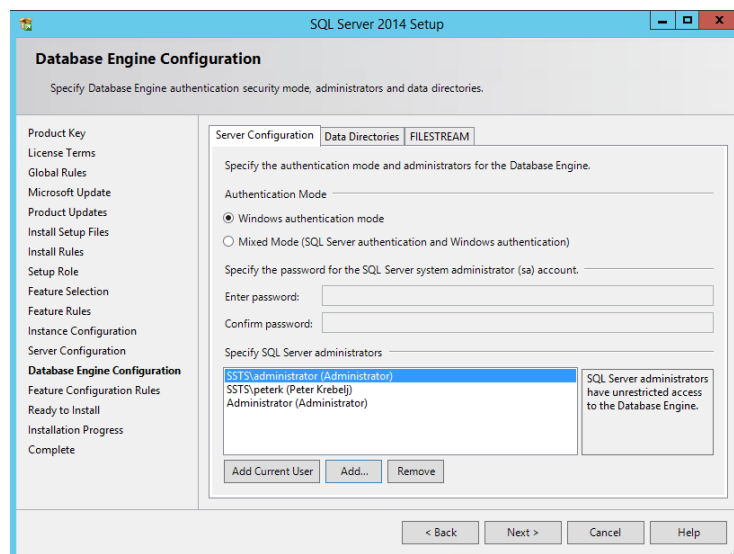


Priloga 3: Podatkovna baza SQL

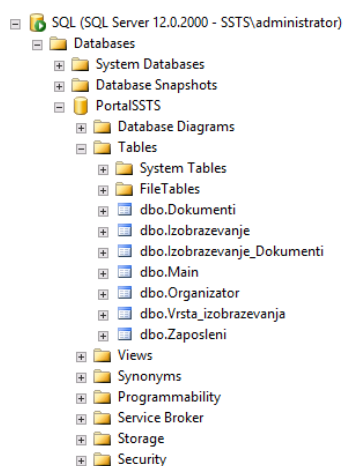
Slika 1: Začetek namestitve strežnika SQL



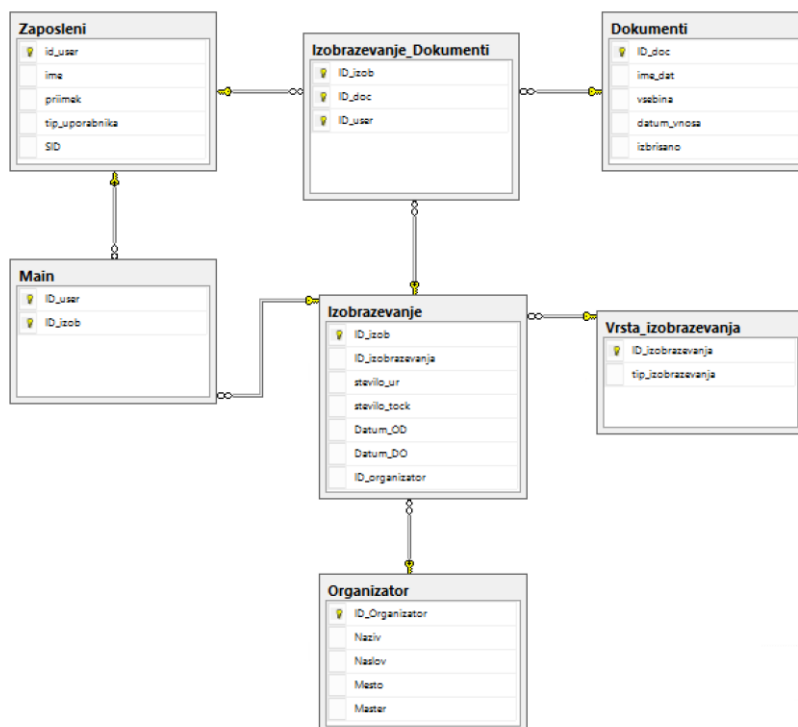
Slika 2: Strežnik SQL konfiguracija pravic



Slika 3: Prikaz strukture podatkovne baze PortalSSTS

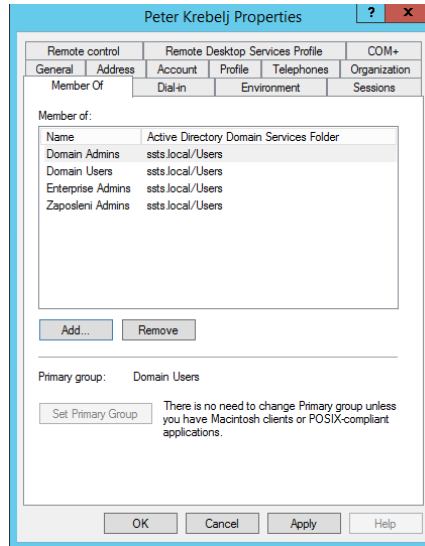


Slika 4: Prikaz končne shreme baze SQL

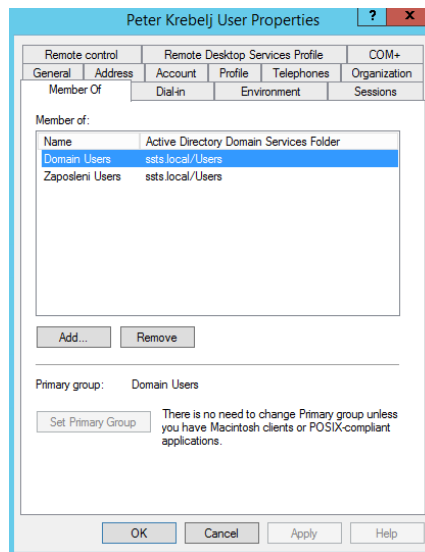


Priloga 4: Konfiguracija Aktivnega imenika

Slika 1: Primer uporabnika s pravicami administratorja

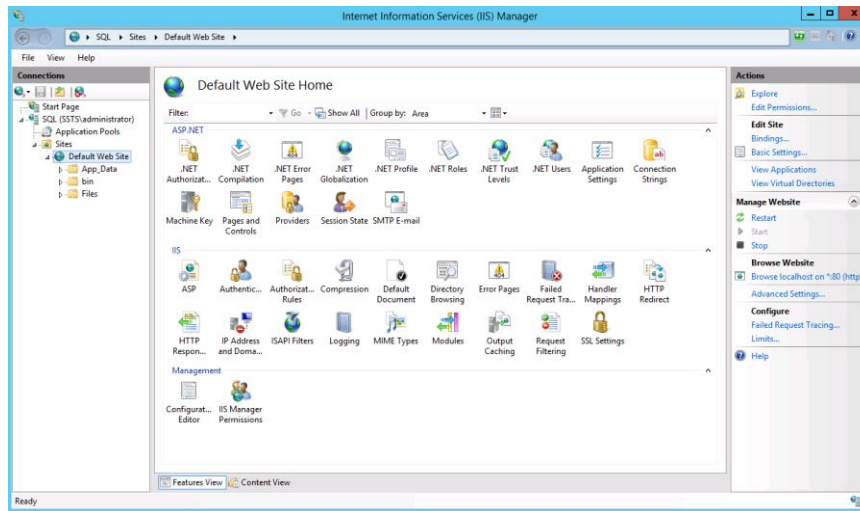


Slika 2: Prikaz testnega uporabnika z običajnimi pravicami

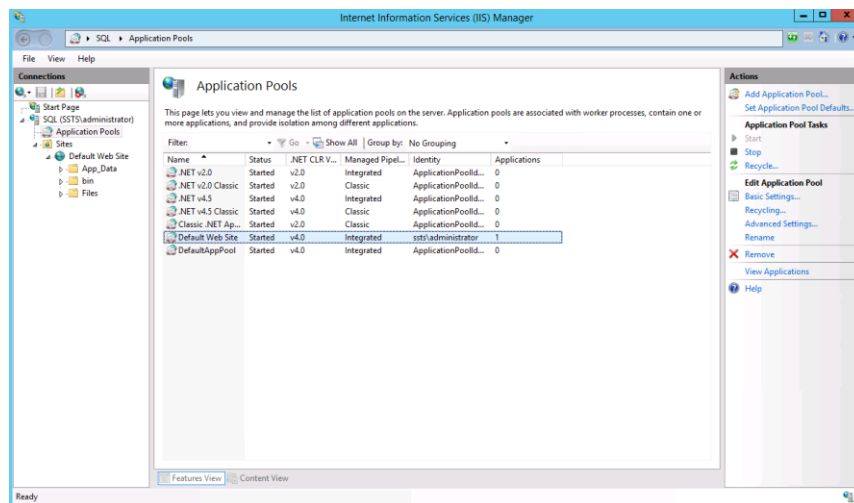


Priloga 5: Konfiguracija IIS

Slika 1: Osnovni pogled na IIS



Slika 2: Nastavitev dostopa spletne strani



Priloga 6: Izdelava spletne aplikacije

Slika 1: Prijava uporabnika

Upor. ime.:

Geslo:

Slika 2: Določanje prikaza glede na pravice

```
try
{
    if (Session["permission"] != null)
    {
        Response.Clear();
        Response.Write("Prijavljen: <b>" + Session["loggeduser"].ToString().ToUpper() + "</b>");
        Response.Write(" ");

        if (Session["isadmin"].ToString().Contains("true"))
        {
            Response.Write("Administrator");
        }
        else
        {
            Response.Write("Uporabnik");
        }
        Response.Write(" - <a href='\"Login.aspx\"'>003AVA</a>");
        Response.Write("<div style='clear:both;height:10px'></div>");
        if (Session["id_user"] != null)
        {
            Response.Write("<div class='menu' onclick='\"top.location.href='\"Default.aspx'\">Moja izobraževanja</div>");
            Response.Write("<div class='menu' onclick='\"top.location.href='\"Izobrazevanje.aspx'\">Izobrazevanjec</div>");

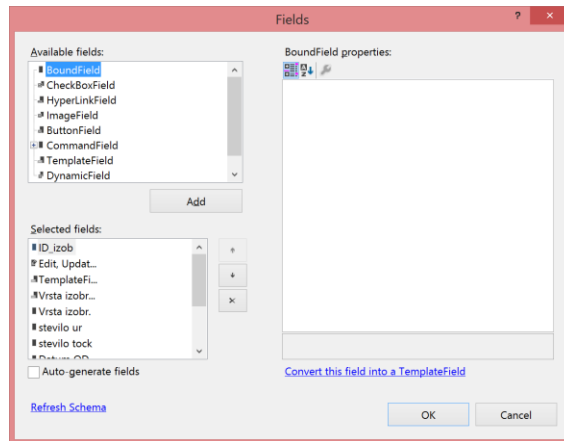
            if (Session["isadmin"].ToString().Contains("true"))
            {
                Response.Write("<div class='menu' onclick='\"top.location.href='\"Zaposleni.aspx'\">Zaposleni</div>");
                Response.Write("<div class='menu' onclick='\"top.location.href='\"VrstaIzobrazevanja.aspx'\">Vrsta izobrazevanj</div>");
                Response.Write("<div class='menu' onclick='\"top.location.href='\"Organizator.aspx'\">Organizator</div>");
                Response.Write("<div style='clear:both'></div>");
            }
        }
        Response.Flush();
    }
}
```

Slika 3: Uporaba GridView

The screenshot shows a web browser window with the following tabs: MasterPage.master, Izobrazevanje.aspx, Default.aspx, Login.aspx.cs, and Web.config. The page title is "Portal za zaposlene" and the main heading is "Vodenje izobrazevanj". Below the heading is a "Dodaj" button. The main content is a GridView table with the following columns: ID_izob, Vrsta Izobrazevanja, Vrsta Izobr., stevilo ur, stevilo tock, Datum OD, Datum DO, Organizator, and Orga. The table contains several rows of data, each with a "Databound" label and "Edit" and "Delete" buttons. The footer of the table shows "SqlDataSource" and "SqlDataSource3".

ID_izob	Vrsta Izobrazevanja	Vrsta Izobr.	stevilo ur	stevilo tock	Datum OD	Datum DO	Organizator	Orga
Databound	Edit	Delete	Databound	Databound	Databound	Databound	Databound	Data
Databound	Edit	Delete	Databound	Databound	Databound	Databound	Databound	Data
Databound	Edit	Delete	Databound	Databound	Databound	Databound	Databound	Data
Databound	Edit	Delete	Databound	Databound	Databound	Databound	Databound	Data
Databound	Edit	Delete	Databound	Databound	Databound	Databound	Databound	Data
SqlDataSource			SqlDataSource3					

Slika 4: Izbira polj za prikaz podatkov



Slika 5: Prikaz kode za zapis izobraževanja v podatkovno bazo

```
if (err == false)
{
    try
    {
        new DataSetTableAdapters.IzobrazevanjeTableAdapter().Insert(Convert.ToInt32(ddlVrstIzobrazevanja.SelectedValue),
            Convert.ToInt32(txtUre.Text), Convert.ToDouble(txtTocke.Text), DateTime.Parse(txtDatumOd.Text),
            DateTime.Parse(txtDatumDo.Text), Convert.ToInt32(ddlOrganizator.Text));
        lbl.ForeColor = Color.Green;
        lbl.Text = "Novo izobraževanje uspešno dodano v bazo";
        ClearAllField();
    }
    catch (Exception)
    {
        lbl.ForeColor = Color.Red;
        lbl.Text = "Prišlo je do napake pri vnosu v bazo, poskusite ponovno.";
    }
}
else
{
    divDodajIzobrazevanje.Style.Add("display", "block");
}
```

Slika 6: Dodajanje dokumenta v podatkovno bazo

```
protected void btnSubmit_Click(object sender, EventArgs e)
{
    try
    {
        string guid = Guid.NewGuid().ToString().Substring(0, 3);
        fu.SaveAs(ConfigurationManager.AppSettings["DokumentiStorage"] + guid + "_" + fu.FileName);
        int returnID = 0;

        try
        {
            returnID = Convert.ToInt32(new DataSetTableAdapters.DokumentiTableAdapter().InsertQuery(guid + "_" + fu.FileName, true, DateTime.Now, false));
        }
        catch (Exception)
        {
        }

        if (returnID != 0)
        {
            new DataSetTableAdapters.Izobrazevanje_DokumentiTableAdapter().Insert(Convert.ToInt32(Request.QueryString["ID_izob"]), returnID,
                Convert.ToInt32(Request.QueryString["ID_user"]));
            Response.Redirect("Default.aspx");
        }
    }
    catch (Exception)
    {
        lbl.Text = "Prišlo je do napake, poskusite ponovno.";
    }
}
```

Priloga 7: Skripta za kreiranje baze SQL

```
USE [master]
GO
CREATE DATABASE [SSTS_PORTAL]
    CONTAINMENT = NONE
    ON PRIMARY
    ( NAME = N'SSTS_PORTAL', FILENAME = N'X:\SSTS\Files\SSTS_PORTAL.mdf' , SIZE = 5120KB
    , MAXSIZE = UNLIMITED, FILEGROWTH = 1024KB )
    LOG ON
    ( NAME = N'SSTS_PORTAL_log', FILENAME = N'X:\SSTS\Logs\SSTS_PORTAL_log.ldf' , SIZE =
    20480KB , MAXSIZE = 2048GB , FILEGROWTH = 10%)
GO
ALTER DATABASE [SSTS_PORTAL] SET COMPATIBILITY_LEVEL = 120
GO
IF (1 = FULLTEXTSERVICEPROPERTY('IsFullTextInstalled'))
begin
EXEC [SSTS_PORTAL].[dbo].[sp_fulltext_database] @action = 'enable'
end
GO
ALTER DATABASE [SSTS_PORTAL] SET ANSI_NULL_DEFAULT OFF
GO
ALTER DATABASE [SSTS_PORTAL] SET ANSI_NULLS OFF
GO
ALTER DATABASE [SSTS_PORTAL] SET ANSI_PADDING OFF
GO
ALTER DATABASE [SSTS_PORTAL] SET ANSI_WARNINGS OFF
GO
ALTER DATABASE [SSTS_PORTAL] SET ARITHABORT OFF
GO
ALTER DATABASE [SSTS_PORTAL] SET AUTO_CLOSE OFF
GO
ALTER DATABASE [SSTS_PORTAL] SET AUTO_SHRINK OFF
GO
ALTER DATABASE [SSTS_PORTAL] SET AUTO_UPDATE_STATISTICS ON
GO
ALTER DATABASE [SSTS_PORTAL] SET CURSOR_CLOSE_ON_COMMIT OFF
GO
ALTER DATABASE [SSTS_PORTAL] SET CURSOR_DEFAULT GLOBAL
GO
ALTER DATABASE [SSTS_PORTAL] SET CONCAT_NULL_YIELDS_NULL OFF
GO
ALTER DATABASE [SSTS_PORTAL] SET NUMERIC_ROUNDABORT OFF
GO
ALTER DATABASE [SSTS_PORTAL] SET QUOTED_IDENTIFIER OFF
GO
ALTER DATABASE [SSTS_PORTAL] SET RECURSIVE_TRIGGERS OFF
GO
ALTER DATABASE [SSTS_PORTAL] SET DISABLE_BROKER
GO
ALTER DATABASE [SSTS_PORTAL] SET AUTO_UPDATE_STATISTICS_ASYNC OFF
GO
ALTER DATABASE [SSTS_PORTAL] SET DATE_CORRELATION_OPTIMIZATION OFF
GO
ALTER DATABASE [SSTS_PORTAL] SET TRUSTWORTHY OFF
GO
ALTER DATABASE [SSTS_PORTAL] SET ALLOW_SNAPSHOT_ISOLATION OFF
GO
ALTER DATABASE [SSTS_PORTAL] SET PARAMETERIZATION SIMPLE
GO
ALTER DATABASE [SSTS_PORTAL] SET READ_COMMITTED_SNAPSHOT OFF
GO
ALTER DATABASE [SSTS_PORTAL] SET HONOR_BROKER_PRIORITY OFF
```

```

GO
ALTER DATABASE [SSTS_PORTAL] SET RECOVERY FULL
GO
ALTER DATABASE [SSTS_PORTAL] SET MULTI_USER
GO
ALTER DATABASE [SSTS_PORTAL] SET PAGE_VERIFY CHECKSUM
GO
ALTER DATABASE [SSTS_PORTAL] SET DB_CHAINING OFF
GO
ALTER DATABASE [SSTS_PORTAL] SET FILESTREAM( NON_TRANSACTED_ACCESS = OFF )
GO
ALTER DATABASE [SSTS_PORTAL] SET TARGET_RECOVERY_TIME = 0 SECONDS
GO
ALTER DATABASE [SSTS_PORTAL] SET DELAYED_DURABILITY = DISABLED
GO
EXEC sys.sp_db_vardecimal_storage_format N'SSTS_PORTAL', N'ON'
GO
USE [SSTS_PORTAL]
GO

CREATE USER [testiranje] FOR LOGIN [testiranje] WITH DEFAULT_SCHEMA=[dbo]
GO

CREATE USER [SSTS\Domain admins] FOR LOGIN [SSTS\Domain admins] WITH
DEFAULT_SCHEMA=[dbo]
GO

CREATE USER [SSTS\Administrator] FOR LOGIN [SSTS\Administrator] WITH
DEFAULT_SCHEMA=[dbo]
GO

CREATE USER [SSTS\peterk] FOR LOGIN [SSTS\peterk] WITH DEFAULT_SCHEMA=[dbo]
GO
ALTER ROLE [db_owner] ADD MEMBER [testiranje]
GO
ALTER ROLE [db_datareader] ADD MEMBER [testiranje]
GO
ALTER ROLE [db_datawriter] ADD MEMBER [testiranje]
GO
ALTER ROLE [db_owner] ADD MEMBER [SSTS\Domain admins]
GO
ALTER ROLE [db_owner] ADD MEMBER [SSTS\Administrator]
GO
ALTER ROLE [db_owner] ADD MEMBER [SSTS\peterk]
GO
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
SET ANSI_PADDING ON
GO
CREATE TABLE [dbo].[Dokumenti](
    [ID_doc] [int] IDENTITY(1,1) NOT NULL,
    [ime_dat] [varchar](50) NULL,
    [vsebina] [bit] NULL,
    [datum_vnosa] [datetime] NULL,
    [izbrisanost] [bit] NULL,
    CONSTRAINT [PK_Dokumenti] PRIMARY KEY CLUSTERED
(
    [ID_doc] ASC
)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF, IGNORE_DUP_KEY = OFF,
ALLOW_ROW_LOCKS = ON, ALLOW_PAGE_LOCKS = ON) ON [PRIMARY]
) ON [PRIMARY]

```



```

GO
SET ANSI_PADDING OFF
GO
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE TABLE [dbo].[Izobrazevanje](
    [ID_izob] [int] IDENTITY(1,1) NOT NULL,
    [ID_izobrazevanja] [int] NULL,
    [stevilo_ur] [int] NULL,
    [stevilo_tock] [float] NULL,
    [Datum_OD] [datetime] NULL,
    [Datum_DO] [datetime] NULL,
    [ID_organizator] [int] NULL,
    CONSTRAINT [PK_Izobrazevanje] PRIMARY KEY CLUSTERED
(
    [ID_izob] ASC
)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF, IGNORE_DUP_KEY = OFF,
ALLOW_ROW_LOCKS = ON, ALLOW_PAGE_LOCKS = ON) ON [PRIMARY]
) ON [PRIMARY]

GO
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE TABLE [dbo].[Izobrazevanje_Dokumenti](
    [ID_izob] [int] NOT NULL,
    [ID_doc] [int] NOT NULL,
    [ID_user] [int] NOT NULL,
    CONSTRAINT [PK_Izobrazevanje_Dokumenti] PRIMARY KEY CLUSTERED
(
    [ID_izob] ASC,
    [ID_doc] ASC,
    [ID_user] ASC
)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF, IGNORE_DUP_KEY = OFF,
ALLOW_ROW_LOCKS = ON, ALLOW_PAGE_LOCKS = ON) ON [PRIMARY]
) ON [PRIMARY]

GO
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE TABLE [dbo].[Main](
    [ID_user] [int] NOT NULL,
    [ID_izob] [int] NOT NULL,
    CONSTRAINT [PK_Main] PRIMARY KEY CLUSTERED
(
    [ID_user] ASC,
    [ID_izob] ASC
)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF, IGNORE_DUP_KEY = OFF,
ALLOW_ROW_LOCKS = ON, ALLOW_PAGE_LOCKS = ON) ON [PRIMARY]
) ON [PRIMARY]

GO
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
SET ANSI_PADDING ON

```

```

GO
CREATE TABLE [dbo].[Organizator](
    [ID_Organizator] [int] IDENTITY(1,1) NOT NULL,
    [Naziv] [varchar](50) NULL,
    [Naslov] [varchar](255) NULL,
    [Mesto] [varchar](50) NULL,
    [Master] [bit] NULL,
    CONSTRAINT [PK_Organizator] PRIMARY KEY CLUSTERED
(
    [ID_Organizator] ASC
)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF, IGNORE_DUP_KEY = OFF,
ALLOW_ROW_LOCKS = ON, ALLOW_PAGE_LOCKS = ON) ON [PRIMARY]
) ON [PRIMARY]

GO
SET ANSI_PADDING OFF
GO
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
SET ANSI_PADDING ON
GO
CREATE TABLE [dbo].[Vrsta_izobrazevanja](
    [ID_izobrazevanja] [int] IDENTITY(1,1) NOT NULL,
    [tip_izobrazevanja] [varchar](50) NULL,
    CONSTRAINT [PK_Vrsta_izobrazevanja] PRIMARY KEY CLUSTERED
(
    [ID_izobrazevanja] ASC
)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF, IGNORE_DUP_KEY = OFF,
ALLOW_ROW_LOCKS = ON, ALLOW_PAGE_LOCKS = ON) ON [PRIMARY]
) ON [PRIMARY]

GO
SET ANSI_PADDING OFF
GO
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
SET ANSI_PADDING ON
GO
CREATE TABLE [dbo].[Zaposleni](
    [id_user] [int] IDENTITY(1,1) NOT NULL,
    [ime] [varchar](50) NULL,
    [priimek] [varchar](50) NULL,
    [tip_uporabnika] [varchar](50) NULL,
    [SID] [varchar](50) NULL,
    CONSTRAINT [PK_Zaposleni] PRIMARY KEY CLUSTERED
(
    [id_user] ASC
)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF, IGNORE_DUP_KEY = OFF,
ALLOW_ROW_LOCKS = ON, ALLOW_PAGE_LOCKS = ON) ON [PRIMARY]
) ON [PRIMARY]

GO
SET ANSI_PADDING OFF
GO
ALTER TABLE [dbo].[Izobrazevanje] WITH CHECK ADD CONSTRAINT
[FK_Izobrazevanje_Organizator] FOREIGN KEY([ID_organizator])
REFERENCES [dbo].[Organizator] ([ID_Organizator])
GO
ALTER TABLE [dbo].[Izobrazevanje] CHECK CONSTRAINT [FK_Izobrazevanje_Organizator]

```

```

GO
ALTER TABLE [dbo].[Izobrazevanje] WITH CHECK ADD CONSTRAINT
[FK_Izobrazevanje_Vrsta_izobrazevanja] FOREIGN KEY([ID_izobrazevanja])
REFERENCES [dbo].[Vrsta_izobrazevanja] ([ID_izobrazevanja])
GO
ALTER TABLE [dbo].[Izobrazevanje] CHECK CONSTRAINT
[FK_Izobrazevanje_Vrsta_izobrazevanja]
GO
ALTER TABLE [dbo].[Izobrazevanje_Dokumenti] WITH CHECK ADD CONSTRAINT
[FK_Izobrazevanje_Dokumenti_Dokumenti] FOREIGN KEY([ID_doc])
REFERENCES [dbo].[Dokumenti] ([ID_doc])
GO
ALTER TABLE [dbo].[Izobrazevanje_Dokumenti] CHECK CONSTRAINT
[FK_Izobrazevanje_Dokumenti_Dokumenti]
GO
ALTER TABLE [dbo].[Izobrazevanje_Dokumenti] WITH CHECK ADD CONSTRAINT
[FK_Izobrazevanje_Dokumenti_Izobrazevanje] FOREIGN KEY([ID_izob])
REFERENCES [dbo].[Izobrazevanje] ([ID_izob])
GO
ALTER TABLE [dbo].[Izobrazevanje_Dokumenti] CHECK CONSTRAINT
[FK_Izobrazevanje_Dokumenti_Izobrazevanje]
GO
ALTER TABLE [dbo].[Izobrazevanje_Dokumenti] WITH CHECK ADD CONSTRAINT
[FK_Izobrazevanje_Dokumenti_Zaposleni] FOREIGN KEY([ID_user])
REFERENCES [dbo].[Zaposleni] ([id_user])
GO
ALTER TABLE [dbo].[Izobrazevanje_Dokumenti] CHECK CONSTRAINT
[FK_Izobrazevanje_Dokumenti_Zaposleni]
GO
ALTER TABLE [dbo].[Main] WITH CHECK ADD CONSTRAINT [FK_Main_Izobrazevanje] FOREIGN
KEY([ID_izob])
REFERENCES [dbo].[Izobrazevanje] ([ID_izob])
GO
ALTER TABLE [dbo].[Main] CHECK CONSTRAINT [FK_Main_Izobrazevanje]
GO
ALTER TABLE [dbo].[Main] WITH CHECK ADD CONSTRAINT [FK_Main_Zaposleni] FOREIGN
KEY([ID_user])
REFERENCES [dbo].[Zaposleni] ([id_user])
GO
ALTER TABLE [dbo].[Main] CHECK CONSTRAINT [FK_Main_Zaposleni]
GO
USE [master]
GO
ALTER DATABASE [SSTS_PORTAL] SET READ_WRITE
GO

```

Priloga 8: Spletna aplikacija

Zaradi večjega obsega celotne kode, ki bi zavzela preveliko število strani za prikaz celotne kode, je končen projekt priložen v elektronski obliki na CD/DVD mediju.

Priloga 9: Seznam uporabljenih kratic

AD – (angl.) Active Directory; aktivni imenik

ALE – (angl.) Annualized Loss Expectancy; pričakovana vrednost izgube podatkov

API – (angl.) Application Program Interface; programski vmesnik

AS-IS – informacijski sistem, kot je trenutno zasnovan

BPM – (angl.) Business Process Management; upravljanje poslovnih procesov

CSS – (angl.) Cascading Style Sheet; prekrivni slogi

DHCP – (angl.) Dynamic Host Configuration Protocol; protokol za samodejno dodeljevanje naslova IP

DOS – (angl.) Disk Operating System; diskovni operacijski sistem

DNS – (angl.) Domain Name Server; strežnik domenskih imen

EMŠO – enotna matična številka občana

E-R – (angl.) entity relationship model; model entitet in povezav

ERP – (angl.) Enterprise Resource Planning; celovito povezana programska rešitev

FTP – (angl.) File Transfer Protocol; protokol za prenos datotek

GB – (angl.) Gigabyte; enota informacije, ki vsebuje 1024 MB

GHz – (angl.) Gigahertz; izhaja iz osnovne enote Hz (hertz), ki ponazarja osnovno enoto merjenja hitrosti delovanja mikro procesorja

HTTP – (angl.) Hypertext Transfer Protocol; protokol za izmenjavo spletnih vsebin

HTTPS – (angl.) Hypertext Transfer Protocol Secured; varovani protokol za izmenjavo spletnih vsebin

ID – (angl.) identifier; identifikator

IIS – (angl.) Internet Information Services; rešitev podjetja Microsoft za podporo izvajanju spletnih aplikacij na strežniku

IP – (angl.) Internet Protocol; internetni protokol

LAN – (angl.) Local Area Network; krajevno omrežje

LDAP – (angl.) Lightweight Directory Access Protocol; internetni protokol za dostop do imenikov na osnovi arhitekture odjemalec - strežnik

LDN – Letni delovni načrt

LUN – Letni učni načrt

LUR – Ljubljanska Urbana Regija

MSDN – (angl.) Microsoft Developer Network; storitev podjetja Microsoft za dostop do programske opreme

NAS – (angl.) Network Attached Storage; omrežno diskovno polje

OS – (angl.) operating system; operacijski sistem

PDC – (angl.) Primary DNS Server, korenski domenski strežnik

PDF – (angl.) Portable Document Format; format datoteke, ki je prenosljiv med sistemi in neodvisen od platforme

PTI – Poklicno Tehnično Izobraževanje

SA – (angl.) School Agreement; pogodba z Microsoft o posebni ceni za uporabo storitev

SAP – (angl.) System Applications Products

SDC – (angl.) Secondary DNS Server; redundantni domenski strežnik

SDLC – (angl.) System Development Life Cycle; življenjski cikel razvoja aplikacije

SISP – (angl.) Strategic Information System Planning; proces identifikacije računalniško prilagojenih aplikacij

SMTP – (angl.) Simple Mail Transfer Protocol; internetni protokol za prenos spletne pošte

SPI – Strokovno Poklicno Izobraževanje

SSI – Srednje Strokovno Izobraževanje

SŠTS – Srednja Šola Tehniških Strok

STP – (angl.) Spanning Tree Protocol; protokol za preprečevanje zank v omrežju

STP – (angl.) Shielded Twisted Pair; oklopljena sukana parica

SQL – (angl.) Structured Query Language; strukturiran povpraševalni jezik za delo s podatkovnimi bazami

TCO – (angl.) Total Cost of Ownership; celotni stroški lastništva

TO-BE – Informacijski sistem, kot želimo, da bi bil zasnovan

UML – (angl.) Unified Modeling Language; poenoten jezik modeliranja

UPS – (angl.) Uninterruptible Power Supply; brezprekinitveno napajanje

VLAN – (angl.) Virtual Local Area Network; virtualno lokalno omrežje, ki nam preko istega medija omogoča sočasni prenos različnih kanalov

UI – (angl.) User Interface; uporabniški vmesnik

ZOFVI – Zakon o organizaciji in financiranju vzgoje in izobraževanja

XML – (angl.) Extensible Markup Language; format podatkov za izmenjavo strukturiranih dokumentov