# Primjena CRM-a u funkciji praćenja studenata od strane korisnika informacijskog sustava visokog učilišta

**Anić, Ante**

**Undergraduate thesis / Završni rad**

**2016**

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

*Permanent link / Trajna poveznica:* https://urn.nsk.hr/urn:nbn:hr:200:703098

*Rights / Prava:* In copyright / Zaštićeno autorskim pravom.

*Download date / Datum preuzimanja:* **2024-04-19**

*Repository / Repozitorij:*

Faculty of Electrical Engineering, Computer Science and Information Technology Osijek

**SVEUČILIŠTE JOSIPA JURJA STROSSMAYERA U OSIJEKU**

**FAKULTET ELEKTROTEHNIKE, RAČUNARSTVA I INFORMACIJSKIH TEHNOLOGIJA**

**Sveučilišni studij**

# PRIMJENA CRM-A U FUNKCIJI PRAĆENJA STUDENATA OD STRANE KORISNIKA INFORMACIJSKOG SUSTAVA VISOKOG UČILIŠTA

**Završni rad**

**Ante Anić**

**Osijek, 2016**

Sveučilište Josipa Jurja Strossmayera u Osijeku

ETFOS ELEKTROTEHNIČKI FAKULTET OSIJEK

**Obrazac Z1P - Obrazac za ocjenu završnog rada na preddiplomskom studiju**

**Osijek,**

**Odboru za završne i diplomske ispite**

# Prijedlog ocjene završnog rada

| | |
|---|---|
| **Ime i prezime studenta:** | |
| **Studij, smjer:** | |
| **Mat. br. studenta, godina upisa:** | |
| **Mentor:** | |
| **Sumentor:** | |
| **Naslov završnog rada:** | |
| **Primarna znanstvena grana rada:** | |
| **Sekundarna znanstvena grana (ili polje) rada:** | |
| **Predložena ocjena završnog rada:** | |
| **Kratko obrazloženje ocjene prema Kriterijima za ocjenjivanje završnih i diplomskih radova:** | Primjena znanja stečenih na fakultetu:<br>Postignuti rezultati u odnosu na složenost zadatka:<br>Jasnoća pismenog izražavanja:<br>Razina samostalnosti: |

| Potpis sumentora: | Potpis mentora: |
|---|---|

Dostaviti:

1. Studentska služba

Potpis predsjednika Odbora: _____

Dostaviti:

1. Studentska služba

Sveučilište Josipa Jurja Strossmayera u Osijeku

ELEKTROTEHNIČKI FAKULTET OSIJEK

# IZJAVA O ORIGINALNOSTI RADA

**Osijek,**

| | |
|---|---|
| **Ime i prezime studenta:** | Ante Anić |
| **Studij** : | |
| **Mat. br. studenta, godina upisa:** | |

Ovom izjavom izjavljujem da je rad pod nazivom**:**

izrađen pod vodstvom mentora

i sumentora

Moj vlastiti rad i prema mom najboljem znanju ne sadrži prethodno objavljene ili neobjavljene pisane materijale drugih osoba, osim onih koji su izričito priznati navođenjem literature i drugih izvora informacija.
Izjavljujem da je intelektualni sadržaj navedenog rada proizvod mog vlastitog rada,osim u onom dijelu za koji mi je bila potrebna pomoć mentora, sumentora i drugih osoba, a što je izričito navedeno u radu.

Potpis studenta:

# Sadržaj


1. UVOD ..... 3
   1.1. Zadatak Završnog rada ..... 3
2. CATEGORY MANAGMENT U POSLOVNOM OKRUŽENJU ..... 4
   2.1. Pojam i evolucija CRM-a ..... 5
   2.2. CRM poslovna strategija i njezina implementacija ..... 5
   2.3. Primjena u Republici Hrvatskoj ..... 6
3. KONCEPT UPRAVLJANJA ODNOSIMA S KORISNICIMA USLUGA ..... 8
4. INFORMACIJSKI SUSTAVI I BAZE PODATAKA ..... 9
5. DIZAJNIRANJE I IZRADA BAZE PODATAKA ..... 12
   5.1. Dizajniranje baze podataka ..... 12
   5.2. Shema baze podataka ..... 13
6. IMPLEMENTIRANJE CRM-A NA PRIMJERU FERIT OSIJEK ..... 15
   6.1. Uspostavljanje veze ..... 15
   6.2. Zapisivanje podataka u bazu podataka ..... 16
   6.3. Prikaz rezultata upita ..... 22
      6.3.1. Upiti nad bazom podataka ..... 22
      6.3.2. Prikaz studenata ..... 24
   6.4. Rezultati upita ..... 26
7. ZAKLJUČAK ..... 29

LITERATURA ..... 31

SAŽETAK ..... 33

PRILOZI ..... 36

# 1. UVOD

Intenzivnim razvojem informacijske tehnologije, povećavanjem mogućnosti skupljanja, skladištenja i upotrebe podataka, otvorila se mogućnost da informacijska tehnologija i upotreba baza podataka koje sadrže sve podatke o klijentima, ponovno poslovanju vrate osobnu notu i personaliziran pristup klijentu. Upravljanje odnosom s klijentima ili Category Relationship Managment se kao sveobuhvatni poslovni koncept s pripadajućom softverskom podrškom počinje intenzivnije spominjati i implementirati u vodećim tvrtkama krajem devedesetih godina 20.stoljeća, kako bi u ovom desetljeću postao nezaobilazna stavka za svaku ozbiljniju suvremenu tvrtku ili organizaciju koja želi potpuno orijentirati i prilagoditi svoje poslovanje klijentima te njihovim željama i potrebama. Radom se želi ukazati na mogućnosti primjene CRM-a kao koncepta rada u visokoobrazovanim ustanovama u kojima su studenti kao klijenti kojima se svakodnevno nude određena znanja te ih se osposobljava za tržište rada. Svrha korištenja ovakvog sustava je poboljšanje rada visoko obrazovanih institucija te personaliziranim pristupom svakom studentu osigurati bolju kvalitetu studiranja. U drugom će poglavlju biti opisan pojam CRM, bit će govora o CRM poslovnoj strategiji i njezinoj implementaciji te primjeni u Republici Hrvatskoj. U četvrtom će poglavlju biti objašnjeni pojmovi informacijski sustavi te baze podataka kao i primjena informacijskih sustava u sustavu visokih učilišta Republike Hrvatske. Nakon toga u petom poglavlju predstavljena će biti shema baze podataka kroz dva predložena modela baze podataka Fakulteta elektrotehnike, računarstva i informacijskih tehnologija u Osijeku kako bi se lakše prikazao način praćenja prolaznosti studenata po kolegijima te u dodatku tomu ispisat će se upiti na istu bazu kako bi se mogla pratiti prolaznost. Nakon toga predočena će biti rješenja samoga zadatka.

## 1.1. Zadatak Završnog rada

Zadatak je rada izraditi programsku podršku za praćenje prolaznosti studenata po kolegijima dostupnim djelatnicima zaduženima za organiziranje i izvođenje nastave te druge oblike nastavnog procesa, kao i upravljanje, planiranje i organiziranje rada Fakulteta. Primjena *Category Management-a* (CRM) u ovom kontekstu bila bi poboljšati način djelovanja sudionika u nastavnom procesu i obvezama koje se nameću uz optimizaciju prostornih i ljudskih resursa, kao i vremena učinkovitosti odgovora na potrebe koje se javljaju u procesu rada visoko obrazovne ustanove.

# 2. CATEGORY MANAGMENT U POSLOVNOM OKRUŽENJU

Odgovorno poslovno ponašanje (engl. Category Relationship Managment (CRM)) poslovna je strategija koja uključuje selekciju i upravljanje odnosom s klijentima u svrhu optimiranja njihove dugoročne vrijednosti za tvrtku. Primarna je svrha Category Managmenta pružanje integralne i cjelovite slike odnosa korisnika i poduzeća ili organizacije, kao i omogućavanje uspostave odnosa osoblja te organizacije s korisnicima njezinih proizvoda ili usluga putem kvalitetnije podatkovne podrške.

Mjerljive su pogodnosti koje može donijeti CRM efikasniji poslovni procesi usmjereni klijentu, smanjen prebjeg klijenata konkurenciji, povećana prodaja i tržišni udio, smanjenje pritužbi od strane klijenata te veća efikasnost tvrtke.[3]

## 2.1. Pojam i evolucija CRM-a

CRM je novi model poslovanja koji racionalizira i upravlja prodajnim i marketinškim procesima, osigurava automatizaciju procesa isporuke i održavanja usluga te omogućava kvalitetnu podršku korisnicima i upravljanje korisničkim uslugama. [1]

Sredinom devedesetih godina prošloga stoljeća javljaju se sustavi koji predstavljaju svojevrsnu podršku prodaji kao sustavi namijenjeni klijentima, odnosno podržavanju istih, nakon 1996. godine dolazi do spajanja različitih manjih sustava podrške i podsustava u jedan objedinjeni (zajednički) čime dolazi do pojave druge generacije sustava za upravljanje odnosima s klijentima, nakon 2002. godine, pojavljuje se treća generacija poznatija pod nazivom strateški sustav upravljanja odnosima s klijentima. CRM zahtijeva klijentu usmjerenu poslovnu filozofiju i kulturu tvrtke, kao i potporu procesima prodaje, marketinga i usluge klijentima. CRM softverske aplikacije mogu pružiti tvrtki kvalitetno i efikasno upravljanje odnosom s klijentima. Kako bi CRM inicijativa bila uspješna za početak je nužno odrediti odgovarajuću poslovnu strategiju kreiranja dugoročnih, održivih i profitabilnih odnosa s klijentima, potom prilagoditi poslovne procese tvrtke, a po potrebi i prilagoditi organizacijsku kulturu u svrhu stvaranja klijentu potpuno fokusiranog okružja. CRM tehnologija predstavlja kritični faktor poslovnih procesa koji omogućava pretvaranje CRM strategije u konkretne poslovne rezultate, bez kojih CRM strategija ne bi bila provediva na efikasan i isplativ način. [1] ,[13]

## 2.2. CRM poslovna strategija i njezina implementacija

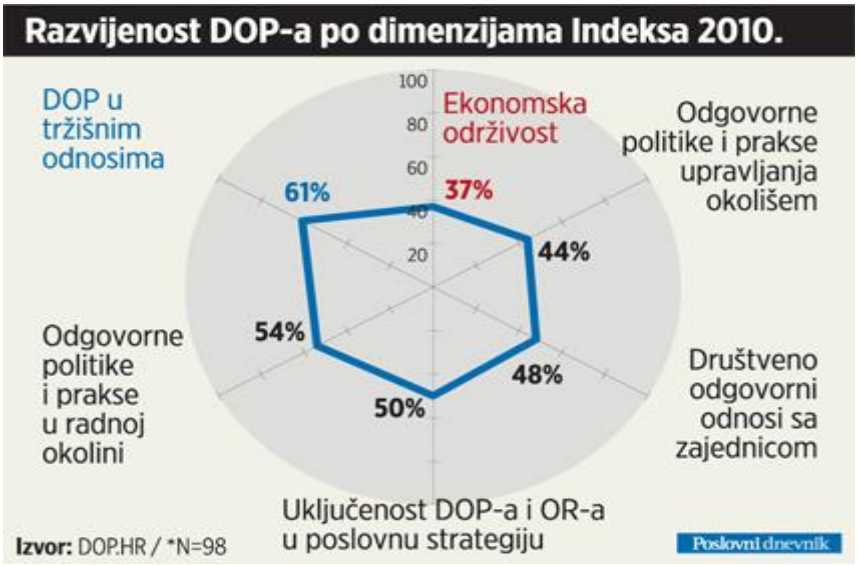
CRM je prije svega poslovna strategija, a tek potom tehnološko rješenje. Kako bi tvrtka mogla uspješno definirati svoju klijentima orijentiranu poslovnu strategiju, potrebno je uzeti u obzir određena pravila koja mora imati u vidu prije formulacije bilo kakve CRM strategije. Pravila za uspješnu CRM strategiju su: svaki posao treba promatrati iz perspektive klijenta, svaki posao treba promatrati kao dio vrijednosnog lanca do krajnjeg korisnika, treba postojati spremnost na promjene i menadžment sponzorstvo. Nužna je spoznaja da je tehnologija samo sredstvo za stvaranje CRM poslovne strategije jer puno je lakše mijenjati tehnologiju nego ljude. Potrebna je integracija tehnologija, poslovnih funkcija i kanala kontakata s klijentima jer je ujedno i preduvjet za konzistentan doživljaj tvrtke od strane klijenata te za prepoznatljiv i vjerodostojan *branding* tvrtke i njezinih proizvoda. Operativna izvrsnost pri toj integraciji, uz poštivanje prethodnih preduvjeta odredit će pobjednike od gubitnika pri implementaciji CRM-a. Promatrajući nastupe na tržištu vodećih svjetskih tvrtki koje demonstriraju zavidan nivo orijentacije prema svojim korisnicima, možemo identificirati 5 osnovnih CRM strategija koje

tvrtka može poduzeti u svrhu postizanja diferencijacije i konkurentnosti na tržištu: ponuda rješenja umjesto proizvoda ili tzv. „ključ u ruke“ primjeri: IBM, GE (organizacija koja se bavi proizvodnjom digitalnih uređaja poput fotoaparata, video kamera, tableta i sl.), ponuda superiorne usluge, konzistentne kroz sve kanale prema klijentu najbolji je primjer lanac trgovina (Tesco), ponuda potpuno klijentu prilagođenih proizvoda i/ili usluga (primjer: Dell, tvrtka za proizvodnju računala i njima bliskih proizvoda ), rješavanje problema klijenta umjesto klasičnog procesa prodaje (primjer: P&G, tvrtka koja se bavi proizvodnjom robnih marki), izgradnja suvremenog kontakt centra koji omogućuje efikasno komuniciranje s klijentima (*web*, samoposluga), te time stvaranje organizacijskog znanja o klijentima kao baze za *cross-sell* i *up-sell* (primjer: Amazon, najpoznatija internetska trgovina). *Cross-sell* predstavlja marketinški izraz koji se odnosi na praksu predlaganja povezanih proizvoda i usluga korisnicima koji su zainteresirani za kupnju istih dok se termin *up-sell* odnosi na tehniku prodaje kojom prodavač potiče korisnika na kupnju skupljih proizvoda i nadogradnji kako bi povećao dobit od prodaje. Za uspješnu implementaciju bilo koje CRM strategije ključna je snažna potpora menadžmenta, koji mora čvrsto stati iza ciljeva implementacije CRM-a i dosta se osobno angažirati na njihovoj realizaciji. CRM sustavi predstavljaju trenutno jedno od najbrže rastućih područja investicija tvrtki u IT rješenja. [1]

## 2.3. Primjena u Republici Hrvatskoj

U Hrvatskoj se društveno odgovorno poslovanje može smatrati nešto novijom poslovnom praksom. Tranzicijski šok, rat kojim je država bila pogođena, ekonomska kriza kao i nedostatak poticaja od strane državne vlasti samo su neki od razloga zbog čega se ova poslovna praksa počinje sporije primjenjivati na našem tržištu. Unatoč navedenim problemima važno je napomenuti da se situacija posljednjih nekoliko godina mijenja na bolje. Hrvatska poduzeća su uočila društveno odgovorno poslovanje kao učinkovit alat za stjecanje konkurentskih prednosti. Društvena odgovornost poduzeća u Hrvatskoj nastala je pod utjecajem mnogobrojnih činitelja, od kojih su neki povijesno utemeljeni i specifični za Hrvatsku, dok su drugi suvremeni i više međunarodnog značaja. Tek se od 1995. godine govori o teritorijalnoj „normalizaciji“. Iznad gospodarskih aspekata tranzicije, kako bi koncept društvene odgovornosti poduzeća mogao funkcionirati i napredovati, mora postojati poticajno okruženje u državi u smislu zakonodavstva i politike. Za učinkovitiji razvoj društvene odgovornosti poduzeća u Hrvatskoj trebalo bi se značajnije usredotočiti na podizanje svijesti o istoj, vršiti edukaciju o izgradnji partnerstva za sve sektore, poboljšavati koordiniranost među poslovnim organizacijama u izradi platforme za

razmjenu najboljih praksi društvene odgovornosti te pomagati gospodarstvu u ugradnji i primjeni strategija društvene odgovornosti u izvještavanju o tim aktivnostima. Jedan je od pozitivnih primjera primjena Indeks DOP-a. No potrebno je aktivnije angažiranje od strane sustava te u obliku formiranja poticajnih mjera za društveno odgovorno poslovanje od strane državnih i EU institucija. Metodologija definira niz kriterija za ocjenjivanje razine primjene društveno odgovornih praksi koji su podijeljeni u šest osnovnih područja: ekonomska održivost, uključenost društveno odgovornog poslovanja u poslovnu strategiju, radna okolina, zaštita okoliša, tržišni odnosi i odnosi sa zajednicom kao što je prikazano na slici 2.3.1 [10]




Sl. 2. 3. 1. Razvijenost DOP-a [3]

Postoji mnogo tvrtki koje primjenjuju koncept društveno odgovornog poslovanja, a među hrvatskima se još može izdvojiti: Hrvatska lutrija, Podravka, Privredna banka Zagreb, Končar i Ciklopea. Indeks DOP-a kao nagrada za društveno odgovorno poslovanje u Republici Hrvatskoj dodatno utječe na poduzeća da ulažu u društveno odgovorno poslovanje i brinu se za zajednicu. Ukoliko korisnici prepoznaju napore određenog poduzeća, odnosno njegovog marketinga koji primjenjuje koncept CRM- a, spremni su nagraditi to poduzeće svojom lojalnošću kao i kazniti poduzeća kojima je razina društveno odgovornog ponašanja vrlo niska. Kako bi se to uspješno provodilo bitno je paziti na emocije korisnika, vrstu dobrotvorne svrhe koja se podupire, visinu donacija i sl. U kontekstu ovoga rada korisnici su studenti kojima bi primjena CRM-a omogućila lakše i kvalitetnije studiranje te mogućnost daljnjega usmjeravanja prema budućim poslodavcima. [1], [12]

# 3. KONCEPT UPRAVLJANJA ODNOSIMA S KORISNICIMA USLUGA

Zadovoljstvo klijenata najvažnija je odrednica suvremenoga poslovanje te je to rezultiralo sustavima praćenja zadovoljstva klijenata. Danas se zadovoljstvo klijenata kvalificira prema sljedećim točkama: vrlo nezadovoljni, nezadovoljni, ravnodušni, zadovoljni i vrlo zadovoljni. U mjerenju zadovoljstva mjeri se svaki dio ponude kao i post-prodajne aktivnosti. Upravljanje je odnosima s klijentima u neposrednoj korelaciji i s marketingom kao i sa prodajom. Upravljanje odnosima s kupcima predstavlja poslovnu strategiju orijentiranu kupcima s osnovnim ciljem povećanja prihoda pomoću poboljšanje interakcije s kupcima. Ključne su funkcionalnosti CRM-a jednostavna lista svih korisnika, partnera na nivou poduzeća, cjeloviti pogled na korisnike odnosno partnere na jednom mjestu, jedinstveno vođenje cjelokupne korespondencije s korisnicima, upravljanje prijedlozima i poslovnim prilikama, vođenje prodajnog procesa, upravljanje segmentima, marketinškim i prodajnim akcijama te praćenje konkurencije. Sve ove stavke pod nadležnosti su prodaje i marketinga dok su upravljanje servisnim zahtjevima, automatizacija rješavanja zahtjeva, knjižnica znanja, hodogrami za automatizaciju prodajnih, marketinških i servisnih aktivnosti, upravljanje ugovorima, narudžbama te integracija s elektronskom poštom i potpora za mail *merge* pod djelovanjem korisničke podrške. Čimbenici koji utječu na CRM brojni su, a čine ih: prepoznavanje potreba korisnika, dugoročan odnos s korisnicima, djelotvoran prijenos informacija, tehnologija, organizacijska struktura, interakcija s korisnikom, upravljanje korisničkim uslugama, kao i procesi. CRM se može podijeliti na 3 dijela: operativni, kolaborativni i analitički. Korisnosti upravljanja odnosima s potrošačima su sljedeće: trenutno prikupljanje i raspolaganje „svježim“ podatcima i rezultatima izravno s „terena“ tj. tržišta, pouzdanije predviđanje buduće prodaje, objedinjavanje i razmjena znanja o kupcima prodaje i post-prodajnih usluga (dostava, montaža, recikliranje i dr.). [2]

U kontekstu ove teme rada, koncept upravljanja odnosima s korisnicima (studentima) omogućavao bi lakše vođenje evidencije o položenim odnosno ne položenim kolegijima i obavezama studenata te bi profesorima bilo omogućeno lakše provođenje ispitnih rokova na temelju podataka o svakom studentu. Optimiziralo bi se vrijeme i troškovi održavanja ispita, a time i poboljšala kvaliteta rada. Izvrsnost pojedinog studenta u pojedinim nastavnim kolegijima bi se lakše sustavno pratila, a to bi rezultiralo njihovom lakšem usmjeravanju prema praksi kod budućih mogućih poslodavaca koji su spremni studente završnih godina studija regrutirati i još poučavati u smjeru svojih potreba.

# 4. INFORMACIJSKI SUSTAVI I BAZE PODATAKA

Informacijski sustav (IS) nekog tehnološkog ili organizacijskog sustava dio je sustava koji opskrbljuje informacijama sve razine upravljanja i odlučivanja u sustavu. Zbog potreba za brzim i efikasnim upravljanjem velikim brojem podataka CRM sustav dio je poslovnog informacijskog sustava. Podatci su osnovni elementi informacijskog sustava. Operacije koje su pod djelovanjem informacijskog sustava su zapisivanje, obrada, čuvanje i razdioba podataka i informacija. Pod pojmom zapisivanja smatra se zapisivanje podataka i informacija na nosioce koji su za sustav čitljivi. Obrada podataka i informacija predstavlja proces transformacije ulaznih veličina podataka i informacija u izlazne veličine odnosno rezultate obrade. Čuvanje podataka i informacija predstavlja bitnu operaciju unutar informacijskog sustava koju karakterizira memoriranje podataka i informacija u baze podataka kako bi se korisnici njima služili u kasnijim procesima obrade ili odlučivanja. Razdioba podataka i informacija omogućuje korištenje i rad s podatcima i informacijama u željenom trenutku. Time se povećava učinkovitost samog informacijskog sustava. Uspostavljanjem informacijskoga sustava unutar poduzeća, institucija ili organizacija javlja se opći porast profesionalnosti rada svih djelatnika u dosegu sustava.[14]

Informacijski sustavi informacijski poslužuju poslovne procese i operacije, poslovno odlučivanje te razvijanje i implementaciju kompetitivnih strategija poslovanja. Svrha je informacijskog sustava opskrbiti poslovni sustav informacijama potrebnim za izvođenje poslovnog procesa, upravljanje poslovnim procesom kao i uspostavljanje i održavanje komunikacije u sustavu i sustava s okruženjem. Potpora su izvršavanju poslovnih procesa te potpora u odlučivanju i upravljanju. Zadatak im je osigurati podatke i informacije za potrebe izvršavanja poslovnog procesa kao i osigurati informacije i znanje za potrebe odlučivanja i upravljanja poslovanjem. Kao takvi informacijski sustavi olakšavaju rad, oslobađaju od rutinskih poslova, čine korisnike efikasnijim u obavljanju svakodnevnih zadataka, čine informacije lakše dostupnima i pouzdaniji su od čovjeka u pogledu mogućih pogrešaka te omogućavaju brže donošenje odluka. [17]

U svrhu informatizacije visokih učilišta u Republici Hrvatskoj pokrenut je projekt Ministarstva znanosti, obrazovanja i sporta 2000. godine pod nazivom Informacijski sustav visokih učilišta (ISVU). Informacijski sustav visokih učilišta besplatan je informacijski sustav koji nudi besplatnu edukaciju za njegovo korištenje te omogućuje izradu izvještaja, manju opterećenost studenske referade te izradu rasporeda sati. Studentima omogućuje lakšu prijavu i odjavu ispita, dobivanje različitih potvrda te upis godine bez čekanja u redu. Profesorima olakšava uređivanje

tjednog plana nastave, opisivanje podataka o predmetu kao i pregled podataka o ispitnim rokovima i statusima studenata.[15]

Kriteriji za podjelu informacijskih sustava različiti su, a najčešće korišteni su podjela prema konceptualnom ustrojstvu poslovodstva, prema namjeni ili prema modelu poslovnih funkcija u poslovnom sustavu. Prema ustrojstvu poslovodstva, poslovodstvo dijelimo na tri razine operativni nivo, strateški nivo i poslovni nivo. Operativnom nivou namijenjeni su transakcijski sustavi (namijenjeni za izvođenje procesa osnovne djelatnosti). To su primjerice sustavi kojima se knjiže bankarske transakcije. Operativni nivo ISVU-a kao informacijskog sustava visokih učilišta Republike Hrvatske omogućuje izdavanje različitih potvrda izvan radnoga vremena studenske referade. Taktičkoj razini namijenjeni su izvršni informacijski sustavi čiji su rezultat izvješća važna za upravljanje poslovodstvom. ISVU omogućuje izradu izvještaja koje periodički zahtijevaju visoko učilište, stipenditori ili slično. Strateškoj razini namijenjeni su sustavi potpore odlučivanja. ISVU kao sustav potpore odlučivanju omogućava profesorima lakše uređivanje plana nastave za predavanja i vježbe te pregled podataka o ispitnim rokovima kao i statusima studenata.[18]

Prema namjeni informacijske sustave dijelimo na sustave obrade podataka i informacija (služe za obradu i pohranjivanje podataka i informacija o stanju sustava i poslovnim događajima), sustave podrške uredskom radu (sustavi za obradu dokumenata, prezentacija i slično), sustave podrške u odlučivanju i ekspertne sustave (sustavi koji služe kao podrška stručnjacima i ekspertima te služe za rješavanja različitih problema primjerice dijagnosticiranja problema.).[19] Jedan od primjera takvih sustava je ISVU.

Neovisno o tipu i vrsti informacijskog sustava, u njegovoj bazi podataka pohranjeni su podaci i informacije potrebni za daljnju obradu i izvještavanje. O kvaliteti tih podataka i informacija ovisit će i kvaliteta informacijskog sustava.

Baza podataka predstavlja skup međusobno povezanih podataka pohranjenih u vanjskoj memoriji računala. Podatci unutar same baze dostupni su različitim korisnicima, a najvažnije operacije nad podatcima su zapisivanje, promjena, brisanje i čitanje podataka. Podatci u bazi logički su organizirani u skladu s nekim modelom podataka. Modelom podataka određena su pravila kako može izgledati logička struktura baze. Osnovni modeli su relacijski model (podatci i veze prikazuju se pravokutnim tablicama), mrežni model (baza je predočena usmjerenim grafom), hijerarhijski model (baza je predočena jednim stablom ili skupom stabala) i objektni

model (baza je skup trajno pohranjenih objekata). Korištenjem baze podataka želi se postići fizička i logička neovisnost podataka, fleksibilnost pristupa podatcima, čuvanje integriteta, mogućnost oporavka u slučaju kvara te zaštita od neovlaštenog korištenja. Razvoj programskih jezika omogućio je korisnicima jednostavniji i lakši rad s podatcima u bazi. Najvažniji programski jezici koji se koriste za rad s podatcima u bazi podatak su: SQL (engl. Structured Query Language) integrirani jezik koji služi za definiranje podataka, manipuliranje i pretraživanje, XML (engl. Extensible Markup Language) jezik za označavanje podataka te PHP (engl. Hypertext predprocessor) jezik koji služi za dinamičko programiranje web stranica.[16]

# 5. DIZAJNIRANJE I IZRADA BAZE PODATAKA

U sljedećim odlomcima prikazano je dizajniranje i upravljanje bazom podataka studenata Fakulteta elektrotehnike, računarstva i informacijskih tehnologija u Osijeku te će biti definiran i način izrade baze podataka potrebne za kasniji rad s podatcima.

## 5.1. Dizajniranje baze podataka

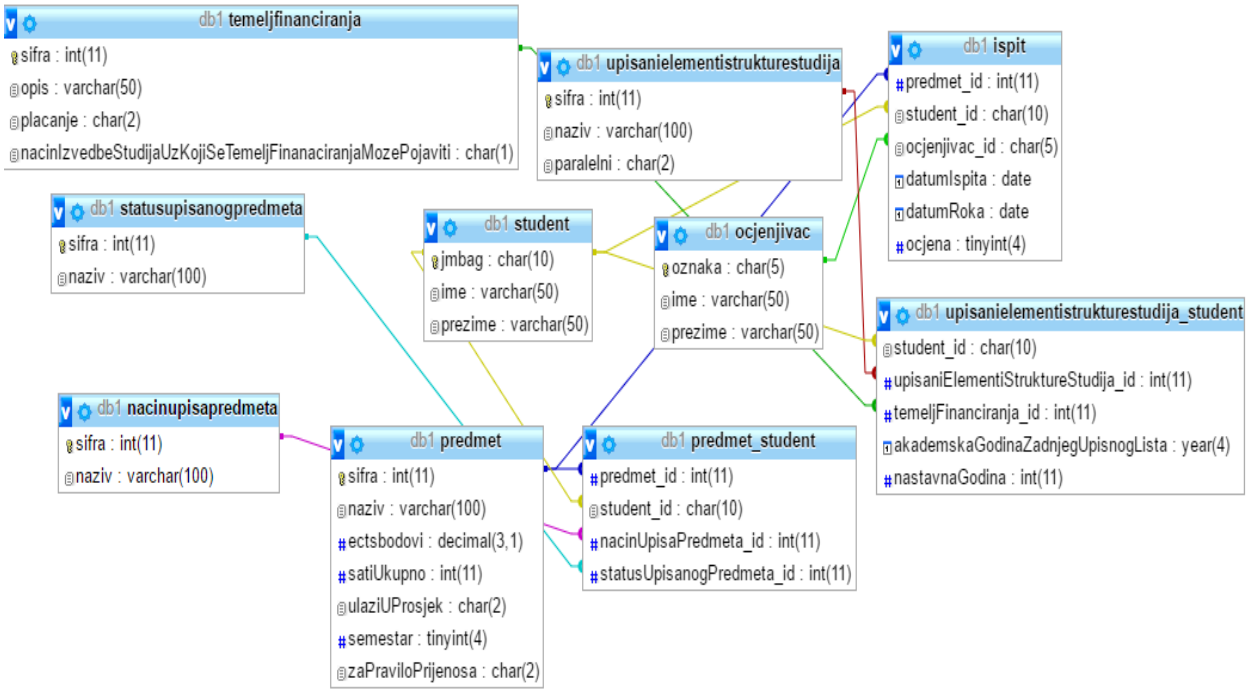
Upravljanje podatcima i provođenje kontrole nad podatcima u relacijskim bazama podataka provodi se naredbama SQL. Kao što je vidljivo na slici 5.1.1. najznačajniji tipovi varijabli koje se koriste su INT (cijeli broj pohranjen u 4 byte-a) zatim CHAR (predstavlja znakovni niz (engl. string)), DECIMAL (predstavlja u posebnom formatu pohranjen decimalni broj), VARCHAR (predstavlja niz znakova varijabilne duljine) i TINYINT (cijeli broj pohranjen u jednom bajtu), a od ostalih korištenih tipova varijabli prilikom dizajniranja ističe se još i tip varijable DATE (vrijednost se prikazuje u obliku datuma).

```
DROP DATABASE IF EXISTS db1;
CREATE DATABASE db1 CHARACTER SET utf8 COLLATE utf8_unicode_ci;
use db1;
CREATE TABLE predmet(
     sifra                    INT NOT NULL,
     naziv                    VARCHAR(100),
     ectsbodovi               DECIMAL(3,1),
     satiUkupno               INT,
     ulaziUProsjek            CHAR(2),
     semestar                 tinyint,
     zaPraviloPrijenosa       CHAR(2),
     CONSTRAINT PK_predmet PRIMARY KEY(sifra)
);
```

Sl. 5.1.1. Dizajniranje baze podataka ; izvor:autor

## 5.2. Shema baze podataka

Shema baze podataka prikazana na slici 5.2.1. sadrži tablice *Predmet* s atributima (*sifra*, *naziv*, *ectsbodovi*, *satiUkupno*, *ulaziUProsjek*, *semestar*, *zaPraviloPrijenosa*) s primarnim ključem *sifra*. *Ocjenjivac* s atributima (*oznaka*, *ime*, *prezime*, s primarnim ključem *oznaka*). *Student* s atributima (*jmbag*, *ime* i *prezime*). Zatim tablice *statusupisanogpredmeta* i *nacinupisapredmeta* s primarnim ključem *sifra* u obje tablice. Nadalje tu su još tablice *upisanielementistrukturestudija_Student*; *upisanielementistrukturestudija*, *temeljfinanciranja*, *ispit* i *predmet_student.*




Sl. 5.2.1. Shema baze podataka, izvor : autor

Kako bi povezali podatke u bazi podataka naredbom ALTER TABLE dodavani su strani ključevi radi lakše izmjene podataka u slučaju pogreške te preglednosti samoga koda kao što je prikazano na slici 5.2.2.

```
ALTER TABLE predmet_student ADD CONSTRAINT FK_predmet_student_student
FOREIGN KEY(student_id) REFERENCES student(jmbag);
```

Sl. 5.2.2. Korištenje naredbe ALTER TABLE; izvor autor

# 6. IMPLEMENTIRANJE CRM-A NA PRIMJERU FERIT OSIJEK

U sljedećim potpoglavljima će biti pokazan način uspostave veze s bazom podataka, zapisivanje podataka u bazu podataka, kreiranje upita i prikaz rezultata upita. Za potrebe prikaza rezultata upita korišteni su podatci studenata Fakulteta elektrotehnike, računarstva i informacijskih tehnologija Osijek.

## 6.1. Uspostavljanje veze

Za uspostavu veze s bazom podataka koristi se PHP skriptni serverski programski jezik. Kako bi veza bila uspostavljena važno je definirati ime servera kao što je prikazano na slici 6.1.1., u ovome slučaju to je broj "127.0.0.1". Šifra nije bila potrebna dok je korisničko ime uvijek jednako i glasi "Root". Naredbom $CONN uspostavlja se veza baze podataka sa korištenim serverom kao što je prikazano na slici 6.1.1.

```
<?php
ini_set('default_charset', 'utf-8');
$servername = "127.0.0.1";
$username = "root";
$password = "97865210";
$database = "db1";


try {
    $conn = new
PDO("mysql:host=$servername;dbname=$database;charset=utf8",
$username, $password);
    // set the PDO error mode to exception
    $conn->setAttribute(PDO::ATTR_ERRMODE, PDO::ERRMODE_EXCEPTION);
```

Sl. 6.1.1. Uspostavljanje veze; izvor: autor

## 6.2. Zapisivanje podataka u bazu podataka

Zapisivanje podataka u samu bazu omogućeno je naredbom INSERT INTO koja vrijednosti atributa tablica upisuje u bazu podataka. Iza naredbe VALUES nalazi se u zagradi lista vrijednosti korištenih atributa, u ovom slučaju to su *sifra, naziv, paralelni* kao što je prikazano na slici 6.2.1.

```
$upisaniElementStruktureStudija_sql = $conn->prepare("INSERT INTO
upisanielementistrukturestudija (sifra, naziv, paralelni) VALUES
(:sifra, :naziv, :paralelni)");

$temeljFinanciranja_sql = $conn->prepare("INSERT INTO
temeljfinanciranja (sifra, opis, placanje,
nacinIzvedbeStudijaUzKojiSeTemeljFinanaciranjaMozePojaviti) VALUES
(:sifra, :opis, :placanje,
:nacinIzvedbeStudijaUzKojiSeTemeljFinanaciranjaMozePojaviti)");
```

Sl. 6.2.1. Zapisivanje podataka u bazu; izvor: autor

Naredba *bindParam* služi za zapisivanje podataka u bazu podataka tako da se u tablicu *student_sql* prikazanu na slici 6.2.2. zapisuju svi podatci vezani za pojedinoga studenta odnosno njegov *jmbag, ime i prezime*. Isto to se odnosi i na sve ostale tablice baze podataka.

```
$student_sql->bindParam(':jmbag', $student_jmbag);

$student_sql->bindParam(':ime', $student_ime);

$student_sql->bindParam(':prezime', $student_prezime);

$upisaniElementStruktureStudija_sql->bindParam(':sifra',
$upisaniElementStruktureStudija_sifra);

$upisaniElementStruktureStudija_sql->bindParam(':naziv',
$upisaniElementStruktureStudija_naziv);
```

Sl. 6.2.2. Korištenje naredbe *bindParam;* izvor :autor

Naredbom *scandir* vraća se polje datoteka specifičnog foldera kao što je prikazano na slici 6.2.3. Dohvaćaju se datoteke s podatcima o studentima iz foldera *studentsumarno*. Naredba *count* vraća broj datoteka u folderu *studentsumarno* te se u *for* petlji prolazi kroz sve podatke počevši od treće datoteke budući da prve dvije datoteke nisu datoteke već označavaju trenutni folder u

kojem se korisnik nalazi i njemu nadređeni folder te su oni predstavljeni simbolima „ .“ za trenutni folder u kojem se korisnik nalazi i „ :" kao njemu nadređeni (engl. *Parent directory*).[20] U datotekama su zapisani podatci o svim studentima neovisno o prestanku rada programa.

```
$files = scandir('./studentsumarno');

$n = count($files);

var_dump($n);

try

{for( $i = 2; $i < $n; ++$i)

{

$myXMLData = file_get_contents('./studentsumarno/' . $files[$i]);

$xml = simplexml_load_string($myXMLData) or die("Error:
Cannotcreateobject");

$student_jmbag = $xml->student->jmbag;

$student_ime = $xml->student->ime;

$student_prezime = $xml->student->prezime;

if( empty((array) $student_jmbag) )

{continue;

}

$student_sql->execute();
```

Sl. 6.2.3. Korištenje naredbi *scandir* i *count;* izvor: autor

U podatcima o studentima Fakulteta elektrotehnike, računarstva i informacijskih tehnologija pronađen je ne mali broj studenata koji nisu imali pohranjene informacije o *jmbag*-u pa je *if* naredbom definirano da ukoliko je polje *student_jmbag* prazno tada se taj student zanemaruje, zbog toga baza podataka sadrži informacije od oko 2404 studenta umjesto 2500 studenta. Naredba *array* označava  polje te je njezina upotreba kao i *if* naredbe prikazana na slici 6.2.4.

```
if( empty((array) $student_jmbag) )
{continue;
}
$student_sql->execute();
```

Sl. 6.2.4. Korištenje funkcije *array;* izvor: autor

Nakon zapisivanja podataka o studentima vrlo važno je i zapisivanje podataka o predmetu, odnosno statusu samoga predmeta. Prvo su definirani nepoloženi predmeti. Naredbom *foreach* omogućeno je izvršavanje naredbi nad elementima u nizu. Za svaki predmet atributi su *sifra*, *naziv*, *ectsbodo*vi, *satiUkupno* zatim *ulaziUProsjek*, *semestar* i *zaPraviloPrijenosa*. Naredbom *if* provjerava se nalazi li se vrijednost atributa tablice *predmet_sifra* u polju *data_predmet_sifra* te ukoliko se ne nalazi zapisuje se vrijednost atributa tablice *predmet* u ovome slučaju *sifra*, a ukoliko se nalazi provjeravaju se ostale vrijednosti budući da su podatci već zapisani u tablicu. Također to vrijedi i za sva ostale vrijednosti te je cijeli postupak prikazan na slici 6.2.5. Naredba *intval* pretvara neki drugi tip podatka npr. string u cjelobrojnu (integer) vrijednost.

```
                        /*
                        NE POLOZENI PREDMETI
                        */
foreach($a->popisNepolozenihPredmeta->nepolozenPredmet as $b)
{
$predmet_sifra = $b->predmet['sifra'];
$predmet_naziv = $b->predmet->naziv;
$predmet_ectsbodovi = $b->predmet->ectsBodovi;
$predmet_satiUkupno = $b->predmet->satiUkupno;
$predmet_ulaziUProsjek = $b->predmet->ulaziUProsjek;
$predmet_semestar = $b->predmet->semestar['redniBroj'];
$predmet_zaPraviloPrijenosa = $b->predmet->zaPraviloPrijenosa;
if( !in_array( intval($predmet_sifra), $data_predmet_sifra) )
{
$data_predmet_sifra[] = intval($predmet_sifra);
$predmet_sql->bindParam(':sifra', $predmet_sifra);
$predmet_sql->bindParam(':naziv', $predmet_naziv);
$predmet_sql->bindParam(':ectsbodovi', $predmet_ectsbodovi);
```

Sl. 6.2.5. Zapisivanje podataka o nepoloženim predmetima; izvor: autor

Osim nepoloženih predmeta definirani su i položeni predmeti s razlikom da ukoliko je predmet položen potrebne su informacije o ispitu, dakle datumu ispitnog roka i datumu ispita i informacije o ocjenjivaču i ocjeni. Podatci o položenim predmetima definirani su na isti način kao i nepoloženi kao i zapisivanje podataka o položenim predmetima *if* naredbom kao što je prikazano na slici 6.2.6.

```
/*
                            POLOZENI PREDMETI
                        */
foreach($a->popisPolozenihPredmeta->polozenPredmet as $c)
{
$predmet_sifra = $c->predmet['sifra'];
$predmet_naziv = $c->predmet->naziv;
$predmet_ectsbodovi = $c->predmet->ectsbodovi;
$predmet_satiUkupno = $c->predmet->satiUkupno;
$predmet_ulaziUProsjek = $c->predmet->ulaziUProsjek;
$predmet_semestar = $c->predmet->semestar['redniBroj'];
$predmet_zaPraviloPrijenosa = $c->predmet->zaPraviloPrijenosa;
if( !in_array( intval($predmet_sifra), $data_predmet_sifra) )
{
$data_predmet_sifra[] = intval($predmet_sifra);
$predmet_sql->bindParam(':sifra', $predmet_sifra);
$predmet_sql->bindParam(':naziv', $predmet_naziv);
$predmet_sql->bindParam(':ectsbodovi', $predmet_ectsbodovi);
```

Sl. 6.2.6. Zapisivanje podataka o položenim predmetima; izvor:autor

Kao što je već navedeno potrebno je dohvatiti podatke o ocjenjivaču, dakle njegovu *oznaku*, *ime* i *prezime* te je potrebno zapisati vrijednosti atributa tablice *ocjenjivac* u polje tablice *ocjenjivac* kako bi mogli koristiti informacije o pojedinom ocjenjivaču kako je prikazano na slici 6.2.7.

```
$ispit_ocjenjivac_oznaka = $c->ispit->ocjenjivac->oznaka;

$ispit_ocjenjivac_ime = $c->ispit->ocjenjivac->ime;

$ispit_ocjenjivac_prezime = $c->ispit->ocjenjivac->prezime;

if( !in_array( (string) $ispit_ocjenjivac_oznaka, $data_ocjenjivac)
){

$data_ocjenjivac[] = (string) $ispit_ocjenjivac_oznaka;

$ocjenjivac_sql->bindParam(':oznaka', $ispit_ocjenjivac_oznaka);

$ocjenjivac_sql->bindParam(':ime', $ispit_ocjenjivac_ime);

$ocjenjivac_sql->bindParam(':prezime', $ispit_ocjenjivac_prezime);

$ocjenjivac_sql->execute();
```

Sl. 6.2.7. Zapisivanje podataka o ocjenjivaču; izvor : autor

Isto tako potrebno je zapisati i definirati podatke tablice *ispit* te atribute *datumIspita*, *datumRoka* i *ocjenu*. Također naredbom *bindParam* zapisujemo podatke u tablicu *ispit* kao što je prikazano na slici 6.2.8.

```
$ispit_datumIspita = $c->ispit['datumIspita'];

$ispit_datumRoka = $c->ispit->datumRoka;

$ispit_ocjena = $c->ispit->ocjena;

$ispit_sql->bindParam(':predmet_id', $predmet_sifra);

$ispit sql->bindParam(':student id', $student jmbag);
```

Sl. 6.2.8. Zapisivanje podataka o ispitu; izvor: autor

## 6.3. Prikaz rezultata upita

U ovom dijelu rada objašnjen je način upravljanja s podatcima u bazi. Zapisivanjem podataka o svim čimbenicima baze podataka omogućeno je upravljanje i rad s podatcima u bazi podataka. Kao rezultat toga izvršavaju se određeni upiti nad samom bazom.

### 6.3.1. Upiti nad bazom podataka

Pisanje upita nad podatcima baze podataka preduvjet je za dobivanje informacija o imenima i prezimenima studenata koji su položili ili nisu položili neki predmet. U prvom primjeru zahtijevaju se imena i prezimena studenata koji nisu položili neki predmet. Prvenstveno se mora naredbom JOIN spojiti tablice *student* i *predmet_student* preko ključeva *jmbag* i *student_id* . Nakon toga treba spojiti tablice *predmet_student* i *predmet* preko ključeva *predmet_id* i *sifra*, a zatim tablicu *statusupisanogpredmeta* i *predmet_student* preko ključeva *sifra* i *statusUpisanogPredmeta_id* kao što je prikazano na slici 6.3.1.1.

```
JOIN predmet_student ps ON ps.student_id=s.jmbag

JOIN predmet p ON p.sifra=ps.predmet_id

JOIN statusupisanogpredmeta sup ON
sup.sifra=ps.statusUpisanogPredmeta_id
```

Sl. 6.3.1.1. Spajanje tablica *student* i *predmet_student* , *predmet_student* i *predmet* te tablica *statusupisanogpredmeta* i *predmet_student*; izvor: autor

Nakon toga spajaju se još tablice *upisanielementistrukturestudija_student* i *student* te *upisaniElementiStruktureStudija* i *upisanielementistrukturestudija_student*. Podatci su sortirani prema atributu tablice *upisaniElementiStruktureStudija*, *naziv* upisanog elementa studija kao što je prikazano je na slici 6.3.1.2.

```
JOIN upisanielementistrukturestudija_student uess_student ON
uess_student.student_id=s.jmbag

JOIN upisaniElementiStruktureStudijauess ON
uess.sifra=uess_student.upisaniElementiStruktureStudija_id

WHERE p.sifra=:predmet_sifra AND sup.naziv=:status_predmeta  AND
uess.sifra=:smjer

ORDER BY uess.naziv");
```

Sl. 6.3.1.2. Spajanje tablica *upisanielemtistrukturestudija_student* i *student* i *upisaniElementiStruktureStudija* te *upisanielementistrukturestudija_student*; izvor: autor

U drugome primjeru kada se zahtijevaju imena i prezime studenata koji su položili predmet ili im je isti priznat tada se naredbom JOIN spajaju tablice *student* i *predmet_student*, nakon toga *predmet_student* i *predmet*, zatim tablice *statusupisanogpredmeta* i *predmet_student*. Spajaju se i tablice *upisanielementistrukturestudija_student* i *student* te kao i u prvom primjeru tablice *upisaniElementiStruktureStudija* i *upisanielementistrukturestudija_student* kao što je prikazano je na slici 6.3.1.3.

```
JOIN predmet_student ps ON ps.student_id=s.jmbag

JOIN predmet p ON p.sifra=ps.predmet_id

JOIN statusupisanogpredmeta sup ON
sup.sifra=ps.statusUpisanogPredmeta_id

JOIN upisanielementistrukturestudija_student uess_student ON
uess_student.student_id=s.jmbag

JOIN upisaniElementiStruktureStudijauess ON
uess.sifra=uess_student.upisaniElementiStruktureStudija_id
```

Sl. 6.3.1.3. Spajanje tablica kada se zahtijevaju imena i prezimena studenata koji su položili neki predmet; izvor: autor

Novina u ovome primjeru je spajanje tablica *ispit* i *student* preko ključeva *student_id* i *jbmag* te tablica *ocjenjivac* i *ispit* jer je potrebno ispisati imena i prezimena studenata koji su položili taj kolegij ili im je priznat te ocjene i datum polaganja uz ime i prezime ocjenjivača koji im je istu

ocjenu i upisao. Također je sve sortirano prema atributu tablice *upisaniElementiStruktureStudija*, *naziv* upisanog elementa studija kao što je prikazano je na slici 6.3.1.4.

```
JOIN ispit i ON s.jmbag=i.student_id

JOIN ocjenjivacoc ON i.ocjenjivac_id=oc.oznaka

WHERE p.sifra=:predmet_sifra AND sup.naziv=:status_predmeta  AND
uess.sifra=:smjer

ORDER BY uess.naziv");
```

Sl. 6.3.1.4. Spajanje tablica *ispit* i *student* te *ocjenivac* i *ispit*; izvor: autor

### 6.3.2. Prikaz studenata

Kako bi rezultati upita bili vidljivi treba se definirati način prikaza i dohvaćanje podataka o studentima iz svakog stupca tablica nizom naredbi koje su prikazane na slici 6.3.2.1.

```
$get_predmet_sql->bindParam(':sifra_predmeta', $predmet_sifra);

$get_predmet_sql->execute();

$predmet = $get_predmet_sql->fetchColumn();

$get_smjer_sql->bindParam(':sifra_smjer', $smjer_sifra);

$get_smjer_sql->execute();

$smjer = $get_smjer_sql->fetchColumn();

?>
```

Sl. 6.3.2.1. Dohvaćanje podataka o studentima; izvor:autor

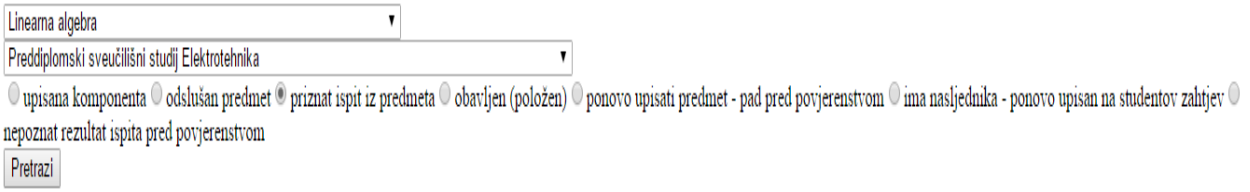
Funkcija *echo* ima svrhu ispisa podataka na ekran kao što vidljivo na slici 6.3.2.2.

```
echo '<td>Ocjena</td><td>Datum Ispita</td><td>Datum
roka</td><td>Ocjenjivac</td</tr>';

echo '<tr><td>' . $polozen_predmet['jmbag'] . '</td><td>' .
$polozen_predmet['student'] . '</td><td>' .
$polozen_predmet['ocjena'] . '</td>';echo '<td>' .

echo '<tr><td>' . $nepolozen_predmet['jmbag'] . '</td><td>'
. $nepolozen_predmet['student'] . '</td></tr>';
```

Sl. 6.3.2.2. Ispis podataka o studentima; izvor:autor

## 6.4. Rezultati upita

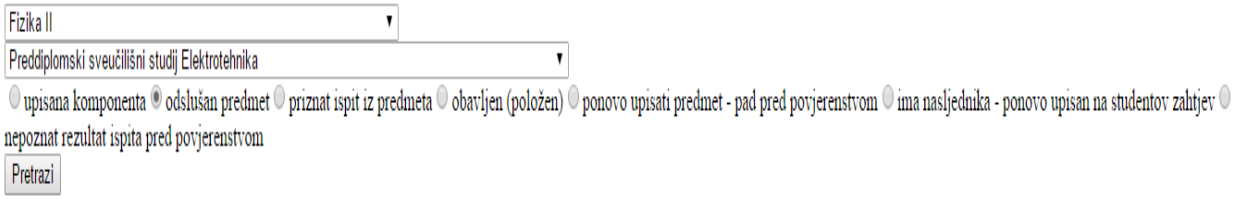
Nakon što su definirani upiti nad podatcima u bazi te je omogućen prikaz rezultata definiranih upita konačni će rezultat biti njihovo izvršavanje te ispis. U prvom primjeru odabire se kolegij *Linearna algebra*, smjer *Preddiplomski sveučilišni studij Elektrotehnika* te opcija *priznat ispit iz predmeta* kao što je prikazano na slici 6.4.1. te se kao rezultat ispisuju imena studenata kojima je priznat ispit iz toga kolegija kao što je prikazano na slici 6.4.2.




Sl. 6.4.1. Odabir opcije *priznat ispit iz predmeta*; izvor: autor

Sl. 6.4.2. Studenti kojima je priznat ispit iz kolegija *Linearna algebra*; izvor: autor

U drugome primjeru odabire se kolegij *Fizika 2* dok smjer ostaje nepromijenjen. Uz to kao razlika u odnosu na prvi primjer odabire se opcija *odslušan predmet* kao što je prikazano na slici 6.4.3. te se kao rezultat ispisuju imena i prezimena studenata koji su odslušali, ali nisu položili kolegij *Fizika 2* kao što je prikazano na slici 6.4.4.



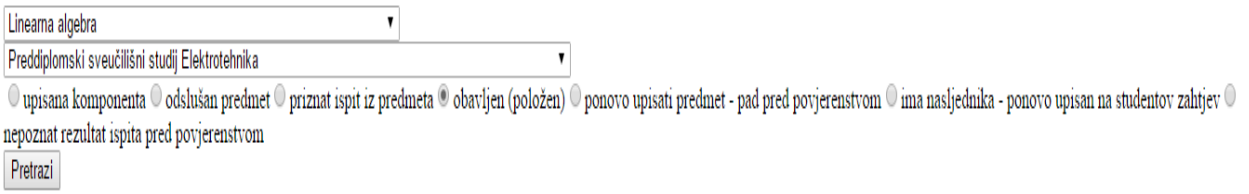

Sl. 6.4.3. Izbor kolegija *Fizika 2* i opcije odslušan predmet; izvor: autor

**Preddiplomski sveučilišni studij Elektrotehnika: Fizika II**

| JMBAG | Student |
|---|---|
| | Adam |
| | Matej |
| | Luka |
| | Antonio |
| | Zvonimir |
| | Nikola |
| | Emil |
| | Karlo |
| | Alan |
| | Sven |
| | Adam |
| | Kristijan |
| | Ivan |
| | Matija |
| | Filip |
| | Filip |
| | Matteo |
| | Lazar |
| | Boris |
| | Franjo |

Sl. 6.4.4. Popis studenata koji su odslušali kolegij *Fizika 2*, a nisu ga položili; izvor: autor

U trećem primjeru traži se popis studenata koji su položili kolegij *Linearna algebra* preddiplomskog studija Elektrotehnike s datumom ispitnoga roka i datumom ispita te imenom i prezimenom ocjenjivača kao što je prikazano na slici 6.4.5. i slici 6.4.6.




Sl. 6.4.5. Odabir opcije *obavljen(položen)*, kolegija i smjera; izvor: autor

## Preddiplomski sveučilišni studij Elektrotehnika: Linearna algebra

| JMBAG | Student | Ocjena | Datum Ispita | Datum roka | Ocjenjivac |
|---|---|---|---|---|---|
| | Ilija | 4 | 07.02.2014. | 03.02.2014. | Tomislav Rudec |
| | Filip | 3 | 21.02.2014. | 17.02.2014. | Tomislav Rudec |
| | Ivan | 2 | 10.09.2010. | 02.09.2010. | Radoslav Galić |
| | Mario | 2 | 12.07.2011. | 05.07.2011. | Radoslav Galić |
| | Filip | 3 | 14.07.2014. | 23.06.2014. | Tomislav Rudec |
| | Mijo | 4 | 16.09.2013. | 09.09.2013. | Radoslav Galić |
| | Perica | 4 | 21.02.2014. | 17.02.2014. | Tomislav Rudec |
| | Matija | 3 | 14.02.2015. | 09.02.2015. | Tomislav Rudec |
| | Luka | 3 | 14.02.2015. | 09.02.2015. | Tomislav Rudec |
| | Viktorija | 4 | 08.02.2013. | 30.01.2013. | Radoslav Galić |
| | Ante | 3 | 10.02.2016. | 04.02.2016. | Radoslav Galić |
| | Mihael | 2 | 24.09.2010. | 16.09.2010. | Radoslav Galić |
| | Mateja | 3 | 28.07.2007. | 28.07.2007. | Radoslav Galić |
| | Nikola | 3 | 09.07.2009. | 09.07.2009. | Radoslav Galić |
| | Jurica | 3 | 19.02.2007. | 19.02.2007. | Radoslav Galić |
| | Matija | 2 | 15.10.2009. | 15.10.2009. | Radoslav Galić |
| | Milena | 3 | 07.09.2007. | 28.08.2007. | Radoslav Galić |
| | Zoran | 2 | 02.10.2008. | 02.10.2008. | Radoslav Galić |
| | Nikola | 3 | 06.10.2010. | 01.10.2010. | Radoslav Galić |
| | Davor | 2 | 21.09.2009. | 21.09.2009. | Radoslav Galić |
| | Igor | 3 | 01.07.2008. | 18.06.2008. | Radoslav Galić |
| | Zoran | 2 | 18.06.2008. | 18.06.2008. | Radoslav Galić |
| | Zlata | 2 | 10.10.2008. | 02.10.2008. | Radoslav Galić |

Sl. 6.4.6. Popis studenata koji su obavili kolegij *Linearna algebra*; izvor: autor

# 7. ZAKLJUČAK

Poslovna strategija koja uključuje selekciju i upravljanje odnosom s korisnicima proizvoda i usluga u svrhu optimiranja njihove vrijednosti za tvrtku naziva se Category Relationship Managment (CRM). Sredinom devedesetih godina prošloga stoljeća javljaju se sustavi koji predstavljaju svojevrsnu podršku prodaji kao sustavi namijenjeni korisnicima, odnosno podržavanju istih. Nakon 1996. godine dolazi do spajanja različitih manjih sustava podrške i podsustava u jedan objedinjeni (zajednički) čime dolazi do pojave druge generacije sustava za upravljanje odnosima s korisnicima. 2002. godine, pojavljuje se treća generacija poznatija pod nazivom strateški sustav upravljanja odnosima s korisnicima. Prednosti koje CRM omogućuje su efikasniji poslovni procesi upućeni korisniku, smanjenje gubitaka korisnika od strane konkurenata, povećanje prodaje i tržišnog udjela te smanjenje pritužbi. CRM je prije svega poslovna strategija pa kao takva ima nekoliko vrsta: operativni, kolaborativni i analitički. Podatci o svim studentima Fakulteta elektrotehnike, računarstva i informacijskih tehnologija u Osijeku kao i poboljšanje organizacijskih i komunikacijskih usluga primjenom CRM-a karakteristični su za operativni CRM. Među tvrtkama koje koriste koncept društveno odgovornog ponašanja prednjače Hrvatska lutrija, Podravka, Privredna banka Zagreb, Končar i Ciklopea, no to je još nedovoljan broj tvrtki koje koristi ovakav način poslovnog ponašanja. Primjenom CRM-a u funkciji praćenja studenta omogućilo bi se lakše organiziranje termina nastave, ispita te bi prostorije fakulteta bile bolje iskorištene, odnosno studenti ne bi provodili toliko vremena u njima budući kako bi bili podijeljeni u manje grupe radi lakšega i bržega ispitivanja te bi brže dobili informacije o uspjehu na pojedinom ispitnom roku. Također zbog smanjenih kapaciteta Fakulteta elektrotehnike, računarstva i informacijskih tehnologija u Osijeku studenti do sada nisu mogli u potpunosti koristiti svu opremu koja bi im pomagala u učenju i polaganju ispita. Primjenom CRM-a bilo bi im omogućeno korištenje istih zbog bolje organizacije nastavnih sati kao i manjeg broja studenata po grupama. Sukladno tome to bi vodilo k poboljšanju kvalitete rada kao i povećanju zadovoljstva studenata, ali i profesora. Baza podataka predočena u ovome radu omogućuje lakše snalaženje i pretragu podataka o studentima Fakulteta elektrotehnike, računarstva i informacijskih tehnologija, to jest lakše vođenje evidencije uspješnosti istih. Vrlo preciznim postupkom izrade same baze podataka kao i zapisivanja podataka u bazu korisnicima se omogućuje brža, jednostavnija i efikasnije pretraga evidencije o uspješnosti studenta, a od korisnika se zahtjeva isključivo odabir željenog smjera, predmeta te odabir odgovarajuće opcije ovisno o informacijama koje korisnik želi saznati. Na taj način korisnici neće morati dugotrajnim

postupkom pronalaziti studente koji su položili ili nisu položili neki predmet odnosno kojima su ocjene iz pojedinih predmeta priznate što se od ovog zadatka i očekivalo.

# LITERATURA


[1] J. Muller, J.Singh,CategoryManagment: efikasan odgovor na potrebe kupaca u maloprodaji,Internacionalni centar za profesionalnu edukaciju, 2006

[2] B.Dukić,V.Gale,Upravljanje odnosima s potrošačima u funkciji zadržavanja potrošača ,Ekonomski vjesnik, Sveučilište Josipa Jurja Strossmayera u Osijeku,Ekonomski fakultet u Osijeku,2015

[3] Društveno odgovorno poslovanje u Hrvatskoj, http://www.dop.hr/?tag=indeks-dop-a (zadnji pristup 14.6.2016)

[4] W3Schools, http://www.w3schools.com/ (zadnji pristup 08.06.2016 )

[5] CRM Switch, https://www.crmswitch.com/crm-value/understanding-crm-erp/ (zadnji pristup 14.06.2016 )

[6] Php manual, http://php.net/manual/en/simplexml.examples-basic.php (zadnji pristup 08.06.2016 )

[7] Data types,https://msdn.microsoft.com/en-us/library/ms176089.aspx (zadnji pristup 08.06.2016 )

[8] MYSQL,https://www.mysql.com/ (zadnji pristup 08.06.2016)

[9] J.Muller, M.Bezjak, CRM, Visoka škola tržišnih komunikacija AGORA, http://www.vsa.hr/hr/diplomski-studiji/diplomski-studij-kreativnog-upravljanja/crm (zadnji pristup 16.06.2016)

[10] A.Glavoćević, A.Radman Peša, Društveno odgovorno poslovanje i CRM kao način integriranja društvene odgovornosti u marketinške aktivnosti, od 40-42.str, Sveučilište u Zadru, 2013

[11] J.Müller, V.Srića, Upravljanje odnosom s klijentima, Delfin, Zagreb, 2009.

[12] M.Varga, Baze podataka–Konceptualno, logičko i fizičko modeliranje podataka, Društvo za razvoj informacijske pismenosti (DRIP), Zagreb, 1994.

[13] L.Welling, L.Thomson, PHP i MySQL: Razvoj aplikacija za Web, Mikro knjiga, 2009

[14] Uvod u informacijske sustave, UNIZG, http://www.fpz.unizg.hr/ztos/iszp/a2.pdf (zadnji pristup 17.08.2016)

[15] Informacijski sustav visokih učilišta ISVU, http://www.isvu.hr/javno/hr/index.shtml (zadnji pristup 17.08.2016)

[16] R.Manger, Baze podataka, Sveučilište u Zagrebu, Prirodoslovno Matematički fakultet-Matematički odjel, 2003

[17] Željko Panian, Katarina Ćurko, Vesna Bosilj Vukšić, Vlatko Čerić, Mirjana Pejić Bach, Željka Požgaj, Ivan Strugar, Mario Spremić, Mladen Varga, Poslovni informacijski sustavi, Element, Zagreb, 2010

[18] V. Čerić, Poslovno računarstvo, Znak, Zagreb, 1998

[19] V. Strahonja, Projektiranje informacijskih sustava, Zavod za informatičku djelatnost RH i Ina Info, Zagreb, 1992

[20] Stackoverflow, http://stackoverflow.com/questions/11048057/php-scandir-returns-extra-periods (zadnji pristup 16.09.2016)

# SAŽETAK

Category Relationship Managment (CRM)) poslovna je strategija koja uključuje selekciju i upravljanje odnosom s korisnicima proizvoda i usluga u svrhu optimiranja njihove dugoročne vrijednosti za tvrtku. Čimbenici koji utječu na CRM su brojni, a čine ih: prepoznavanje potreba korisnika, dugoročan odnos s korisnicima, djelotvoran prijenos informacija, tehnologija, organizacijska struktura, interakcija s korisnikom, upravljanje korisničkim uslugama, kao i procesi. Primjena CRM u funkciji praćenja studenata od stranke korisnika informacijskog sustava visokog učilišta omogućavala bi lakše vođenje evidencije o položenim i nepoloženim ispitima studenata visokih učilišta, što bi profesorima olakšavalo vođenje evidencije o ispitnim rokovima kao i raspoređivanje studenata u manje grupe na ispitima kako bi se ubrzao proces ispitivanja i ocjenjivanja studenata. Studenti bi bili u mogućnosti više koristiti instrumente koji bi im pomagali u učenju zbog bolje organizacije radnih sati. Kao rezultat toga došlo bi do poboljšanja kvalitete rada. Baza podataka dizajnirana je od deset tablica od kojih su najvažnije ispit, predmet, student koje sadrže sve podatke vezane za pojedine ispitne rokove, studente te informacije o svakom predmetu posebno. Zapisivanjem podataka u svaku od tablica baze omogućen je rad nad podatcima te izvršavanje različitih upita ovisno o željama odnosno potrebama samih korisnika.

## Ključne riječi

CRM, Korisnici, Baza podataka, Visoko obrazovanje, Studenti,

# Application of the CRM approach for monitoring students by a higher education institution


## Abstract

Category Relationship Management (CRM) is a business strategy that includes a selection and customer relationship management products and services to optimize their long-term value for a company. Factors influencing CRM are numerous and they include the recognition of user needs, long-term relationship with customers, effective transfer of information, technology, organizational structure, user interaction, user management services, and processes. A higher education institution can use the CRM approach to keep records of exams students had passed. The approach would facilitate keeping exam records and allocating students to exam group thus speeding up the grading and examining process. Students would be able to use more instruments that would help them in studying due to better organization of working hours. Consequently, the work quality would improve. The database is designed as ten tables the most important of which are exam, course and a student. The database contains all information related to a particular exam period, students and each course separately. Writing data in each database table enables the operation of the data and execution of different queries depending on the wishes and needs of users themselves.



**Keywords**

CRM, Clients, Datebase, Highere education, Students

# Životopis

Ante Anić rođen je 14.studenog 1994. godine u Osijeku. Završio je Osnovnu školu „Ljudevita Gaja“ u Osijeku te nakon toga upisuje I gimnaziju u Osijeku koju također uspješno završava. Osim završene osnovne škole završava i Glazbenu školu „Franje Kuhača“ u Osijeku sa vrlo dobrim uspjehom na instrumentu gitari. Srednju školu završava uspješno, prva dva razreda s vrlo dobrim uspjehom dok treći i četvrti sa odličnim. Sudjeluje na mnogobrojnim natjecanjima poglavito iz sporta, povijesti i biologije. Uzoran je sportaš te se bavi nogometom gdje postiže značajne rezultate od kojih se posebno ističe titula prvaka Hrvatske u amaterskom nogometu 2010. godine. Nakon završene srednje škole upisuje sveučilišni preddiplomski studij elektrotehnike na Elektrotehničkom fakultetu Osijek.

# PRILOZI

## P. 5. 1.

```
DROP DATABASE IF EXISTS db1;
CREATE DATABASE db1 CHARACTER SET utf8 COLLATE utf8_unicode_ci;
use db1;
CREATE TABLE predmet(
     sifra                      INT NOT NULL,
     naziv                      VARCHAR(100),
     ectsbodovi                 DECIMAL(3,1),
     satiUkupno                 INT,
     ulaziUProsjek              CHAR(2),
     semestar                   tinyint,
     zaPraviloPrijenosa         CHAR(2),
     CONSTRAINT PK_predmet PRIMARY KEY(sifra)
);
CREATE TABLE ocjenjivac(
     oznaka                          CHAR(5),
     ime                             VARCHAR(50),
     prezime                         VARCHAR(50),
     CONSTRAINT PK_ocjenjivac PRIMARY KEY(oznaka)
);
CREATE TABLE student(
     jmbag                      CHAR(10),
     ime                             VARCHAR(50),
     prezime                         VARCHAR(50),
     CONSTRAINT PK_student PRIMARY KEY(jmbag)
);
```

```
CREATE TABLE predmet_student(
      predmet_id            INT,
      student_id            CHAR(10),
      nacinUpisaPredmeta_id INT,
      statusUpisanogPredmeta_id INT
);

CREATE TABLE nacinUpisaPredmeta(
      sifra           INT,
      naziv           VARCHAR(100),
      CONSTRAINT PK_nacinUpisaPredmeta PRIMARY KEY(sifra)
);

CREATE TABLE statusUpisanogPredmeta(
      sifra           INT,
      naziv           VARCHAR(100),
      CONSTRAINT PK_statusUpisanogPredmeta PRIMARY KEY(sifra)
);

CREATE TABLE upisaniElementiStruktureStudija_student(
      student_id                CHAR(10),
      upisaniElementiStruktureStudija_id INT,
      temeljFinanciranja_id INT,
      akademskaGodinaZadnjegUpisnogLista YEAR,
      nastavnaGodina            INT
);
```

```
CREATE TABLE upisaniElementiStruktureStudija(

      sifra            INT,

      naziv            VARCHAR(100),

      paralelni        CHAR(2),

      CONSTRAINT PK_upisaniElementiStruktureStudija PRIMARY
KEY(sifra)

);


CREATE TABLE temeljFinanciranja(

      sifra            INT,

      opis             VARCHAR(50),

      placanje         CHAR(2),

      nacinIzvedbeStudijaUzKojiSeTemeljFinanaciranjaMozePojaviti
CHAR(1),

      CONSTRAINT PK_temeljFinanciranja PRIMARY KEY(sifra)

);



CREATE TABLE ispit(

      predmet_id            INT,

      student_id            CHAR(10),

      ocjenjivac_id         CHAR(5),

      datumIspita                DATE,

      datumRoka             DATE,

      ocjena                     TINYINT

);
```

```
ALTER TABLE ispit ADD CONSTRAINT FK_ispit_predmet FOREIGN
KEY(predmet_id) REFERENCES predmet(sifra);

ALTER TABLE ispit ADD CONSTRAINT FK_ispit_student FOREIGN
KEY(student_id) REFERENCES student(jmbag);

ALTER TABLE ispit ADD CONSTRAINT FK_ispit_ocjenjivac FOREIGN
KEY(ocjenjivac_id) REFERENCES ocjenjivac(oznaka);


ALTER TABLE upisaniElementiStruktureStudija_student ADD CONSTRAINT
FK_upisaniElementiStruktureStudija_student_student FOREIGN
KEY(student_id) REFERENCES student(jmbag);

ALTER TABLE upisaniElementiStruktureStudija_student ADD CONSTRAINT
FK_uess_student_uess FOREIGN KEY(upisaniElementiStruktureStudija_id)
REFERENCES upisaniElementiStruktureStudija(sifra);

ALTER TABLE upisaniElementiStruktureStudija_student ADD CONSTRAINT
FK_upisaniElementiStruktureStudija_student_temeljFinanciranja
FOREIGN KEY(temeljFinanciranja_id) REFERENCES
temeljFinanciranja(sifra);


ALTER TABLE predmet_student ADD CONSTRAINT
FK_predmet_student_student FOREIGN KEY(student_id) REFERENCES
student(jmbag);

ALTER TABLE predmet_student ADD CONSTRAINT
FK_predmet_student_predmet FOREIGN KEY(predmet_id) REFERENCES
predmet(sifra);

ALTER TABLE predmet_student ADD CONSTRAINT
FK_predmet_student_nacinUpisaPredmeta FOREIGN
KEY(nacinUpisaPredmeta_id) REFERENCES nacinUpisaPredmeta(sifra);

ALTER TABLE predmet_student ADD CONSTRAINT
FK_predmet_student_statusUpisanogPredmeta FOREIGN
KEY(statusUpisanogPredmeta_id) REFERENCES
statusUpisanogPredmeta(sifra);
```

**P. 6. 1.**

```
<?php

ini_set('default_charset', 'utf-8');

$servername = "127.0.0.1";

$username = "root";

$password = "97865210";

$database = "db1";


try {

    $conn = new
PDO("mysql:host=$servername;dbname=$database;charset=utf8",
$username, $password);

    // set the PDO error mode to exception

    $conn->setAttribute(PDO::ATTR_ERRMODE, PDO::ERRMODE_EXCEPTION);

    }

catch(PDOException $e)

    {

echo "Connectionfailed: " . $e->getMessage();

    }

?>
```

```
<html>

<head>

<meta charset="utf8" />

</head>

<body>

<?php

require_once('connect.php');

require_once('upiti.php');

?>

<formmethod="POST" action="prikazStudenata.php">

<selectname="predmet">

<?php

$get_predmeti_sql->execute();

$get_predmeti_result = $get_predmeti_sql-
>setFetchMode(PDO::FETCH_ASSOC);

foreach( $get_predmeti_sql->fetchAll() as $predmet ){

echo '<optionvalue="' . $predmet['sifra'] . '">' . $predmet['naziv']
. '</option>';}

?>

</select>

<br />

<selectname="smjer">

<?php

$get_smjerovi_sql->execute();

$get_smjerovi_result = $get_smjerovi_sql-
>setFetchMode(PDO::FETCH_ASSOC);

foreach( $get_smjerovi_sql->fetchAll() as $smjer ){

echo '<optionvalue="' . $smjer['sifra'] . '">' . $smjer['naziv'] .
'</option>';}

?>

</select>

<br />
```

```
<?php

$get_statusi_predmeta_sql->execute();

$get_statusi_predmeta_results = $get_statusi_predmeta_sql-
>setFetchMode(PDO::FETCH_ASSOC);

$i = 0;

foreach( $get_statusi_predmeta_sql->fetchAll() as $status ){

if( $i === 0 )

{

echo '<inputtype="radio" name="status" value="' . $status['naziv'] .
'" / checked><label for="status">'.$status['naziv'].'</label>';

++$i;

}

else

{

echo '<inputtype="radio" name="status" value="' . $status['naziv'] .
'" /><label for="status">'.$status['naziv'].'</label>';

}

}

?>

<br />

<inputtype="submit" value="Pretrazi" />

</form>

</body>

</html>
```

## P. 6. 2.

```
$student_sql = $conn->prepare("INSERT INTO student (jmbag, ime,
prezime) VALUES (:jmbag, :ime, :prezime)");

$upisaniElementStruktureStudija_sql = $conn->prepare("INSERT INTO
upisanielementistrukturestudija (sifra, naziv, paralelni) VALUES
(:sifra, :naziv, :paralelni)");

$temeljFinanciranja_sql = $conn->prepare("INSERT INTO
temeljfinanciranja (sifra, opis, placanje,
nacinIzvedbeStudijaUzKojiSeTemeljFinanaciranjaMozePojaviti) VALUES
(:sifra, :opis, :placanje,
:nacinIzvedbeStudijaUzKojiSeTemeljFinanaciranjaMozePojaviti)");


$upisaniElementiStruktureStudija_student_sql = $conn-
>prepare("INSERT INTO upisaniElementiStruktureStudija_student
(student_id, upisaniElementiStruktureStudija_id,
temeljFinanciranja_id, akademskaGodinaZadnjegUpisnogLista,
nastavnaGodina) VALUES (:student_id,
:upisaniElementiStruktureStudija_id, :temeljFinanciranja_id,
:akademskaGodinaZadnjegUpisnogLista, :nastavnaGodina)");

$predmet_sql = $conn->prepare("INSERT INTO predmet (sifra, naziv,
ectsbodovi, satiUkupno, ulaziUProsjek, semestar, zaPraviloPrijenosa)
VALUES(:sifra, :naziv, :ectsbodovi, :satiUkupno, :ulaziUProsjek,
:semestar, :zaPraviloPrijenosa)");

$ocjenjivac_sql = $conn->prepare("INSERT INTO ocjenjivac (oznaka,
ime, prezime) VALUES(:oznaka, :ime, :prezime)");

$predmet_student_sql = $conn->prepare("INSERT INTO predmet_student
(predmet_id, student_id, nacinUpisaPredmeta_id,
statusUpisanogPredmeta_id) VALUES(:predmet_id, :student_id,
:nacinUpisaPredmeta_id, :statusUpisanogPredmeta_id)");

$nacinUpisaPredmeta_sql = $conn->prepare("INSERT INTO
nacinUpisaPredmeta (sifra, naziv) VALUES(:sifra, :naziv)");

$statusUpisanogPredmeta_sql = $conn->prepare("INSERT INTO
statusUpisanogPredmeta (sifra, naziv) VALUES(:sifra, :naziv)");

$ispit_sql = $conn->prepare("INSERT INTO ispit (predmet_id,
student_id, ocjenjivac_id, datumIspita, datumRoka, ocjena)
VALUES(:predmet_id, :student_id, :ocjenjivac_id,
STR_TO_DATE(:datum_ispita, '%d.%m.%Y.'), STR_TO_DATE(:datum_roka,
'%d.%m.%Y.'), :ocjena)");
```

```
$ocjenjivac_sql = $conn->prepare("INSERT INTO ocjenjivac (oznaka,
ime, prezime) VALUES(:oznaka, :ime, :prezime)");

$student_sql->bindParam(':jmbag', $student_jmbag);

$student_sql->bindParam(':ime', $student_ime);

$student_sql->bindParam(':prezime', $student_prezime);
```

```
$upisaniElementStruktureStudija_sql->bindParam(':sifra',
$upisaniElementStruktureStudija_sifra);

$upisaniElementStruktureStudija_sql->bindParam(':naziv',
$upisaniElementStruktureStudija_naziv);

$upisaniElementStruktureStudija_sql->bindParam(':paralelni',
$upisaniElementStruktureStudija_paralelni);

$temeljFinanciranja_sql->bindParam(':sifra',
$temeljFinanciranja_sifra);

$temeljFinanciranja_sql->bindParam(':opis',
$temeljFinanciranja_opis);

$temeljFinanciranja_sql->bindParam(':placanje',
$temeljFinanciranja_placanje);

$temeljFinanciranja_sql-
>bindParam(':nacinIzvedbeStudijaUzKojiSeTemeljFinanaciranjaMozePojav
iti',
$temeljFinanciranja_nacinIzvedbeStudijaUzKojiSeTemeljFinanaciranjaMo
zePojaviti);
```

```
header('Content-Type: text/plain');

require_once('connect.php');

require_once('queries.php');

ini_set('max_execution_time', 15000);

$data_upisaniElementStruktureStudija_sifra = array();

$data_temeljFinanciranja_sifra = array();

$data_predmet_sifra = array();

$data_nacinUpisaPredmeta = array();

$data_statusUpisanogPredmeta = array();

$data_ocjenjivac = array();


$files = scandir('./studentsumarno');

$n = count($files);

var_dump($n);

try

{for( $i = 2; $i < $n; ++$i)

{

$myXMLData = file_get_contents('./studentsumarno/' . $files[$i]);

$xml = simplexml_load_string($myXMLData) or die("Error:
Cannotcreateobject");

$student_jmbag = $xml->student->jmbag;

$student_ime = $xml->student->ime;

$student_prezime = $xml->student->prezime;
```

```
foreach( $xml->upisaniElementiStruktureStudija-
>upisaniElementStruktureStudija as $a )

{

$temeljFinanciranja_sifra = $a->temeljFinanciranja['sifra'];

$temeljFinanciranja_opis = $a->temeljFinanciranja->opis;

$temeljFinanciranja_placanje = $a->temeljFinanciranja->placanje;

$temeljFinanciranja_nacinIzvedbeStudijaUzKojiSeTemeljFinanaciranjaMo
zePojaviti = $a->temeljFinanciranja-
>nacinIzvedbeStudijaUzKojiSeTemeljFinanaciranjaMozePojaviti['oznaka'
];

if( !in_array( intval($temeljFinanciranja_sifra),
$data_temeljFinanciranja_sifra) )

{

$data_temeljFinanciranja_sifra[] =
intval($temeljFinanciranja_sifra);

$temeljFinanciranja_sql->execute();

}

$upisaniElementStruktureStudija_sifra = $a['sifra'];

$upisaniElementStruktureStudija_paralelni = $a['paralelni'];

$upisaniElementStruktureStudija_naziv = $a->naziv;

if( !in_array( intval($upisaniElementStruktureStudija_sifra),
$data_upisaniElementStruktureStudija_sifra) )

{

$data_upisaniElementStruktureStudija_sifra[] =
intval($upisaniElementStruktureStudija_sifra);

$upisaniElementStruktureStudija_sql->execute();

}

$upisaniElementiStruktureStudija_student_sql-
>bindParam(':student_id', $student_jmbag);
```

```
$upisaniElementiStruktureStudija_student_sql-
>bindParam(':upisaniElementiStruktureStudija_id',
$upisaniElementStruktureStudija_sifra);

$upisaniElementiStruktureStudija_student_sql-
>bindParam(':temeljFinanciranja_id', $temeljFinanciranja_sifra);

$upisaniElementiStruktureStudija_student_sql-
>bindParam(':akademskaGodinaZadnjegUpisnogLista', $a-
>akademskaGodinaZadnjegUpisnogLista);

$upisaniElementiStruktureStudija_student_sql-
>bindParam(':nastavnaGodina', $a->nastavnaGodina);

$upisaniElementiStruktureStudija_student_sql->execute();
```

```
                        /*
                        NE POLOZENI PREDMETI
                        */
foreach($a->popisNepolozenihPredmeta->nepolozenPredmet as $b)
{
$predmet_sifra = $b->predmet['sifra'];
$predmet_naziv = $b->predmet->naziv;
$predmet_ectsbodovi = $b->predmet->ectsBodovi;
$predmet_satiUkupno = $b->predmet->satiUkupno;
$predmet_ulaziUProsjek = $b->predmet->ulaziUProsjek;
$predmet_semestar = $b->predmet->semestar['redniBroj'];
$predmet_zaPraviloPrijenosa = $b->predmet->zaPraviloPrijenosa;
if( !in_array( intval($predmet_sifra), $data_predmet_sifra) )
{
$data_predmet_sifra[] = intval($predmet_sifra);
$predmet_sql->bindParam(':sifra', $predmet_sifra);
$predmet_sql->bindParam(':naziv', $predmet_naziv);
$predmet_sql->bindParam(':ectsbodovi', $predmet_ectsbodovi);
$predmet_sql->bindParam(':satiUkupno', $predmet_satiUkupno);
$predmet_sql->bindParam(':ulaziUProsjek', $predmet_ulaziUProsjek);
$predmet_sql->bindParam(':semestar', $predmet_semestar);
$predmet_sql->bindParam(':zaPraviloPrijenosa',
$predmet_zaPraviloPrijenosa);
$predmet_sql->execute();}
$predmet_nacinUpisaPredmeta_sifra = $b->predmet-
>nacinUpisaPredmeta['sifra'];$predmet_nacinUpisaPredmeta_naziv = $b-
>predmet->nacinUpisaPredmeta;
```

```
if( !in_array( intval($predmet_nacinUpisaPredmeta_sifra),
$data_nacinUpisaPredmeta) )

{

$data_nacinUpisaPredmeta[] =
intval($predmet_nacinUpisaPredmeta_sifra);

$nacinUpisaPredmeta_sql->bindParam(':sifra',
$predmet_nacinUpisaPredmeta_sifra);

$nacinUpisaPredmeta_sql->bindParam(':naziv',
$predmet_nacinUpisaPredmeta_naziv);

$nacinUpisaPredmeta_sql->execute();}

$predmet_statusUpisanogPredmeta_sifra = $b->predmet-
>statusUpisanogPredmeta['sifra'];

$predmet_statusUpisanogPredmeta_naziv = $b->predmet-
>statusUpisanogPredmeta;

if( !in_array( intval($predmet_statusUpisanogPredmeta_sifra),
$data_statusUpisanogPredmeta) )

{

$data_statusUpisanogPredmeta[] =
intval($predmet_statusUpisanogPredmeta_sifra);

$statusUpisanogPredmeta_sql->bindParam(':sifra',
$predmet_statusUpisanogPredmeta_sifra);

$statusUpisanogPredmeta_sql->bindParam(':naziv',
$predmet_statusUpisanogPredmeta_naziv);

$statusUpisanogPredmeta_sql->execute();

}

$predmet_student_sql->bindParam('predmet_id', $predmet_sifra);

$predmet_student_sql->bindParam('student_id', $student_jmbag);

$predmet_student_sql->bindParam('nacinUpisaPredmeta_id',
$predmet_nacinUpisaPredmeta_sifra);

$predmet_student_sql->bindParam('statusUpisanogPredmeta_id',
$predmet_statusUpisanogPredmeta_sifra);

$predmet_student_sql->execute();

}
```

```
/*

                            POLOZENI PREDMETI

                        */

foreach($a->popisPolozenihPredmeta->polozenPredmet as $c)

{

$predmet_sifra = $c->predmet['sifra'];

$predmet_naziv = $c->predmet->naziv;

$predmet_ectsbodovi = $c->predmet->ectsbodovi;

$predmet_satiUkupno = $c->predmet->satiUkupno;

$predmet_ulaziUProsjek = $c->predmet->ulaziUProsjek;

$predmet_semestar = $c->predmet->semestar['redniBroj'];

$predmet_zaPraviloPrijenosa = $c->predmet->zaPraviloPrijenosa;

if( !in_array( intval($predmet_sifra), $data_predmet_sifra) )

{

$data_predmet_sifra[] = intval($predmet_sifra);

$predmet_sql->bindParam(':sifra', $predmet_sifra);

$predmet_sql->bindParam(':naziv', $predmet_naziv);

$predmet_sql->bindParam(':ectsbodovi', $predmet_ectsbodovi);

$predmet_sql->bindParam(':satiUkupno', $predmet_satiUkupno);

$predmet_sql->bindParam(':ulaziUProsjek', $predmet_ulaziUProsjek);

$predmet_sql->bindParam(':semestar', $predmet_semestar);

$predmet_sql->bindParam(':zaPraviloPrijenosa',
$predmet_zaPraviloPrijenosa);

$predmet_sql->execute();}
```

```
$predmet_nacinUpisaPredmeta_sifra = $c->predmet-
>nacinUpisaPredmeta['sifra'];

$predmet_nacinUpisaPredmeta_naziv = $c->predmet->nacinUpisaPredmeta;

if( !in_array( intval($predmet_nacinUpisaPredmeta_sifra),
$data_nacinUpisaPredmeta) )

{     $data_nacinUpisaPredmeta[] =
intval($predmet_nacinUpisaPredmeta_sifra);

      $nacinUpisaPredmeta_sql->bindParam(':sifra',
$predmet_nacinUpisaPredmeta_sifra);

      $nacinUpisaPredmeta_sql->bindParam(':naziv',
$predmet_nacinUpisaPredmeta_naziv);

      $nacinUpisaPredmeta_sql->execute();}
```

```
$predmet_statusUpisanogPredmeta_sifra = $c->predmet-
>statusUpisanogPredmeta['sifra'];

$predmet_statusUpisanogPredmeta_naziv = $c->predmet-
>statusUpisanogPredmeta;

if( !in_array( intval($predmet_statusUpisanogPredmeta_sifra),
$data_statusUpisanogPredmeta) )

{$data_statusUpisanogPredmeta[] =
intval($predmet_statusUpisanogPredmeta_sifra);

$statusUpisanogPredmeta_sql->bindParam(':sifra',
$predmet_statusUpisanogPredmeta_sifra);

$statusUpisanogPredmeta_sql->bindParam(':naziv',
$predmet_statusUpisanogPredmeta_naziv);

$statusUpisanogPredmeta_sql->execute();}

$predmet_student_sql->bindParam('predmet_id', $predmet_sifra);

$predmet_student_sql->bindParam('student_id', $student_jmbag);

$predmet_student_sql->bindParam('nacinUpisaPredmeta_id',
$predmet_nacinUpisaPredmeta_sifra);

$predmet_student_sql->bindParam('statusUpisanogPredmeta_id',
$predmet_statusUpisanogPredmeta_sifra);

$predmet_student_sql->execute();

if( !$c->ispit->ocjenjivac )

{continue;

}

$ispit_ocjenjivac_oznaka = $c->ispit->ocjenjivac->oznaka;

$ispit_ocjenjivac_ime = $c->ispit->ocjenjivac->ime;

$ispit_ocjenjivac_prezime = $c->ispit->ocjenjivac->prezime;

if( !in_array( (string) $ispit_ocjenjivac_oznaka, $data_ocjenjivac)
){

$data_ocjenjivac[] = (string) $ispit_ocjenjivac_oznaka;

$ocjenjivac_sql->bindParam(':oznaka', $ispit_ocjenjivac_oznaka);

$ocjenjivac_sql->bindParam(':ime', $ispit_ocjenjivac_ime);

$ocjenjivac_sql->bindParam(':prezime', $ispit_ocjenjivac_prezime);

$ocjenjivac_sql->execute();}
```

```
$ispit_datumIspita = $c->ispit['datumIspita'];

$ispit_datumRoka = $c->ispit->datumRoka;

$ispit_ocjena = $c->ispit->ocjena;

$ispit_sql->bindParam(':predmet_id', $predmet_sifra);

$ispit_sql->bindParam(':student_id', $student_jmbag);

$ispit_sql->bindParam(':ocjenjivac_id', $ispit_ocjenjivac_oznaka);

$ispit_sql->bindParam(':datum_ispita', $ispit_datumIspita);

$ispit_sql->bindParam(':datum_roka', $ispit_datumRoka);

$ispit_sql->bindParam(':ocjena', $ispit_ocjena);

$ispit_sql->execute();

}

}

}}

catch(PDOException $e)

{

echo 'error: ' . $e->getMessage();

}

$conn = null;
```

**P. 6. 3.1.**

```
$get_predmeti_sql = $conn->prepare("SELECT * FROM  predmet");

$get_smjerovi_sql = $conn->prepare("SELECT * FROM
upisanielementistrukturestudija");

$get_statusi_predmeta_sql = $conn->prepare("SELECT * FROM
statusupisanogpredmeta");

$get_predmet_sql = $conn->prepare("SELECT naziv FROM predmet WHERE
sifra = :sifra_predmeta");

$get_smjer_sql = $conn->prepare("SELECT naziv FROM
upisanielementistrukturestudija WHERE sifra = :sifra_smjer");

$get_predmet_nepolozen_sql = $conn->prepare("SELECT s.jmbag AS
jmbag, CONCAT(s.ime, ' ', s.prezime) AS student FROM student s

JOIN predmet_student ps ON ps.student_id=s.jmbag

JOIN predmet p ON p.sifra=ps.predmet_id

JOIN statusupisanogpredmeta sup ON
sup.sifra=ps.statusUpisanogPredmeta_id

JOIN upisanielementistrukturestudija_student uess_student ON
uess_student.student_id=s.jmbag

JOIN upisaniElementiStruktureStudija uess ON
uess.sifra=uess_student.upisaniElementiStruktureStudija_id

WHERE p.sifra=:predmet_sifra AND sup.naziv=:status_predmeta  AND
uess.sifra=:smjer

ORDER BY uess.naziv");

$get_predmet_polozen_sql = $conn->prepare("SELECT s.jmbag AS jmbag,
CONCAT(s.ime, ' ', s.prezime) AS student, i.ocjena AS
ocjena,i.datumIspita  AS datumIspita, i.datumRoka  AS datumRoka,
CONCAT(oc.ime, ' ', oc.prezime) AS ocjenjivac FROM student s
```

```
JOIN predmet_student ps ON ps.student_id=s.jmbag

JOIN predmet p ON p.sifra=ps.predmet_id

JOIN statusupisanogpredmeta sup ON
sup.sifra=ps.statusUpisanogPredmeta_id

JOIN upisanielementistrukturestudija_student uess_student ON
uess_student.student_id=s.jmbag

JOIN upisaniElementiStruktureStudijauess ON
uess.sifra=uess_student.upisaniElementiStruktureStudija_id

JOIN ispit i ON s.jmbag=i.student_id

JOIN ocjenjivacoc ON i.ocjenjivac_id=oc.oznaka

WHERE p.sifra=:predmet_sifra AND sup.naziv=:status_predmeta  AND
uess.sifra=:smjer

ORDER BY uess.naziv");
```

**P. 6. 3. 2.**

```
<html>
<head>
<meta charset="utf8" />
<style>
body {
background-color: linen;
}
h1 {
color: maroon;
margin-left: 40px;
}
table trtd{
border: 1px solidblack;
}
table{
border: 2px solidblack;}
</style>
</head>
<body>
<?php
require_once('connect.php');
require_once('upiti.php');
$predmet_sifra = $_POST['predmet'];
$smjer_sifra = $_POST['smjer'];
$predmet_status = $_POST['status'];
```

```
$get_predmet_sql->bindParam(':sifra_predmeta', $predmet_sifra);

$get_predmet_sql->execute();

$predmet = $get_predmet_sql->fetchColumn();

$get_smjer_sql->bindParam(':sifra_smjer', $smjer_sifra);

$get_smjer_sql->execute();

$smjer = $get_smjer_sql->fetchColumn();

?>

<h1><?phpecho $smjer . ': ' . $predmet; ?></h1>

<table>

<tr>

<td>JMBAG</td>

<td>Student</td>

<?php

if( $predmet_status === 'obavljen (položen)' ){

$get_predmet_polozen_sql->bindParam(':predmet_sifra',
$predmet_sifra);

$get_predmet_polozen_sql->bindParam(':smjer', $smjer_sifra);

$get_predmet_polozen_sql->bindParam(':status_predmeta',
$predmet_status);

$get_predmet_polozen_sql->execute();

$get_predmet_polozen_results = $get_predmet_polozen_sql-
>setFetchMode(PDO::FETCH_ASSOC);

echo '<td>Ocjena</td><td>Datum Ispita</td><td>Datum
roka</td><td>Ocjenjivac</td</tr>';

foreach( $get_predmet_polozen_sql->fetchAll() as $polozen_predmet ){

$datum = date_create($polozen_predmet['datumIspita']);     $rok =
date_create($polozen_predmet['datumRoka']);     echo '<tr><td>' .
$polozen_predmet['jmbag'] . '</td><td>' .
$polozen_predmet['student'] . '</td><td>' .
$polozen_predmet['ocjena'] . '</td>';echo '<td>' .
date_format($datum, 'd.m.Y.') . '</td><td>'. date_format($rok,
'd.m.Y.') .'</td><td>' . $polozen_predmet['ocjenjivac'] .
'</td></tr>';}
```

```
}

else

{

$get_predmet_nepolozen_sql->bindParam(':predmet_sifra',
$predmet_sifra);

$get_predmet_nepolozen_sql->bindParam(':smjer', $smjer_sifra);

$get_predmet_nepolozen_sql->bindParam(':status_predmeta',
$predmet_status);

$get_predmet_nepolozen_sql->execute();

$get_predmet_nepolozen_result = $get_predmet_nepolozen_sql-
>setFetchMode(PDO::FETCH_ASSOC);

echo '</tr>';

foreach( $get_predmet_nepolozen_sql->fetchAll() as
$nepolozen_predmet ) {

echo '<tr><td>' . $nepolozen_predmet['jmbag'] . '</td><td>' .
$nepolozen_predmet['student'] . '</td></tr>';

}

}

?>

</table>

</body>

</html>
```