

SVEUČILIŠTE U ZAGREBU

GEODETSKI FAKULTET

Dino Šubat

IZRADA APLIKACIJE ZA UPRAVLJANJE JAVNOM RASVJETOM NA PODRUČJU GRADA

CRIKVENICE

Diplomski rad



ZAGREB, srpanj 2011.

Zahvala:

Zahvaljujem se svome mentoru prof. dr. sc. Damiru Medaku, te voditeljima Draženu Odobašiću i Mariu Mileru na pruženoj pomoći, strpljenju i razumijevanju prilikom pisanja ovog diplomskog rada.

Hvala gradu Crikvenici na ustupljenim podacima za izradu diplomskog rada.

Također veliko hvala i mojoj obitelji, djevojci, te svim prijateljima i kolegama bez kojih bi studiranje bilo nezamislivo.

## I. Autor

Ime i prezime: Dino Šubat

Datum i mjesto rođenja: 26.05.1987., Rijeka, Hrvatska

## **II. Diplomski rad**

Predmet: Programiranje u geoinformacijskim sustavima

Naslov: Izrada aplikacije za upravljanje javnom rasvjetom na području grada Crikvenice

Mentor: prof. dr. sc. Damir Medak

Voditelji: D.Odobašić, M.Miler

### **III. Ocena i obrana**

Datum zadavanja zadatka:

Datum obrane: 13.07.2011.

Sastav povjerenstva pred kojim je branjen rad: prof. dr. sc. Damir Medak

prof. dr. sc Drago Špoljarić

dr.sc. Ivan Medved

## **Sažetak**

Gotovo sve što se događa, događa se negdje u prostoru. Stoga je geografski položaj važan atribut. Geografski informacijski sustavi su posebna vrsta informacijskih sustava koji ne prate samo događaje, već i lokaciju na kojoj se ti događaji događaju. Znati gdje se nešto događa ponekad može biti od ključne važnosti.

Ovaj rad ukratko opisuje geoinformacijske sustave, prostorne baze podataka, te postupak izrade GIS aplikacije za upravljenje sustavom javne rasvjete na području grada Crikvenice.

Aplikacija je izrađena s ciljem lakšeg i učinkovitijeg upravljanja sustavom javne rasvjete.

## **Abstract**

Almost everything that happens, happens somewhere. Therefore geographic location is an important attribute. Geographic information systems are a special class of information systems that keep track not only of events but also of where these events happen or exist. Knowing where something happens can be critically important.

This paper briefly describes the geoinformation systems, spatial databases, and the process of creating GIS application for managing the public lighting system in the town of Crikvenica.

The application is designed to facilitate the efficient management of public lighting system.

## Sadržaj

1.	Uvod .....	7
2.	Što je GIS?.....	8
3.	Povijesni pregled GIS-a.....	12
4.	Podaci u GIS-u .....	16
4.1	Vektorski podaci .....	18
4.1.1	Struktura vektorskih podataka.....	18
4.1.1.1	Špagetna struktura podataka.....	18
4.1.1.2	Topologija.....	19
4.1.1.2.1	Mrežna topologija .....	19
4.1.1.2.2	Plošna topologija.....	19
4.2	Rasterski podaci .....	20
4.2.1	Rezolucija slike .....	21
4.2.1.1	Spektralna rezolucija .....	21
4.2.1.2	Radiometrijska rezolucija .....	22
4.2.1.3	Prostorna rezolucija .....	23
4.2.1.4	Vremenska rezolucija .....	23
4.3	Usporedba rastera i vektora.....	23
5.	Prikupljanje podataka.....	25
6.	Pohrana podataka u bazu podataka .....	27
6.1	Povijest PostgreSQL-a.....	27
6.1.1	Berkeley POSTGRES projekt .....	27
6.1.2	Postgres95 .....	28
6.1.3	PostgreSQL .....	28
6.2	PostGIS .....	30
6.3	Početni unos podataka u bazu podataka .....	31
7.	Izrada aplikacije za upravljanje javnom rasvjetom .....	34
7.1	Microsoft Visual Studio .....	35
7.1.1.	Net Framework.....	36
7.2	Izgled Visual Studia 2010.....	37
7.2.1	Toolbox.....	38
7.2.2	Server Explorer .....	39
7.2.3	Solution Explorer.....	39

7.2.4 Properties .....	40
7.3 Windows projekt .....	41
7.4 Windows kontrole .....	41
7.2 Izrada aplikacije .....	44
7.2.1 ArcGIS .....	44
7.2.2 Razvijanje korisničke ArcReader aplikacije s ArcGIS Publisher ekstenzijom .....	45
7.2.3 Forma "gis_preglednik" .....	49
7.2.4 Forma "stupovi" .....	51
7.5.4.1 Unos podataka .....	51
7.5.4.2 Označavanje kvara .....	54
7.5.4.3 Izrada izvješća .....	55
7.5.4.4 Geotagging .....	58
7.5.5 Forma "kvarovi" .....	60
7.5.6 Forma "pretraga" .....	61
7.5.6.1 Google Earth .....	62
7.5.6.1.1 Dodavanje vektorskih slojeva u Google Earth .....	63
7.5.6.1.2 Dodavanje rasterskih slojeva u Google Earth .....	64
7.5.6.2 Izrada *.kml datoteke .....	65
8. Zaključak .....	67

## 1. Uvod

Prije šire primjene računala, informacije u vezi sa zemljištem bile su iscrtane na prozirnim slojevima koje je bilo moguće preklopiti s topografskom kartom. Tako je npr. plan urbanog područja, osim ulica i kuća, trebao sadržavati i plinovode, vodovode, električne i telefonske kabele, itd. Izravno ucrtavanje svih navedenih tema na osnovni plan brzo bi ga učinio nepreglednim pa je svaki sustav iscrtavan na zasebnome prozirnom sloju, koji se zatim, prema potrebi, mogao preklopiti s osnovnim planom. Dodatno je bilo potrebno čuvati veliki broj pisanih podataka o svim vodovima i čvoristima (materijal, kapacitet, vrijeme izrade, vlasništvo, vlasništvo nad zemljištem ispod kojeg vod prolazi). To je analogni zemljišni informacijski sustav činilo vrlo opsežnim i nepraktičnim. [2]

Na području grada Crikvenice postoji ukupno 2984 stupova javne rasvjete. Iz navedenog broja stupova očito je da je voditi evidenciju o tolikom broj stupova puno lakše uz pomoć računalne aplikacije izrađene upravo za tu namjenu nego u analognom ili u digitalnom obliku (koristeći aplikaciju koja nije primarno izrađena za upravljanje podacima javne rasvjete).

Cilj ovog diplomskog rada bio je izraditi samostalnu aplikaciju za evidenciju svih objekata javne rasvjete na području grada Crikvenice. Razlog zbog kojeg je odlučeno izraditi ovakvu aplikaciju je omogućiti lakše vođenje evidencije o objektima javne rasvjete. Naime, grad Crikvenica posjeduje GIS aplikaciju u kojoj je sadržan veliki broj prostornih podataka, a samim time i podaci o javnoj rasvjeti, no aplikacija nije najpogodnija za uspješno upravljanje sustavom javne rasvjete.

## 2. Što je GIS?

Na samom početku postavlja se pitanje što je zapravo GIS? Kratica GIS dolazi od naziva *Geografski Informacijski Sustav* ili *GeoInformacijski Sustav*. GIS je računalni sustav za prihvatanje, upravljanje, povezivanje, rukovanje, analiziranje i prikazivanje geopodataka, odnosno geoinformacija. Prema ESRI-jevom (*engl. Environmental Systems Research Institute*) rječniku GIS pojmove GIS čine:

- Hardver – uređaji koji fizički izvršavaju određene GIS operacije (računalo, digitalizator, ploter...)
- Softver – skup naredbi računalu radi izvršavanja GIS operacija (upravljanje podacima, prostorne analize i vizualizacije)
- Podaci – ključni element GIS-a. Sadrže eksplicitne prostorne reference (koordinate) ili implicitne prostorne reference (adresa, poštanski broj, broj statističkog kruga, ime ulice...)
- Ljudi – najaktivnija komponenta – bave se dizajnom, programiranjem, posluživanjem i upravljanjem GIS-om
- Procedure – vezane uz upravljački aspekt GIS-a, važne radi osiguravanja visoke kvalitete GIS-a (kontrola kvalitete, načini izvještavanja, kontrolne točke...)
- Mreža – omogućava brzu komunikaciju i razmjenu digitalnih informacija (Internet)

Ovo je vrlo općenit opis za tako kompleksni i široki set alata. U suštini, GIS je centralno skladište i analitički alat za geografske podatke prikupljenje iz različitih izvora. [3] GIS nije samo jednostavni grafički zaslon za prikaz prostornih podataka ili njihovih atributa, nego sustav koji ih visoko razvijenim funkcijama kombinira radi podrške upravljanju i donošenju odluka. Prvi i najvažniji korak je prikupljanje i unos podataka: GIS je upravo toliko kvalitetan koliko i uneseni podaci. Podaci se u računalu ne samo sortiraju, nego se također indeksiraju i organiziraju za brz i kontrolirani pristup. Podaci se moraju strukturirati da bi se osigurala njihova pouzdanost, sigurnost i cjelovitost. GIS osigurava

vezu između prostornih i neprostornih podataka, omogućujući složenu analizu cjelovitog skupa podataka. [2]

Upotreba GIS-a čini analogue (papirnate) karte i odgovarajuću dokumentaciju nepotrebнима. Postupci obrade podataka, kako prostornih tako i neprostornih, neusporedivo su brži: podatke je lako ažurirati i prenijeti na druga umrežena računala. Velika prednost je i mogućnost brzog prijenosa podataka većem broju korisnika. Uz pomoć GIS-a mogu se simulirati različiti scenariji i na temelju rezultata olakšati donošenje kvalitetnih odluka. Kako GIS omogućuje i dobru vizualizaciju prostornih podataka i njihovih međusobnih odnosa, odluke se mogu planirati i pouzdano donositi.

Područje geografskih informacijskih sustava danas se pojavljuje kao jedno od najnaprednijih tehničkih područja. GIS tehnologija se razvila tijekom protekla četiri desetljeća i tek se sada počela intenzivnije koristiti. Sa širokom lepezom korisnika, postoji ne samo značajan interes za GIS, nego i određen stupanj nerazumijevanja ove nove tehnologije. U GIS-u se događa transformacija geografskih podataka u geografske informacije. Naizgled jednostavna transformacija obuhvaća vrlo kompleksne skupove funkcija i procesa. Ukratko, geografski podaci nastaju kao sirovi položajni podaci koji sadrže attribute. Kasnije se podaci preklapaju sa komplementarnim i/ili kontrastnim skupovima podataka, koji onda zajedno čine koencidentne odnose. Podaci i odnosi se analiziraju, geoprocесiraju i tek tada prezentiraju kao produkti geografskih informacija. Ovi produkti geoinformacija su često interaktivne računalne aplikacije koje se koriste za pomoć u donošenju odluka. [3]

Vrste GIS programa su:

- *Professional GIS* – omogućuje prikupljanje podataka i njihovo ispravljanje, administriranje geoprostorne baze podataka, napredne metode obrade prostornih podataka i njihove analize, te ostale specijalizirane alate (*ESRI ArcInfo, SmallWorld...*)
- *Desktop GIS* - osnovna namjena je korištenje postojećih podataka, a ne njihovo stvaranje. Služe za izradu karata, izvještaja i dijagrama (*ESRI ArcView, Intergraph GeoMedia, MapInfo Professional, Clark Lab's Idrisi...*)

- *Hand-held GIS* – sustavi primjereni korištenju u pokretu (*Autodesk Onsite, ESRI ArcPad, Smallworld Scout...*)
- *Componenet GIS* - biblioteke funkcija i ostalih alata, koje služe programerima za razvoj aplikacija specifične, ciljane primjene (*BlueMarble Geographic GeoObjects, MapInfo MapX...*)
- *GIS viewer* - omogućavaju pregled i pretraživanje podataka u standardnim formatima zapisa (*ESRI ArcReader, ESRI ArcExplorer, Intergraph GeoMedia Viewer, MapInfo ProViewer...*)
- *Internet GIS* - usmjeren ka prikazu i pretraživanju prostornih podataka (*Autodesk MapGuide, ESRI ArcIMS, Intergraph GeoMedia Web Map, MapInfo MapXtreme, Google Earth...*) [1]

Geoinformacijski sustavi primjenjivi su svugdje gdje se traži modeliranje, analiza i upravljanje podacima koji se tiču prostora. U nastavku ćemo opisati nekoliko primjera korištenja GIS-a.

- Upravljanje instalacijama: organizacije koje vode brigu o plinovodima, vodovodima, električnim, telefonskim i drugim vodovima odgovorne su za veliki broj vodova, kanala, građevina i zemljišta, koji zahtijevaju nadzor, održavanje i upravljanje kako bi uspješno udovoljili korisnicima.

U ovu kategoriju spada i upravljanje sustavom javne rasvjete, što je i predmet izrade ovog diplomskog rada.

- Održavanje cesta: slično prethodnom primjeru, s tom razlikom što je ovdje riječ o cestama, autocestama, mostovima, vijaduktima, tj. objektima povezanim s prostorom koji zahtijevaju održavanje i upravljanje. Trodimenzionalni modeli terena mogu biti korisni pri projektiranju i proračunu utjecaja na okoliš.
- Stambena izgradnja i iznajmljivanje: odgovornost za gradnju, održavanje, iznajmljivanje i prodaju stambenih objekata. Važan je prostorni raspored objekata, ali i svi raspoloživi detalji o njima. Operativni podaci uključuju

vrste kuća, vrste vlasništva, učestalost i cijene prodaje i iznajmljivanja, populacijske trendove itd. Terenski agenti za nekretnine nisu više nužni, kvalitetna vizualna prezentacija prostornih podataka povećava produktivnost i osigurava objektivnost.

Očito je da se tamo gdje su potrebni različiti tipovi podataka koji se tiču prostornih odnosa GIS nameće kao nezamjenjivi alat. U nastavku ćemo prikazati kratki razvoj GIS-a i geoinformatike kroz povijest. [2]

### 3. Povijesni pregled GIS-a

Prije 3500 godina kromanjonski lovci nacrtali su prvi GIS na pećini u blizini Lascaux-a u Francuskoj. Povezali su životinje sa migracijskim rutama. Ovakav GIS sadrži dva osnovna elementa suvremenog GIS-a. Grafiku povezani s atributnim podacima (slika 1 i 2). [1]



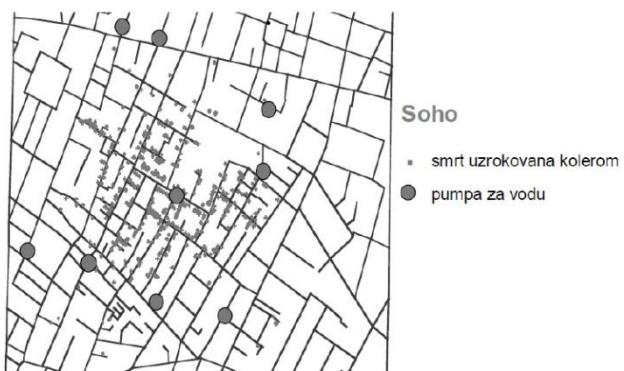
Slika 1. Jedan od prvih GIS-eva



Slika 2. Jedan od prvih GIS-eva

1854.g. doktor John Snow učinio je nešto revolucionarno. Pretvorivši dostupne geopodatke u geoinformacije hrabro je zaustavio koleru. Naime, od 1831. do sredine 1850-tih London je trpio smrtonosnu epidemiju kolere. Dr. John Snow označio je smrtonosne ishode kolere na ručno nacrtanoj karti ulica Londona. Uskoro mu je postalo jasno da se koncentracija smrtnih slučajeva grupira oko pumpe za vodu na Broad Street-u

(slika 3). Odmah je prezentirao svoje otkriće nadležnima te je ručka pumpe uklonjena. Odmah nakon toga, prestalo je širenje kolere. [3]



Slika 3. Položaji mesta na kojima su pronađene umrle osobe, te položaji pumpa za vodu

GIS kakav danas poznajemo nastao je u Kanadi 1962. Godine, u čekaonici zračne luke u Ottawi slučajnim susretom Rogera Tomlinsona i Leeja Pratta. Tomlinson je bio bivši pilot RAF-a, koji je krajem pedesetih emigrirao u Kanadu i radio kao geograf u tvrtki *Spartan Air Lines*. Ta tvrtka se komercijalno bavila snimanjima iz zraka i puno je uložila u opremu. Među prvima se koristila računalima za različite poslove; od planiranja troškova, planiranja letova do obrade fotografija. Pratt je bio viši državni službenik i radio je u *Canada Land Inventoryju (CLI)*, vladinoj instituciji za prikupljanje podataka o vrstama i korištenju tla. Čekajući avione, dvojica stručnjaka našla su zajedničku temu razgovora - izradu tematskih karata. Pratt se požalio kako je pred njima ogroman posao kartiranja oko milijun kvadratnih milja zemljišta (skoro 2.600.000 km<sup>2</sup>): po vrstama tla, naseljenosti i načinu korištenja. Tomlinson, koji je u svom odjelu koristio računalo, primjetio je kako bi se to moglo riješiti programske. Njih dvojica su se srdačno rastali razmijenivši telefonske brojeve. Kad se Pratt vratio u CLI, dao je izračunati koliki će resursi trebati za dovršenje projekta. Brojke su bile obeshrabrujuće i Pratt se sjetio Tomlinsonove primjedbe o primjeni računala. Nazvao ga je ponudivši mu odmah posao u CLI-ju. Tomlinson je

prihvatio i već sljedeće godine dovršio je računalni sustav koji je kombinirao bazu podataka i kartografiju. Nazvao ga je *Canadian Geographic Information System* (CGIS).

Bio je to prvi sustav ikad nazvan "geoinformacijski sustav". Sadržavao je nekoliko aplikacija od kojih je glavna pohranjivala digitalnu kartu i podatke o terenu u standardnom formatu za cijelu Kanadu. U odnosu na današnje geoinformacijske sustave bio je jednostavan: mogao je pohranjivati podatke, preslagivati atribute, mijenjati mjerilo, spajati podatke kreirajući nove poligone, te kreirati izvještaje. "Kanadski GIS" trajao je do 1990-ih, te je izgradio najveću digitalnu bazu podataka o zemljишnim resursima u Kanadi. Njegov početni razvoj i uspjeh potaknuo je različite komercijalne primjene kartiranja, koje su prodavali prodavači poput Intergrapha. Uspjeh CGIS-a pokrenuo je lavinu sličnih projekata. [10]

Amerikanci 1963. godine formiraju *Urban and Regional Information Systems* (URISA) u Illinoisu – neprofitablinu organizaciju za informatičko rješavanje problema u planiranju, javnim radovima, okolišu itd.

Na Sveučilištu Harvard osniva se 1964. godine harvardski laboratorij za računalnu grafiku i prostorne analize pod vodstvom profesora Howarda Fishera. Fisher je vodio razvoj softvera *Synagraphic Mapping System* (SYMAP), a 1966. godine počeo je predavati GIS kao kolegij.

Ured za statistiku SAD-a prihvata novu tehnologiju, razvija se *Dual Independent Map Encoding* (DIME) format podataka. U isto vrijeme Britanci osnivaju *Experimental Cartography Unit* na Royal College of Art u Londonu.

Novi korak u razvoju GIS tehnologije bio je uspostavljanje *Urban and Regional Information Systems Association* (URISA) 1963. godine, neprofitabilne organizacije za informatičko rješavanje problema u planiranju, javnim radovima, okolišu itd. [1]

1967. godine američka centralna obavještajna agencija (CIA) razvila AUTOMAP

1969. godine prve GIS tvrtke ESRI i Intergraph

1972. godine – 23.7. lansiran prvi (ERTS-1) Landsat-1 satelit sa RBV senzorom

1978. godine lansirana prva 4 GPS satelita

1979. godine harvardski laboratorij razvio ODYSSEY GIS – prvi vektorski GIS

1980. godine Dana Tomlison razvila MAP, popularni rasterski GIS

1982. godine ESRI razvio ARC/INFO; francuska svemirska agencija osnovala SPOT Image – prva komercijalna tvrtka za distribuciju geografskih informacija dobivenih sa Zemljinih satelita

1987. godine izašao prvi broj časopisa IJGS (International Journal of Geographic Information Systems)

1988. godine američki federalni ured za statistiku dao na javnu upotrebu TIGER digitalne podatke

1993. godine Steve Putz razvio prvu internetsku, interaktivnu kartu

1999. godine prva proslava GIS dana

2006. godine više od 15000 ljudi iz više od 20 zemalja prisustvovalo je 26-toj godišnjoj konferenciji ESRI-ja [1]

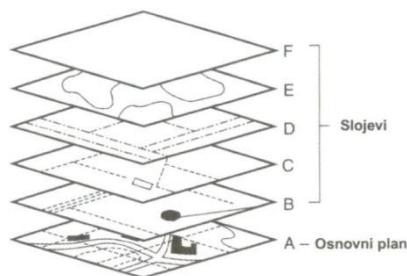
## 4. Podaci u GIS-u

U GIS-u se pojavljuju različiti tipovi podataka:

- grafički
- tekstualni
- animacije
- slike
- video zapisi
- zvučni zapisi

Iz nabrojanih tipova podataka očigledno je kako geoinformacijski sustavi sve više postaju multimedijalna tehnologija. Tu podjelu možemo malo pojednostaviti i reći kako se GIS sastoje od grafičkih i atributnih podataka. Grafički podaci su u vektorskom ili rasterskom formatu. Geografski podaci dolaze iz različitih izvora podataka, kao što su digitalizirane karte, zračne snimke, podaci dobiveni GPS-om, te podaci dobiveni raznim metodama geodetske izmjere. Pravilno odabrana struktura geopodataka smatra se glavnim razlogom učinkovitosti ili neučinkovitosti nekog GIS sustava. Nepravilno upravljanje podacima ili kriva struktura podataka zasigurno vode k neprikladnim GIS proizvodima.

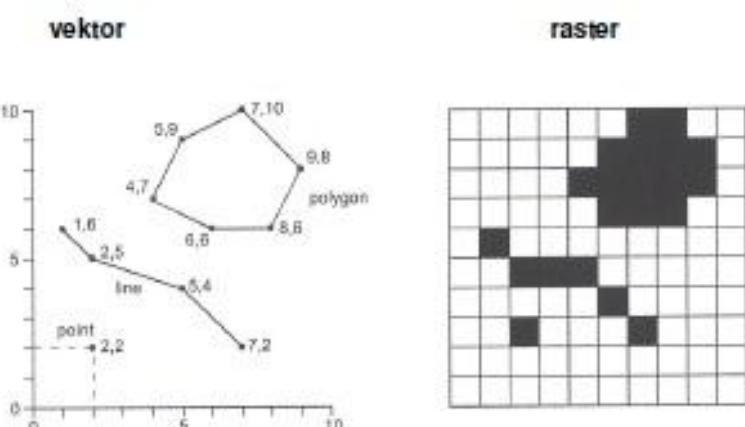
U svim geoinformacijskim sustavima podaci su podijeljeni u slojeve. Takva podjela omogućuje da podaci budu prikazani odvojeno zadržavajući mogućnost usporedbe i analize podataka iz različitih slojeva. Svi slojevi moraju biti povezani i odnositi se na isto područje da bi se mogli točno preklopiti (slika 4). [2]



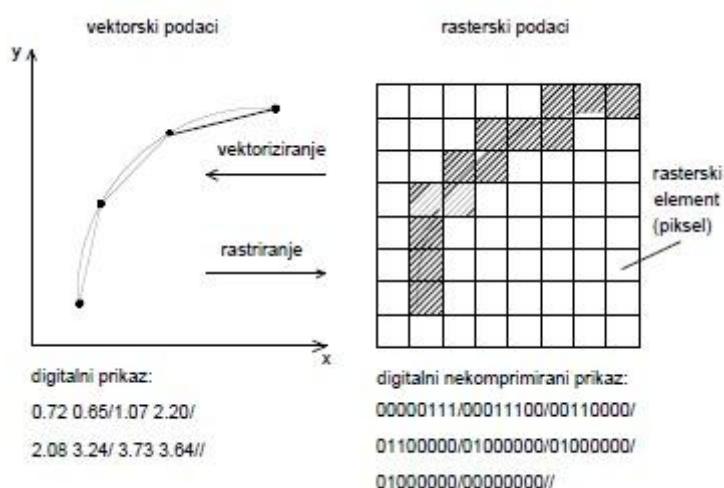
Slika 4. Slojevi u GIS-u

Slojevi podataka pohranjeni su u GIS-u uporabom jednog od dva tipa podataka poznatih kao vektor i raster. Oba modela podataka upotrebljavaju koordinatni sustav kako bi pohranili položaje objekata. Razlika im je u tome što biraju različite načine za prikaz oblika i veličine objekata. Potrebno je znati odabrati između dva tipa podataka: vektorskog i rasterskog (slika 5. i slika 6.), jer različiti softveri koriste različite tipove podataka. [1] Nekad su GIS softveri koristili jedan ili drugi format. Suvremeni softveri omogućuju konverziju podataka između formata, odnosno imaju mogućnost prikaza vektorskih podataka na rasterskoj pozadini. [2]

#### **Modeli prikaza objekata u GISu:**



Slika 5. Vektorski i rasterski tip podataka



Slika 6. Prikaz linije u vektorskem i rasterskom obliku

## 4.1 Vektorski podaci

Vektor je vrijednost s početnim koordinatama, pridruženim pomakom i smjerom. Pri opisivanju prostornih podataka pomoću vektora, pretpostavljamo kako se element može nalaziti na bilo kojoj lokaciji, bez položajnih ograničenja koje nameće raster. [1] Vektorska struktura podataka bazirana je na osnovnom točkastom elementu čija je lokacija poznata do izabrane točnosti. Vektorski prikaz se upotrebljava kada je potrebna veća položajna točnost.

Neke od prednosti vektorskih podataka nad rasterskim su: dobar prikaz objekata, kompaktnost modela, mogućnost primjene mrežne topologije, geometrijska preciznost, mogućnost generalizacije, jednostavnost uređivanja, manji obim podataka, brži pristup podacima.

Neki od nedostataka vektorske strukture podataka su sama složenost strukture podataka, kombiniranja su vremenski dugotrajna, tehnologija je još uvijek skupa (softver).

### 4.1.1 Struktura vektorskih podataka

Vektorski podaci pojavljuju se pri različitim tehnikama prikupljanja podataka. Prostorni podaci mogu se organizirati u dvije vrste strukture: špageti i topologija.

#### 4.1.1.1 Špagetna struktura podataka

Svaki prostorni objekt opisan je koordinatama i parametarskim jednadžbama (pravac, kružnica, krivulja itd.) Često se naziva i CAD struktura jer je vrlo učinkovita za dizajn i kartografiju. Struktura špageta je vrlo ograničena za proučavanje odnosa između objekata jer pojedini objekt ne ovisi o svom susjedu. Zajedničke stranice poligona su udvostručene,a lukovi se mogu sjeći bez presijecanja, mogućnost nezavršenih područja, dvostruko digitalizirane i slične granice između susjednih područja. Podaci se slažu nalik špagetima. Špagetna struktura podataka se upotrebljava kako bi opisali podatke digitalne karte sa objektima koji se presijecaju.

#### **4.1.1.2 Topologija**

Topologija je grana matematike koja se bavi prostornim odnosima geometrijskih oblika, tj. relativnim odnosima između pojedinačnih objekata. U GIS-u se topologija koristi za zapisivanje i rukovanje odnosima između objekata. Postoji mrežna i plošna topologija [2]. Značajni topološki koncepti su:

- Susjedstvo: na kojoj strani linije se nalazi cesta?
- Povezanost: koje je točke nužno spojiti da bi cesta bila definirana?
- Orientacija: koja je točka početna, a koja završna?
- Pripadnost: koje kuće leže unutar zadatog poligona?

##### **4.1.1.2.1 Mrežna topologija**

Struktura podataka koja opisuje veze između različitih lukova koji čine mrežu. Koristi se za pretraživanje (npr. traženje optimalnog puta i slično). Lukovi su orijentirani, a odnosi (početak i kraj) mogu se spremiti na dva načina: topologija lukova – svaki luk zna svoju početnu i završnu točku, te topologija čvorova – svaki čvor poznae početne i završne lukove.

##### **4.1.1.2.2 Plošna topologija**

Plošna topologija je struktura podataka koja opisuje odnose između različitih poligona stvarajući sloj koji se može analizirati i kombinirati s drugim. Obično također sadrži mrežu topologije kod koje svaki luk nosi informacije o poligonima s desne i lijeve strane.

## 4.2 Rasterski podaci

Rasterska struktura podataka karakterizira kontinuirane podatke (npr. slike) i posebno je prikladna za slučajeve gdje informacije o točkama i granicama nisu strogo definirane. Rasterska struktura podataka bazirana je na rasterizaciji karte. U rasterskoj strukturi podataka za prikaz prostornih podataka, naša sposobnost da definiramo položaj u prostoru ograničena je veličinom piksela. Rasterska struktura podataka jedna je od najjednostavnijih struktura podataka. Raster je geografski podatak u kojem je vrijednost pridružena četverokutnom području objekta. U rasterskom prikazu ne razlikujemo točke, linije, površine, tj. ne postoji samo logička veza između slikovnih elemenata, već samo svojstva pojedinog piksela (npr. siva tonska vrijednost).

Rasterski tip podataka prikazuje objekt podijeljen u niz jedinica gdje svaka jedinica prikazuje jednak dio objekta. Jedinice najčešće koriste četverokutni oblik i nazivaju se polja. Polja su organizirana u redove i stupce te se nazivaju grid ili mreža. Mrežu karakterizira ishodište, orijentacija i veličina rasterskog polja. Područje je prekriveno mrežom polja i čini matricu. Prikaz objekata baziran je na elementu matrice (2D ili 3D). Polje u dvodimenzionalnom području se naziva piksel, a u trodimenzionalnom području voksel. Geometrija takvog elementa (mreža točaka ili piksel) dana je redom ili stupcem koji definira taj element, pomakom u odnosu na prvi gornji lijevi piksel i rezolucijom mreže. Geometrijska točnost je ograničena rezolucijom polja. Područje je prekriveno mrežom piksela. Rezolucija rasterske slike ovisi o veličini pixela.

Rasterska struktura podataka ne omogućava preciznu informaciju o položaju jer je prostor podijeljen u diskretna područja, odnosno površina je podijeljena u jednake četverokute. U rasterskoj strukturi podataka, točnost karte ovisi o rezoluciji. Područje koje svaki piksel predstavlja varira od nekoliko centimetara do nekoliko kilometara i naziva se rezolucija mreže. Što je viša rezolucija mreže, više celija je potrebno za predstavljanje danog područja (slika 7). [1]



Slika 7. Odnos rezolucije i broja piksela

#### 4.2.1 Rezolucija slike

Rezoluciju slike možemo definirati kao sposobnost slikovnog sustava da prepozna razdvojeno različite detalje na slici. Razlikujemo četiri osnovne rezolucije slika. To su: spektralna rezolucija, radiometrijska rezolucija, prostorna rezolucija i vremenska rezolucija. [1]

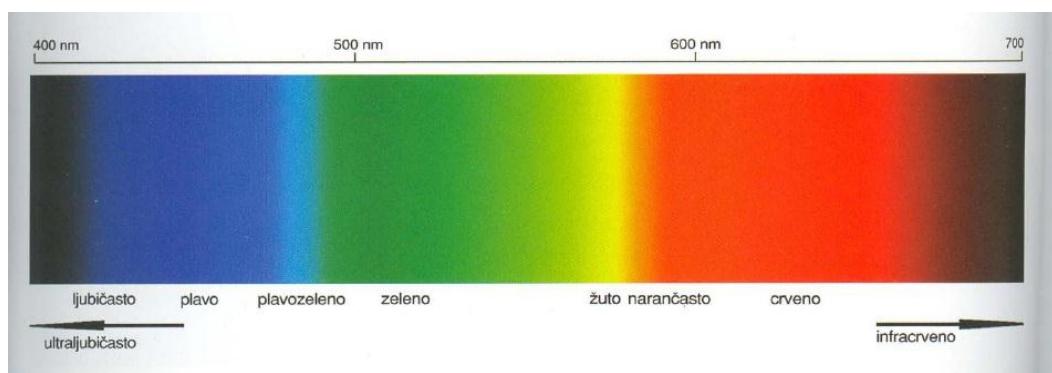
##### 4.2.1.1 Spektralna rezolucija

Odnosi se na raspon spektralnih boja, tj. broj različitih zapisanih frekvencijskih pojaseva – obično je ovaj broj jednak broju senzora na platformi. Različiti materijali na zemlji različito reflektiraju svjetlost. Te spektralne karakteristike definiraju spektralni položaj i spektralnu osjetljivost vezanu za određeni materijal. [1]

Klasični spektar razlikuje sedam boja:



Slika 8. Klasični spektar sa sedam boja



Slika 9. Uporaba cijelog niza spektralnih boja je neophodna kako bi različite ciljeve pravilno interpretirali na slici dobivenoj daljinskim istraživanjima

#### 4.2.1.2 Radiometrijska rezolucija

Broj različitih intenziteta radijacije koje senzor može razlikovati. Uglavnom je prikazana kao binarni broj neophodan za pohranu sloja maksimalne vrijednosti. Tipičan raspon ide od 8 do 14 bita što odgovara 256 razina skale sive boje, te 16.384 intenziteta ili

"sjenki" boje u svakom pojasu. Podaci dobiveni Landsat-om podijeljeni su u 256 slojeva što je predstavljeno s 8 bit-a. [1]

#### **4.2.1.3 Prostorna rezolucija**

Prostornu rezoluciju definira veličina piksela koji se zapisuje u rasterskom snimku. Možemo reći da je prostorna rezolucija mogućnost otkrivanja najmanjeg prostornog objekta na površini Zemlje. Pikseli tipično mogu odgovarati kvadratnim područjima čija veličina stranice može biti od nekoliko centimetara do nekoliko kilometara. [1]

#### **4.2.1.4 Vremenska rezolucija**

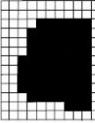
Vremenska rezolucija predstavlja vremenski interval između dva uzastupna snimka nad istim područjem, tj. odgovara frekvenciji preleta satelitom ili zrakoplovom, te je relevantna samo u istraživanjima vremenskih slijedova ili u onima koji zahtijevaju prosječan ili mozaičan snimak kao npr. u nadgledanju deforestacije. [1]

### **4.3 Usporedba rastera i vektora**

Glavna razlika kod vektorskog i rasterskog tipa podataka je razlika u zauzeću memorije prilikom njihova pohranjivanja. Naime vektorska struktura podataka zauzima bitno manje memorijskog prostora. Što su dimenzije rasterskog elementa kod rastera manje, to je za rasterski model potrebno više memorijskog prostora.

Često se postavlja pitanje kada koristiti rastere, kada vektore, a kada oboje? Ovo pitanje je danas puno manje važno nego što je bilo u prošlosti. Mnoge moderne GIS aplikacije imaju mogućnosti rada i s rasterskim i s vektorskим podacima, te mogu raditi konverziju između rastera i vektora. Ipak, navesti ćemo neke glavne razlike između ove dvije strukture podataka:

- rasterski podaci zauzimaju puno više memorije od vektorskih podataka
- računanje sa rasterima je brže od računanja sa vektorima
- rasteri su manje prostorno točni od vektora
- rasterski podaci daju nam više opcija za rad s plohami od vektorskih
- rasteri su više moćniji za modeliranje od vektora
- rasteri su više kompatibilni sa tehnologijom printer-a, a manje kompatibilni sa tehnologijom plotera, dok je kod vektora slučaj obrnut

element	vektorski		rasterski	
	digitalni	analogni	digitalni	analogni
točka	koordinate x,y	.	piksel	
linija	niz koordinata x,y		piksel	
površina	zatvoren niz koordinata x,y		piksel	

Slika 10. Razlika između zapisa vektorskog i rasterskog modela podataka

## 5. Prikupljanje podataka

Podaci o objektima javne rasvjete dobiveni su od strane grada Crikvenice za cijelo područje grada (pod područje grada Crikvenice spadaju još mjesta Selce, Dramalj i Jadranovo), pa tako nije bilo potrebno izlaziti na teren kako bi se podaci prikupili. Prikupljanje podataka grad Crikvenica je naručio od tvrtke Muraj d.o.o. koja je podatke prikupila i isporučila ih gradu u obliku \*.xls tablica.

Podaci su od strane grada Crikvenice ustupljeni na korištenje u dva formata i to: pohranjeni u Microsoft Excel-ovoj proračunskoj tablici, te u MapInfo \*.tab formatu. Naime, GIS grada Crikvenice temelji se na MapInfo arhitekturi. No, iz razloga što nismo posjedovali licencu za MapInfo korišteni su podaci spremljeni u \*.xls datotekama.

Primjer podataka u \*.xls datoteci je dan u tablici 1.

OZNAKA	STUP	SVJETILJKA	SIJALICA	TS	ULICA	NAPAJANJE	KO E	KO N
C1	metal	Syteco veća	NaVT 250W	Kaskada	Kralja	kabelsko	5475379	5004780
C2	metal	Gamalux 06	NaVT 70W	Kaskada	Put mora	kabelsko	5475358	5004772
C3	metal	Gamalux 06	NaVT 70W	Kaskada	Put mora	kabelsko	5475339	5004762
C4	metal	Syteco veća	NaVT 250W	Kaskada	Kralja	kabelsko	5475432	5004838
C5	metal	Syteco veća	NaVT 250W	Kaskada	Kralja	kabelsko	5475424	5004808
C6	metal	Syteco veća	NaVT 250W	Kaskada	Kralja	kabelsko	5475420	5004776
C7	metal	Syteco veća	NaVT 250W	Kaskada	Kralja	kabelsko	5475424	5004751
C8	metal	Syteco veća	NaVT 250W	Kaskada	Kralja	kabelsko	5475413	5004754
C9	metal	Syteco veća	NaVT 250W	Kaskada	Kralja	kabelsko	5475400	5004774
C10	metal	Syteco veća	NaVT 250W	Kaskada	Kralja	kabelsko	5475399	5004746
C11	metal	Syteco veća	NaVT 250W	Kaskada	Tina Ujevića	kabelsko	5475390	5004753
C12	metal	Syteco veća	NaVT 250W	Kaskada	Tina Ujevića	kabelsko	5475392	5004727
C13	metal	2x	2xNaVT	Kaskada	Tina Ujevića	kabelsko	5475390	5004689
C14	metal	Magistrala	WTF 250W	Kaskada	Vukovarska	kabelsko	5475358	5004711

Tablica 1. Prikaz podataka dobivenih u \*.xls datoteci

Podaci su bili pohranjeni u ukupno četiri \*.xls datoteke i to "selce.xls" (499 zapisa), "crikvenica.xls" (1466 zapisa), "dramalj.xls" (584 zapisa), "jadranovo.xls" (435 zapisa). Svaka datoteka sadržavala je sljedeće stupce: oznaka, stup, svjetiljka, sijalica, trafostanica, ulica, napajanje, koordinate E (istočno), koordinate N (sjeverno).

Položajni podaci, tj. koordinate stupova, prikupljeni su pomoću ručnog GPS uređaja u WGS84 koordinatnom sustavu. Koordinate su transformirane u HDKS/GK5 koordinatni sustav pomoći transformacijskih parametara za Primorsko-goransku županiju

(DGU) i iskazane su na metar. Točnost određivanja položaja svakog stupa pomoću ručnog GPS je oko 5-7 metara.

Bitno je naglasiti da ukoliko bi se željelo postići bolju točnost pozicioniranja treba koristiti kvalitetnije, geodetske, uređaje te samim time za posao prikupljanja prostornih podataka i izradu GIS-a uposliti geodetsku tvrtku kojoj je ovaj posao daleko jasniji i bliži.

## 6. Pohrana podataka u bazu podataka

Kako podaci pohranjeni u \*.xls datotekama nisu pogodni za geoprostornu vizualizaciju, te kako je pretpostavljeno da bi podacima moglo istovremeno pristupati više osoba s više različitih računala, odlučeno je da ćemo podatke pohraniti u prostornu bazu podataka. Rad s podacima pohranjenim u bazi podatka ima mnogo prednosti, a višekorisnički pristup je samo jedna od njih.

Baza podataka koju smo odlučili koristiti je Postgres baza podataka s instaliranim Postgis prostornim proširenjem. Zbog svoje liberalne licence, PostgreSQL može biti korišten, mijenjan i distribuiran od strane bilo koga i to besplatno za bilo koju potrebu, bila ona privatna, komercijalna ili akademska.

Nad podacima pohranjenim u ovakvoj bazi podataka možemo osim klasičnih, postavljati i prostorne upite, transformirati koordinate, dobivati podatke u \*.kml obliku, svg obliku itd. U nastavku ćemo opisati Postgresql bazu podataka kao i Postgis prostorno proširenje Postgresql baze podataka.

### 6.1 Povijest PostgreSQL-a

Objektno relacijski sustav za upravljanje bazom podataka poznat kao PostgreSQL je razvijen iz POSTGRES paketa napisanog na Berkeley-u, Sveučilište u Kaliforniji. S preko dva desetljeća razvijanja PostgreSQL je danas najnaprednija dostupna open-source baza podataka.

#### 6.1.1 Berkeley POSTGRES projekt

Implementacija POSTGRES-a je započela 1986. godine. POSTGRES je od tada doživio nekoliko glavnih izdanja. Prvo demo izdanje je zaživjelo 1987.g., a javnosti je prikazano 1988. g.. Verzija 1 je izdana za nekoliko vanjskih korisnika u lipnju 1989.g. Verzija 2 je izdana u lipnju 1990. Verzija 3 se pojavila 1991.

POSTGRES je korišten za implementaciju mnogih različitih istraživačkih i produkcijskih aplikacija. Također je korišten kao edukacijski alat na nekoliko sveučilišta. Konačno, *Illustra Information Technologies* (kasnije spojene u Informix, a koji je danas u vlasništvu IBM-a) je uzeo kod i komercijalizirao ga. U kasnoj 1992, POSTGRES je postao primarni sustav za upravljanje podacima *Sequoia 2000* znanstvenog računalnog projekta.

Broj vanjskih korisnika se približno udvostručio 1993.g. Postalo je očito da održavanje koda i podrška oduzimaju previše vremena koje se trebalo posvetiti istraživanju baze podataka. S ciljem da se smanji teret podrške, Berkeley POSTGRES projekt je službeno završio s verzijom 4.2.

### 6.1.2 Postgres95

1994. g. Andrew Yu i Jolly Chen dodali su interpreter SQL jezika. Pod novim imenom, Postgres95 je pušten na webu da sam nađe svoj put i to kao open-source potomak originalnog POSTGRES Berkeley koda.

Postgres95 kod bio je u cijelosti ANSI C i smanjene veličine za 25%. Mnoge promjene koda poboljšale su performanse i održavanje koda. Postgres95 verzija 1.0.x bila je prema *Wisconsin Benchmark-u* 30-50% brža u odnosu na POSTGRES verziju 4.2. Osim ispravljanja bugova dodano je nekoliko novih poboljšanja.

### 6.1.3 PostgreSQL

1996. godine postalo je jasno je da naziv „Postgres95“ neće biti dobar za buduće verzije. Odabran je novi naziv, PostgreSQL, iz kojega se vidi povezanost između originalnog POSTGRES-a i novih mogućnosti SQL-a. U isto vrijeme odlučeno je da će verzije početi od broja 6.0, kako bi se nastavilo na originalne verzije koje su započele s Berkeley POSTGRES projektom.

Mnogi su nastavili nazivati PostgreSQL kao "Postgres" (rjeđe pisano velikim slovima) iz razloga naviknutosti ili zbog lakšeg izgovaranja. Ovo korištenje naziva "Postgres" je široko prihvaćeno kao nadimak ili alias.

Naglasak tijekom razvoja Postgres-a 95 je bio na identificiranju i razumijevanju postojećih problema u poslužiteljskom kodu. Kod PostgreSQL-a naglasak je pomaknut na dorađivanje mogućnosti, ali se rad nastavlja u svim područjima.

## 6.2 PostGIS

PostGIS dodaje podršku za geografske objekte PostgreSQL objektno relacijskoj bazi podataka. PostGIS prostorno proširuje PostgreSQL server i to tako da se on koristi kao prostorna baza podataka za geografske informacijske sustave (GIS), slično kao što to rade *ESRI SDE* ili *Oracle Spatial* prostorna ekstenzija.

PostGIS slijedi OpenGIS-ovu specifikaciju ("Simple Features Specification for SQL"), te je certificiran kao sukladan s tipovima i funkcijama.

PostGIS je razvijen od strane *Refractions Research-a* kao projekt u open source tehnologiji prostorne baze podataka. PostGIS je objavljen pod uvjetima *GNU General Public licence*. PostGIS se razvija i dalje. Budući projekti će uključivati: potpunu topološku podršku, podršku za rastere, mreže, trodimenzionalne plohe, krivulje i spline-ove.

Neke funkcije PostGIS-a su:

- geometrijski tipovi za točku, linestring-ove, poligone, multi točke, multi linestring-ove, multi poligone i kolekciju geometrija
- prostorni predikati za određivanje odnosa geometrije koristeći 3x3 Egenhofer-ovu matricu (omogućeno korištenjem GEOS modula)
- prostorni operatori za određivanje geoprostornih mjerena kao što su površina, udaljenost, duljina i opseg
- prostorni operatori za određivanje skupa operacija kao što su unija, razlika, simetrična razlika i buffer (također omogućeno korištenjem GEOS modula)
- *R-tree-over-GiST (Generalised Search Tree)* – prostorni indeksi za brzo pretraživanje geometrije

Postoji veliki broj aplikacija koje mogu koristiti PostGIS bazu podataka. Neke od njih su: *Cadcorp SIS*, *Everest GIS*, *Feature Manipulation Engine*, *GeoMedia*, *GeoServer*, *GRASS GIS*, *gvSIG*, *ESRI-jeva ekstenzija za interoperabilnost*, *Ionic Red Spider*, *Kosmo*,

*Manifold System, MapInfo Professional, Mapnik, MapDotNet Server, MapServer, MapGuide, MezoGIS, OpenJUMP, Quantum GIS, TerraLib, TerraView, uDig, ArcGIS, CitySurf Globe.*

### 6.3 Početni unos podataka u bazu podataka

Na samome početku bilo je potrebno na najlakši mogući način unijeti sve već postojeće podatke u bazu podataka. Ovdje nije dolazio u obzir unos podataka za jedan po jedan stup koristeći izrađenu aplikaciju jer bi to oduzelo previše vremena (ukupno 2984 stupova. Kasniji unosi, promjene nad podacima i eventualna brisanja zapisa iz baze podataka mogu se izvršavati kroz izrađenu aplikaciju. Prije samog unosa podataka u bazu, potrebno je kreirati bazu podataka s pripadajućim tablicama.

Baza podataka je za potrebe ovog diplomskog rada instalirana na virtualnom računalu, naziva se "*rasvjeta*" i sastoji se od četiri tablice. Nazivi tablica su: *crikvenica*, *dramalj*, *jadranovo*, *selce*. Nakon instalacije Postgis proširenja na bazu podataka, dodane su još sljedeće dvije tablice: *geometry\_columns* i *spatial\_ref\_sys*. Tablice *crikvenica*, *dramalj*, *jadranovo* i *selce* tako sadrže sljedeće stupce: *gid*, *the\_geom*, *stup*, *svjetiljka*, *sijalica*, *ts*, *oznaka*, *ulica*, *napajanje*, *kvar*, *opis\_kvara*.

Prije samog unosa geometrijskih podataka bilo je potrebno definirati koordinatni sustav u kojem će podaci biti pohranjeni u prostornoj bazi podataka. Koordinatni sustav se definira u *spatial\_ref\_sys* tablici.

U nastavku ćemo prikazati definiciju korištenog koordinatnog sustava.

```
srid = "99995"  
  
auth_name ="dsubat"  
  
auth_srid = "99995"  
  
srtext = "PROJCS["HDKS5",GEOGCS["HDKS5",DATUM["HR1901",SPHEROID["Bessel 1841",  
6377397.155,299.1528128],TOWGS84[528.6481,119.5964,467.1873,6.061424,2.749619,-  
9.525554,-1.064918]],PRIMEM["Greenwich",  
0.0],UNIT["degree",0.017453292519943295],AXIS["Geodetic longitude", EAST],AXIS["Geodetic  
latitude", NORTH]],PROJECTION["Transverse Mercator"],PARAMETER["central_meridian",  
15.0],PARAMETER["latitude_of_origin", 0.0],PARAMETER["scale_factor",  
0.9999],PARAMETER["false_easting", 5500000.0],PARAMETER["false_northing", 0.0],UNIT["m",  
1.0],AXIS["Easting", EAST],AXIS["Northing", NORTH]]"  
  
proj4text = "+proj=tmerc +ellps=bessel +k_0=0.9999 +x_0=5500000 +lon_0=15  
+towgs84=528.6481,119.5964,467.1873,6.061424,2.749619,-9.525554,-1.064918 +no_defs"
```

Unos podataka iz \*.xls datoteke u bazu podataka izveden je kreirajući skriptu u Microsoft Excel-u. Primjer skripte dan je u nastavku:

```
= "INSERT INTO crikvenica (oznaka, stup, svjetiljka, sijalica, ts, ulica, napajanje,the_geom)  
VALUES ('" & A2 & "','" & B2 & "','" & C2 & "','" & D2 & "','" & E2 & "','" & F2 & "','" & G2 &  
"',ST_GeomFromText('POINT(" & H2 & " " & I2 & ")',99995)) "
```

što za prvi zapis iz \*.xls datoteke "crikvenica.xls" rezultira sljedećom SQL naredbom:

```
INSERT INTO crikvenica (oznaka, stup, svjetiljka, sijalica, ts, ulica, napajanje,the_geom)  
VALUES ('C1','metal 11m','Syteco veća','NaVT 250W','Kaskada','Kralja  
Tomislava','kabelsko',ST_GeomFromText('POINT(5475379 5004780)', 99995))
```

Sve kreirane SQL naredbe kopirane su zajedno u *pgAdminIII* (desktop program za upravljanje PostgreSQL bazom podataka) *SQL window* te su pokrenute i izvršene nad bazom podataka (slika 11.). Na ovaj način su u bazu podataka uneseni podaci za svaku od četiri \*.xls datoteke.

```

INSERT INTO crikvenica (oznaka, stup, svjetiljka, sijalica, ts, ulica, napajanje, the_geom) VALUES ('C1','metal l1m','Syt eco veća','NaVT 250W');
INSERT INTO crikvenica (oznaka, stup, svjetiljka, sijalica, ts, ulica, napajanje, the_geom) VALUES ('C2','metal 5,5m','Gamalux 06','NaVT 70W');
INSERT INTO crikvenica (oznaka, stup, svjetiljka, sijalica, ts, ulica, napajanje, the_geom) VALUES ('C3','metal 5,5m','Gamalux 06','NaVT 70W');
INSERT INTO crikvenica (oznaka, stup, svjetiljka, sijalica, ts, ulica, napajanje, the_geom) VALUES ('C4','metal l1m','Syt eco veća','NaVT 250W');
INSERT INTO crikvenica (oznaka, stup, svjetiljka, sijalica, ts, ulica, napajanje, the_geom) VALUES ('C5','metal l1m','Syt eco veća','NaVT 250W');
INSERT INTO crikvenica (oznaka, stup, svjetiljka, sijalica, ts, ulica, napajanje, the_geom) VALUES ('C6','metal l1m','Syt eco veća','NaVT 250W');
INSERT INTO crikvenica (oznaka, stup, svjetiljka, sijalica, ts, ulica, napajanje, the_geom) VALUES ('C7','metal l1m','Syt eco veća','NaVT 250W');
INSERT INTO crikvenica (oznaka, stup, svjetiljka, sijalica, ts, ulica, napajanje, the_geom) VALUES ('C8','metal l1m','Syt eco veća','NaVT 250W');
INSERT INTO crikvenica (oznaka, stup, svjetiljka, sijalica, ts, ulica, napajanje, the_geom) VALUES ('C9','metal l1m','Syt eco veća','NaVT 250W');
INSERT INTO crikvenica (oznaka, stup, svjetiljka, sijalica, ts, ulica, napajanje, the_geom) VALUES ('C10','metal l1m','Syt eco veća','NaVT 250W');
INSERT INTO crikvenica (oznaka, stup, svjetiljka, sijalica, ts, ulica, napajanje, the_geom) VALUES ('C11','metal l1m','Syt eco veća','NaVT 250W');
INSERT INTO crikvenica (oznaka, stup, svjetiljka, sijalica, ts, ulica, napajanje, the_geom) VALUES ('C12','metal l1m','Syt eco veća','NaVT 250W');
INSERT INTO crikvenica (oznaka, stup, svjetiljka, sijalica, ts, ulica, napajanje, the_geom) VALUES ('C13','metal l1m luk','2x Magistrala','2xNe');
INSERT INTO crikvenica (oznaka, stup, svjetiljka, sijalica, ts, ulica, napajanje, the_geom) VALUES ('C14','metal l1m luk','Magistrala','WTF 2');
INSERT INTO crikvenica (oznaka, stup, svjetiljka, sijalica, ts, ulica, napajanje, the_geom) VALUES ('C15','metal l1m luk','Magistrala','WTF 2');
INSERT INTO crikvenica (oznaka, stup, svjetiljka, sijalica, ts, ulica, napajanje, the_geom) VALUES ('C16','metal 5,5m','Elypsø','NaVT 70W');
INSERT INTO crikvenica (oznaka, stup, svjetiljka, sijalica, ts, ulica, napajanje, the_geom) VALUES ('C17','metal 5,5m','Elypsø','NaVT 70W');

```

Slika 11. Primjer SQL skripte za unos podataka u bazu podataka

## 7. Izrada aplikacije za upravljanje javnom rasvjetom

Izrada programa na elektroničkom računalu je sustavan posao koji od čovjeka zahtijeva dobro poznavanje problema koji se rješava, računala na kojemu se izrađuje program i programskega jezika koji se pritom upotrebljava. Proces izrade programa naziva se programiranje, a čovjek koji programira programer. Programiranje je, dakle, složen posao i od programera zahtijeva dobru i temeljitu pripremu. Zbog toga se obično dijeli na više faza. Faze programiranja su:

- analiza problema
- kodiranje programa
- ispitivanje i ispravljanje grešaka
- dokumentiranje
- održavanje programa.

Analiza problema prvi je korak u njegovu rješavanju i potrebno je izvesti temeljitu analizu problema koji je pred programerom. Cilj analize je dobivanje potankog opisa i precizne definicije problema. U toj se fazi preciziraju početni (ulazni) podaci, rezultati koji se očekuju (izlazni) podaci kao i postupci kojima se dobivaju takvi rezultati. Postupci kojima se od ulaznih podataka dobivaju konačni rezultati obrade nazivaju se algoritmom obrade.

Kodiranje programa je proces zapisa algoritma u nekome programskom jeziku.

Ispitivanje i ispravljanje grešaka je proces koji podrazumijeva ispravljanje svih grešaka u kodiranju programa, a zatim veći broj izvođenja programa nad različitim polaznim podacima. U toj fazi potrebno je uočiti ima li program grešaka i daje li dobre (očekivane) rezultate. Svaku uočenu grešku potrebno je ispraviti, a ukoliko se ne dobivaju očekivani rezultati, potrebno je vratiti se u prvu fazu analize problema i ponoviti je. Zatim treba ponovno kodirati program prije te faze.

Dokumentiranje je vrlo važna faza u izradi programa, jer za cijeli program valja opisati što i kako raditi te dati naputak korisniku kako se koristiti programom. To je vrlo bitno jer korisnik programa obično nije programer već čovjek koji s programiranjem nije dosta upoznat. Kako bi korisnici znali koji problem program rješava te što su ulazni podaci, kako se unose i slično, potrebno je vrlo temeljito dokumentirati program. Dokumentiranje je potrebno obaviti i zbog samih programera koji izvode posljednju fazu u programiranju – održavanje programa.

Održavanje programa je posljednja faza i predstavlja naknadna poboljšanja i prepravke programa koji se rade kad je program već u uporabi pa se uoče neki nedostaci.  
[5]

Sama aplikacija za upravljanje sustavom javne rasvjete izrađena je u *VisualBasic-u*.  
.Net koristeći *Microsoft Visual Studio 2010*. U nastavku ćemo prikazati neke osnovne karakteristike *VisualStudio 2010*.

## 7.1 Microsoft Visual Studio

*Microsoft Visual Studio* je integrirana razvojna okolina (engl. *Integrated Development Environment - IDE*) razvijena od strane Microsofta. Može se koristiti za razvijanje konzolnih aplikacija kao i za razvoj aplikacija s grafičkim korisničkim sučeljem. Moguće je razvijati aplikacije za slijedeće platforme: *Microsoft Windows*, *Windows Mobile*, *Windows CE*, *.NET Framework*, *.NET Compact Framework* i *Microsoft Silverlight*.

Jezici koji se mogu koristiti u *Visual Studiu .NET* su *Visual Basic*, *C#*, *C++*, *F#*, a postoje još i mnogi drugi koji su prilagođeni za *.NET framework* (*COBOL*, *Pearl*, *Eiffel*, *Python*, *Pascal*...). Važna je međusobna suradnja različitih jezika kroz *.NET framework*. Ako je netko napisao neku klasu u npr. *C++-u*, ona sa i dalje može iskoristiti u *C#-u* ili *Visual Basic-u*. [9]

### 7.1.1. Net Framework

.NET framework je platforma koja omogućava razvoj aplikacija. Zahtjevi koje ispunjava su:

- konzistentno objektno-orientirano okruženje koje omogućava izvršavanje koda, lokalno izvršavanje i distribuirano preko Interneta ili udaljeno
- okruženje koje minimizira distribuciju softvera i konflikte među verzijama
- okruženje koje omogućava sigurno izvršavanje koda, koji je generiran od nepoznate strane
- eliminira probleme performansi skriptnih jezika i jezika sa interpreterima
- programerima nudi konzistentno okruženje od *Windows* do *web* aplikacija
- sva komunikacija je ostvarena na industrijskim standardima kako bi kod baziran na .NET Framework mogao komunicirati sa bilo kojim drugim kodom

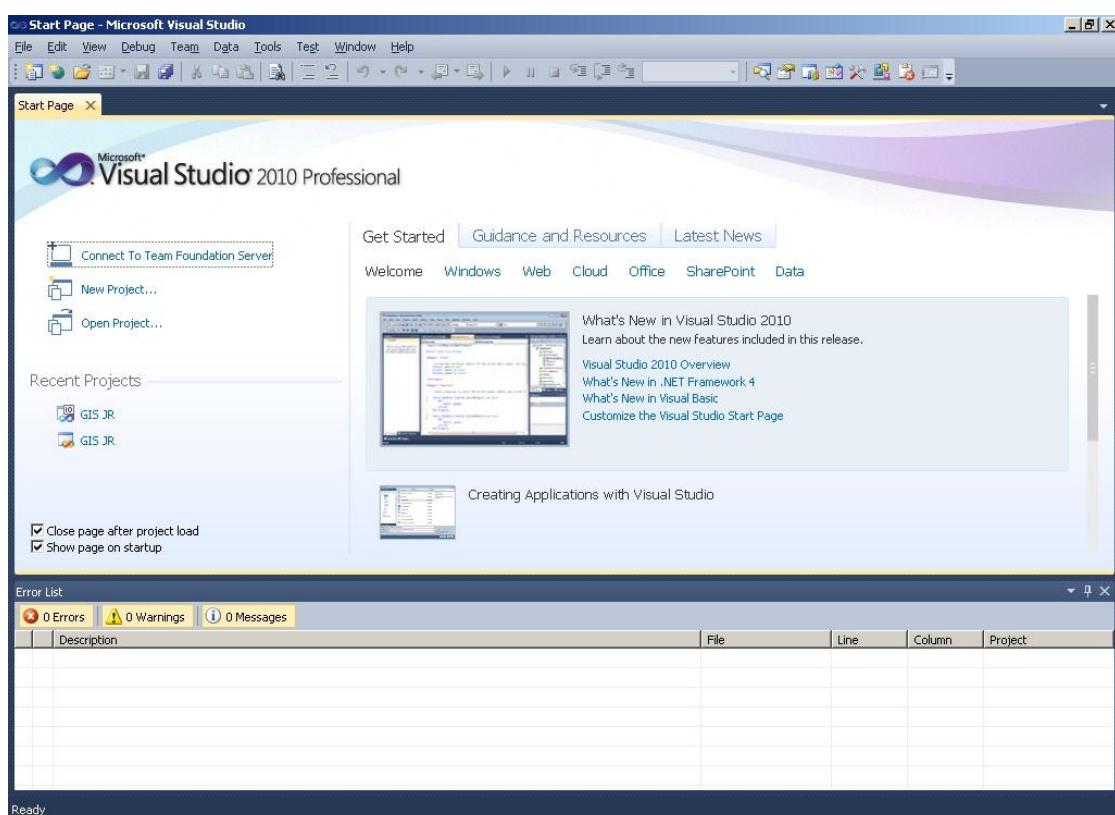
.NET Framework ima dvije važne komponente: *Common Language Runtime (CLR)* i *.NET Framework Class Library*.

*Common Language Runtime* je odgovoran za upravljanje kodom za vrijeme izvršavanja. Brine se o nekim osnovnim stvarima kao što su upravljanje memorijom i upravljanje dretvama (*engl. thread*), ali i drugim stvarima koje omogućavaju sigurnost i robustnost aplikacija.

*.NET Framework Class Library* je skup objektno-orientiranih tipova podataka (klasa), koje se mogu koristiti za pisanje bilo koje aplikacije. [9]

## 7.2 Izgled Visual Studioa 2010

Prilikom pokretanja *Visual Studio* otvara se početna stranica (slika 12.) na kojoj možemo odabrati jedan od projekata na kojem smo prije radili (u opcijama se može podesiti koliko zadnjih projekata će biti vidljivo). Možemo započeti novi projekt, a možemo potražiti i postojeći projekt na disku ukoliko se on već ne nalazi na listi zadnje korištenih projekata. [9]

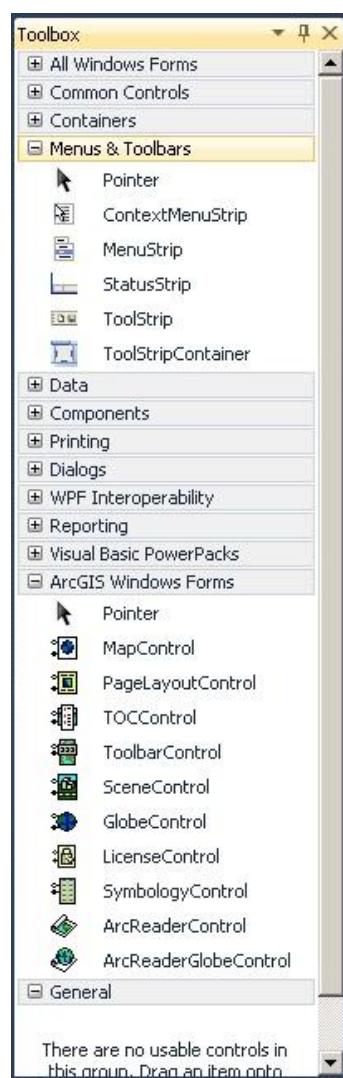


Slika 12. Početna stranica Visual Studioa

U nastavku ćemo opisati neke od važnijih prozora sa pripadajućim alatima.

## 7.2.1 Toolbox

*Toolbox* prozor (slika 13.) sadrži kontrole koje se mogu dodati u postojeći projekt. Kontrole su grupirane u grupe kako bi se olakšalo snalaženje. Kontrole vezane za podatke su grupirane u *Data* grupu, kontrole vezane za menije u *Menus & Toolbars* grupu, kontrole ArcGIS-a u *ArcGIS Windows Forms* itd.



Slika 13. Toolbox prozor Visual Studio

## 7.2.2 Server Explorer

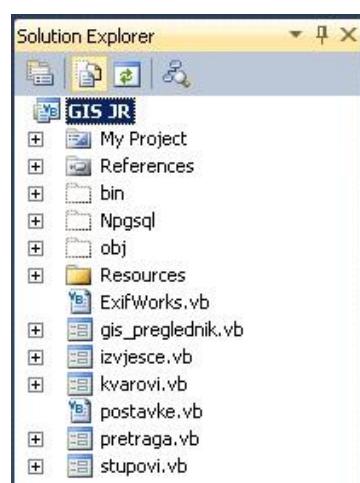
*Server Explorer* (slika 14.) prikazuje veze prema raznim serverima (*SQL, Crystal Services, Event Log, Message Queues,...*). Veze se po potrebi mogu dodati, a *Visual Studio* ih pamti, te su dostupne u svakom projektu.



Slika 14. Server explorer prozor Visual Studio

## 7.2.3 Solution Explorer

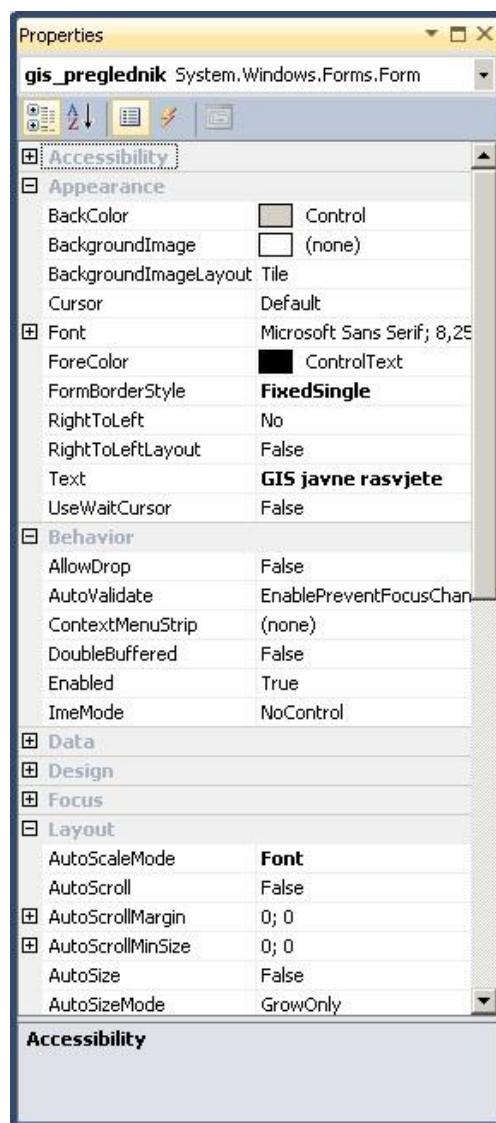
*Solution Explorer* prozor (Slika 15.) prikazuje trenutni radni prostor (*engl. solution*), sve projekte na kojima se trenutno radi i njihove datoteke. Datoteke je moguće grupirati po direktorijima, a rad je vrlo sličan *Windows Exploreru* ili nekoj sličnoj file manager aplikaciji.



Slika 15. Solution explorer prozor Visual Studio

## 7.2.4 Properties

*Properties* prozor (slika 16.) prikazuje svojstva (engl. *properties*) trenutno odabranog elementa. Najkorisniji je prilikom dizajniranja aplikacije. Svojstva i događaje je moguće grupirati po grupama (grupa za izgled, grupa za položaj na formi i slično) ili abecedno.



Slika 16. Properties prozor Visual Studio

### 7.3 Windows projekt

Windows projekt omogućava razvoj aplikacija za windows okruženje. Projekt se sastoji od prozora (formi – *engl. form*). Za svaku formu moguće je podesiti mnogo opcija. Npr. boja, pozadinska slika, izgled kursora, naslov u title-baru, meni kontrolu koja se prikazuje na vrhu prozora, početni izgled prozora (normalni, maksimizirani, minimizirani), početnu poziciju, accept button (defaultni gumb kada pritisnemo Enter), cancel button (defaultni gumb kada pritisnemo Esc), vidljivost *minimize* i *maximize* gumba, vidljivost ikone u gornjem lijevom kutu, vidljivost u *taskbar*-u, itd. Sve ove opcije mogu se za vrijeme *runtime-a* mijenjati iz koda.

Svaka forma sadrži kontrole koje se prikazuju korisniku i služe za interakciju korisnika i programa. Kontrole pripadaju formi (prozoru) i renderiraju se prilikom renderiranja forme. [9]

### 7.4 Windows kontrole

Za sve kontrole možemo definirati njima specifična svojstva (*engl. property*) i događaje (*engl. event*). Svojstva (*engl. property*) određuju neka svojstva kontrole, bilo da se odnose na izgled ili ponašanje kontrole.

Događaji (*engl. events*) su izazvani korisnikovim akcijama. Za svaki događaj se određuje funkcija koja će obraditi taj događaj (*engl. event handler*). *Visual Studio* će za ime *event handler-a* predložiti ime tipa *imekontrole\_događaj*, ali programer po potrebi to može promijeniti. [9]

U tablici (tablica 2) su prikazane vidljive kontrole i kontrole koje se ne prikazuju direktno korisniku, ali služe ostalim "vidljivim kontrolama" (npr. *ImageList* kontrola se ne vidi na formi, ali sadrži slike koje se vide kroz npr. *ToolBar* kontrolu).

<b>Naziv kontrole</b>	<b>Opis kontrole</b>
Label	Statični tekst.
LinkLabel	Statični tekst koji može poslužiti kao hyperlink.
Button	Gumb koji aktivira neku akciju.
TextBox	Unos teksta.
MainMenu	Meni na formi (prozoru).
CheckBox	Omogućava odabir više opcija iz liste ponuđenih.
RadioButton	Omogućava odabir jedne opcije iz liste ponuđenih.
GroupBox	Grupira više kontrola (npr. za grupiranje CheckBox-ova i RadioButton-a).
PictureBox	Slika.
Panel	Grupira više kontrola. Manje razlike u odnosu na GroupBox.
DataGridView	Prikaz podataka u tablici.
ListBox	Izbor jednog elementa iz liste.
CheckedListBox	Izbor više elemenata iz liste.
ComboBox	Padajuća lista elemenata.
ListView	Lista elemenata.
TreeView	Prikaz stabla elemenata.
TabControl	Može sadržavati više kontrola, ali tako da se prikazuju na više "stranica"
DateTimePicker	Kalendar kontrola s izborom datuma.
MonthCalendar	Kalendar kontrola, nešto drugačijeg izgleda od DateTimePicker kontrole.
HscrollBar	Horizontalna kontrola za skrolanje.
VscrollBar	Vertikalna kontrola za skrolanje.
Timer	Služi za odbrojavanje vremena. Ova kontrola se ne vidi na formi.
RichTextBox	Kontrola za upis teksta s minimalnim mogućnostima formatiranja.
ImageList	Lista slika. Kontrola nije vidljiva, samo sadrži slike u memoriji.
HelpProvider	Služi za prikaz help-a.
ToolTip	Padajući tekst koji služi kao pomoć.

ContextMenu	Meni koji se može prikazati npr. desnim klikom miša.
ToolBar	Traka s alatima.
StatusBar	Traka u kojoj se mogu ispisivati razne informacije.
NotifyIcon	Ikona koja se vidi u system tray-u.
OpenFileDialog	Standardni windows dijalog za otvaranje datoteka.
SaveFileDialog	Standardni windows dijalog za spremanje datoteka.
FolderBrowserDialog	Standardni windows dijalog za odabir direktorija.
FontDialog	Standardni windows dijalog za odabir fonta.
ColorDialog	Standardni windows dijalog za odabir boje.
PrintDialog	Standardni windows dijalog za ispis.
PrintPreviewDialog	Standardni windows dijalog za prikaz dokumenta koji treba biti ispisani.
PrintDocument	Standardni windows dijalog za odabir opcija ispisa.
PageSetupDialog	Standardni windows dijalog za odabir opcija stranice prilikom ispisa.
CrystalReportViewer	Prikaz Crystal Report-a.

Tablica 2. Prikaz nekih osnovnih komponenti u Visual Studiu 2010.

## 7.2 Izrada aplikacije

Grafički dio aplikacije izrađen je uz korištenje *ArcGIS*-ove *AxArcReader* kontrole. U nastavku ćemo najprije opisati *ArcGIS* aplikaciju, a zatim nastajanje grafičkog dijela aplikacije.

### 7.2.1 ArcGIS

*ArcGIS* je programski paket koji se sastoji od nekoliko GIS proizvoda razvijenih od strane ESRI-ja. *ArcGIS* uključuje sljedeće *Desktop* programe:

- *ArcReader* - besplatni preglednik karata i GIS podataka objavljenih u odgovarajućem ESRI formatu (\*.pmf) koristeći *ArcGIS Publisher* ekstenziju. *ArcReader* radi jedino s ovim formatom podataka. Ovaj preglednik omogućava neke od osnovnih alata za pregledavanje, ispis i izvršavanje upita nad prostornim podacima. *ArcReader* je uključen s bilo kojim *ArcGIS* produkтом, a također je i dostupan za besplatno skidanje s Interneta.

*ArcGIS Desktop* – licenciran na 3 razine funkcionalnosti i to kao:

- *ArcView* – desktop aplikacija na najnižoj razini licenciranja. Omogućuje pregledavanje prostornih podataka, kreiranje karata i izvršavanje osnovnih upita. S *ArcView*-om moguće je samo pregledavati i uređivati GIS podatke koji su pohranjeni u datotekama, ili pregledavati podatke pohranjene u relacijskim bazama podataka uz pomoć *ArcSDE*-a
- *ArcEditor* je *desktop* aplikacija na srednjoj razini licenciranja, nastavna na *ArcView* desktop aplikaciju. Dizajnirana je za napredno uređivanje prostornih podataka publiciranih u odgovarajućem ESRI formatu. Sadrži alate za izradu karata i prostornih podataka

korištenih u GIS-u, uključujući mogućnosti za uređivanje geoprostornih baza podataka, više korisničko uređivanje geoprostornih baza podataka, verzioniranje, uređivanje rasterskih podataka, vektorizaciju, napredno uređivanje vektorskih podataka, upravljanje rasterima, koordinatnu geometriju (COGO), te uređivanje geometrijskih mreža. *ArcEditor* nije namijenjen za napredne prostorne analize.

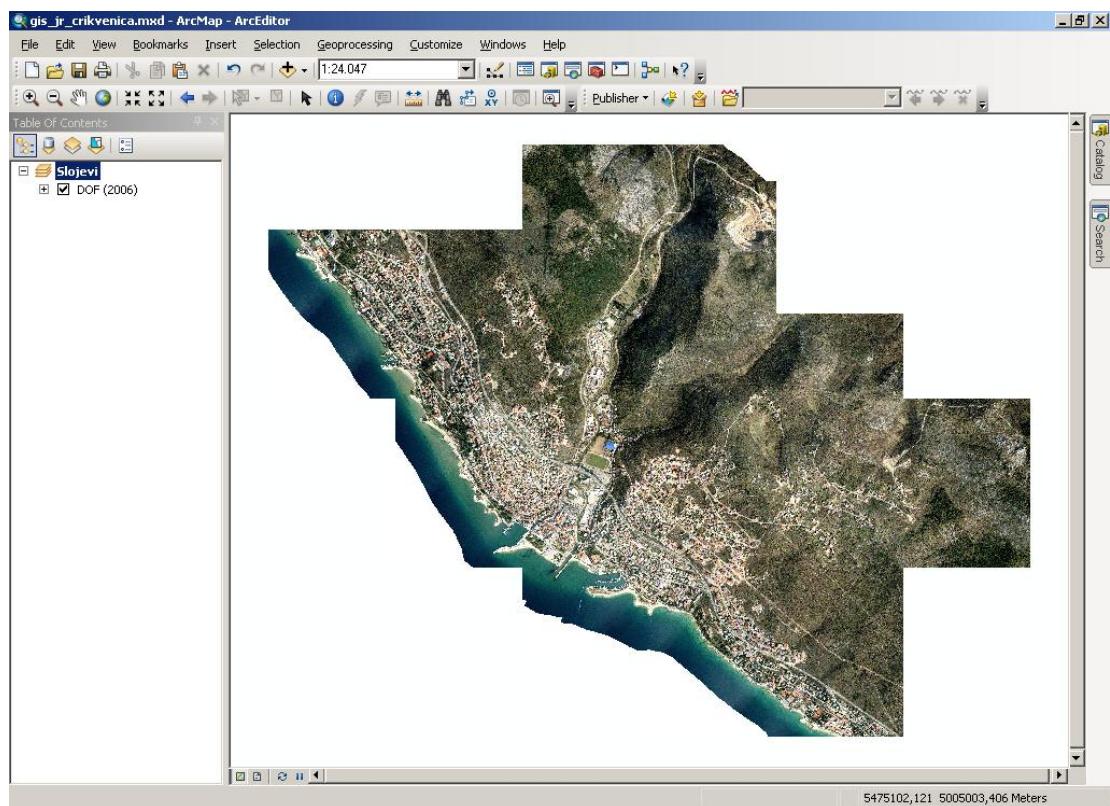
- *ArcInfo* je desktop aplikacija na najvišoj razini licenciranja, nastavna na *ArcEditor* desktop aplikaciju. Uključuje povećane mogućnosti u području prostorne analize, geoprocесiranja, upravljanja podacima itd.

Također postoje i poslužiteljske ArcGIS aplikacije, kao i aplikacije za *PDA* (*engl. Personal Digital Assistant*) uređaje. [URL 2]

### **7.2.2 Razvijanje korisničke ArcReader aplikacije s ArcGIS Publisher ekstenzijom**

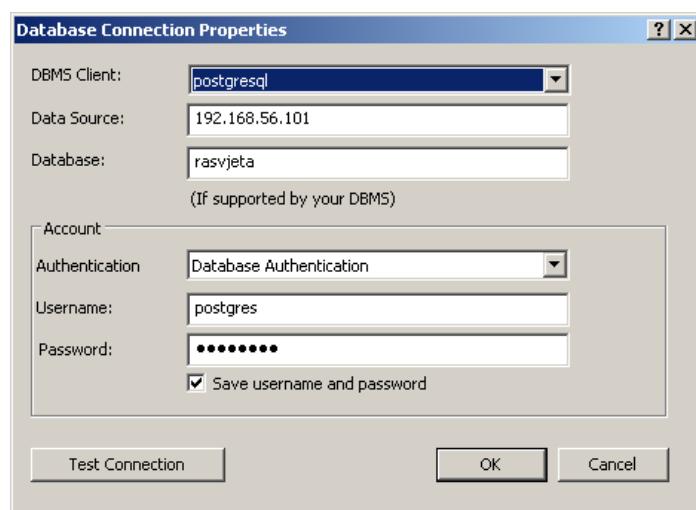
*ArcGIS Publisher* ekstenzija omogućava programerima razvijanje vlastitih GIS aplikacija koristeći *AxArcReader* kontrolu. Tako razvijene aplikacije se mogu instalirati na bilo kojem računalu koje ima *ArcGIS*, *ArcReader* ili *ArcGIS Engine*. Razvijene aplikacije se mogu besplatno distribuirati (jedini trošak koji razvijatelj aplikacije ima je licenca za *ArcGIS Publisher* ekstenziju koja košta oko 2000 američkih dolara). [URL 5]

Kako *ArcReader* program, a samim time i *AxArcReader* kontrola rade samo s \*.pmf datotekama izrađenima uz *ArcGIS Publisher* ekstenziju za *ArcGIS*, najprije ćemo opisati kreiranje \*.pmf datoteka. Za naš slučaj izradili smo četiri \*.pmf datoteke i to za svako mjesto posebno (crikvenica.pmf, dramalj.pmf, jadranovo.pmf, selce.pmf). U *ArcEditoru* smo za svako mjesto najprije učitali rasterske podatake (slika 17).



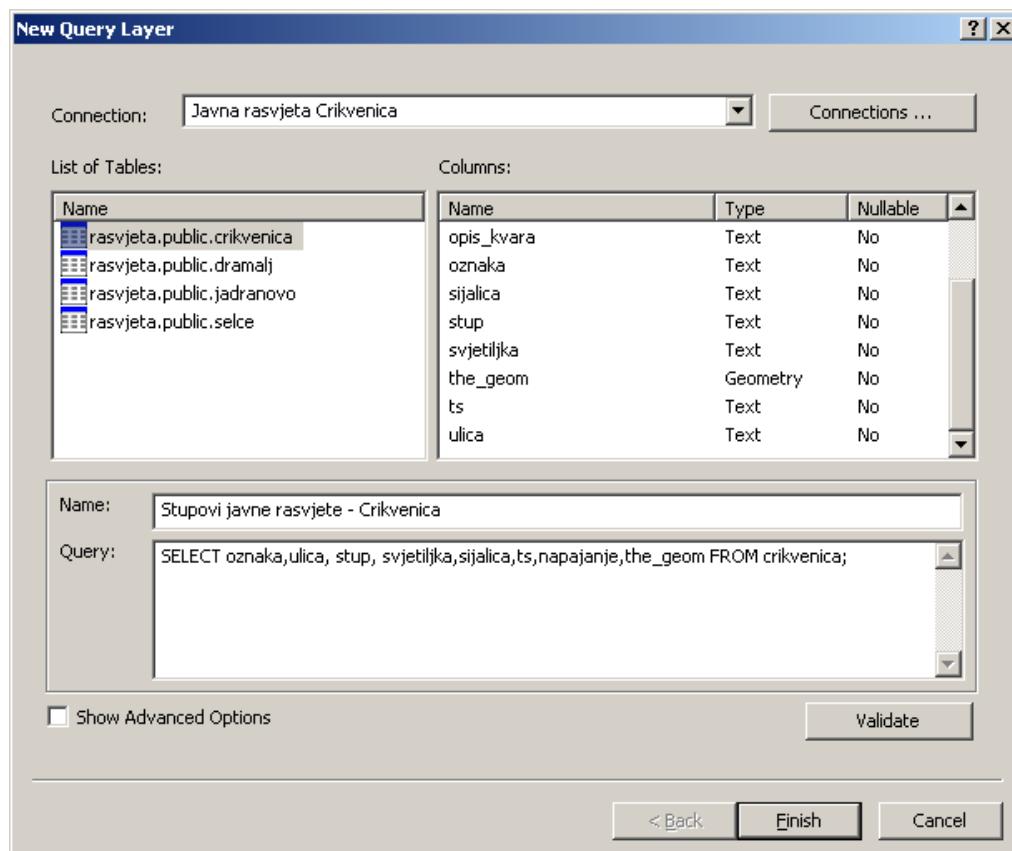
Slika 17. Učitane DOF podloge u ArcEditor

Zatim smo uspostavili vezu sa bazom podataka koristeći naredbu *File -> Add Data -> Add Query Layer*. Na početku smo morali podesiti postavke za vezu sa bazom podataka (slika 18).



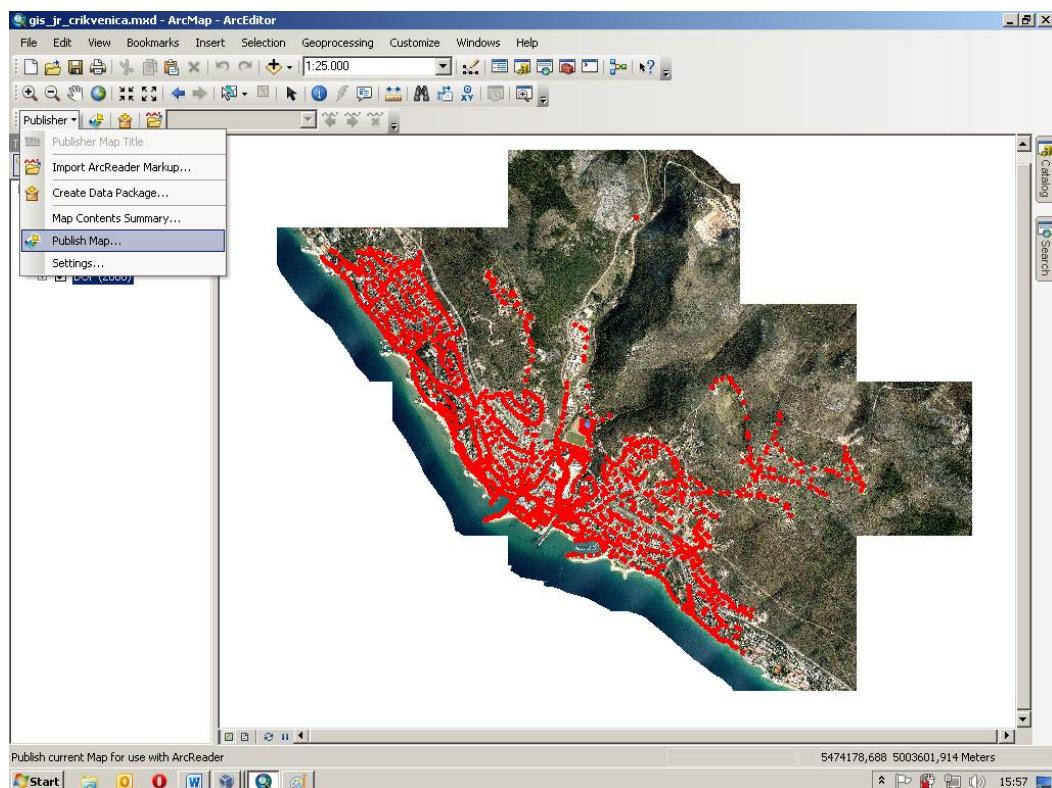
Slika 18. Postavke za vezu sa bazom podataka

Nakon definiranja veze sa bazom podataka bilo je potrebno izraditi upit za dohvaćanje podataka iz baze (slika 19).



Slika 19. Izrada upita za prihvatanje podataka iz baze

Kada je sloj s podacima o stupovima javne rasvjete dodan u *ArcEditor* uslijedilo je podešavanje prikaza. Odlučeno je da ćemo stupove označavati crvenim kružićima s time da se na mjerilu krupnjem od 1 : 2500 ispisuju i oznake stupova. Također dodan je sloj sa kvarovima na kojemu su kvarovi označeni većim žutim kružićima. Nakon završenog stiliziranja prikaza mogli smo pristupiti kreiranju \*.pmf datoteke korištenjem Publisher ekstenzije (slika 20).

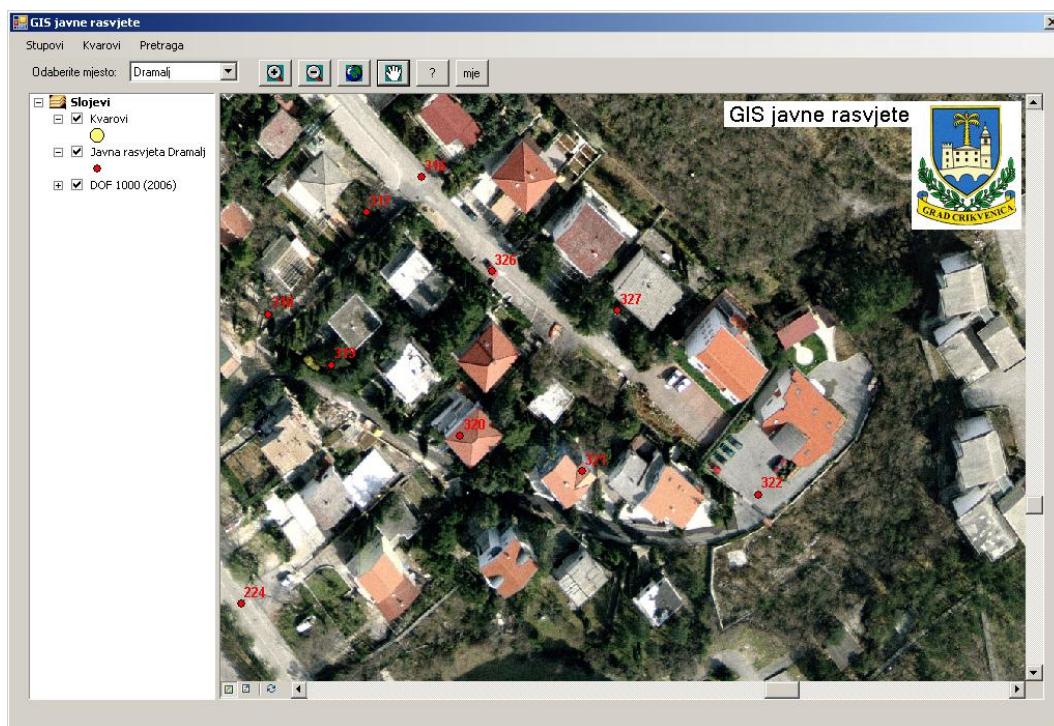


Slika 20. Kreiranje \*.pmf datoteke korištenjem Publisher ekstenzije

Nakon što su izrađene \*.pmf datoteke za svako mjesto, započeli smo s programiranjem aplikacije. Izrađena aplikacija sastoji se od pet formi. Nazivi formi su: *gis\_preglednik, izvješće, kvarovi, pretraga, stupovi*.

### 7.2.3 Forma "gis\_preglednik"

Ova forma je glavna forma aplikacije (slika 21). Na formi se nalazi *AxArcReader* kontrola koja prikazuje prostorne podatke.



Slika 21. Prikaz glavne forme aplikacije

Ova forma osim *AxArcReader* kontrole sadrži još i meni preko kojega se otvaraju ostale forme, te alatnu traku koja sadrži sljedeće alate: *zoom in*, *zoom out*, *zoom extent*, *zoom pan*, *identify* i *measure*. Također, na formi se nalazi *ComboBox* preko kojega odabiremo za koje mjesto trenutno administriramo podatke.

Odabirom mjesta iz *ComboBox-a* izvršava se slijedeći kod:

```
Private Sub cmbMjesto_SelectedIndexChanged(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles cmbMjesto.SelectedIndexChanged
    Try
        AxArcReaderControl1.LoadDocument("C:\GIS_JR\" & cmbMjesto.Text & ".pmf")
    Catch
        MsgBox(Err.Description)
    End Try
End Sub
```

Iz koda je vidljivo da se odabirom mjesta iz *ComboBox-a* u *AxArcReader* kontrolu učitava odgovarajuća \*.pmf datoteka za to mjesto.

U nastavku dajemo primjer za odabir alata iz alatne trake. Klikom na gumb zoom in izvršava se slijedeći kod:

```
Private Sub btnZoomIn_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles btnZoomIn.Click
    AxArcReaderControl1.CurrentARTool =
    ESRI.ArcGIS.PublisherControls.esriARTool.esriARToolMapZoomIn
End Sub
```

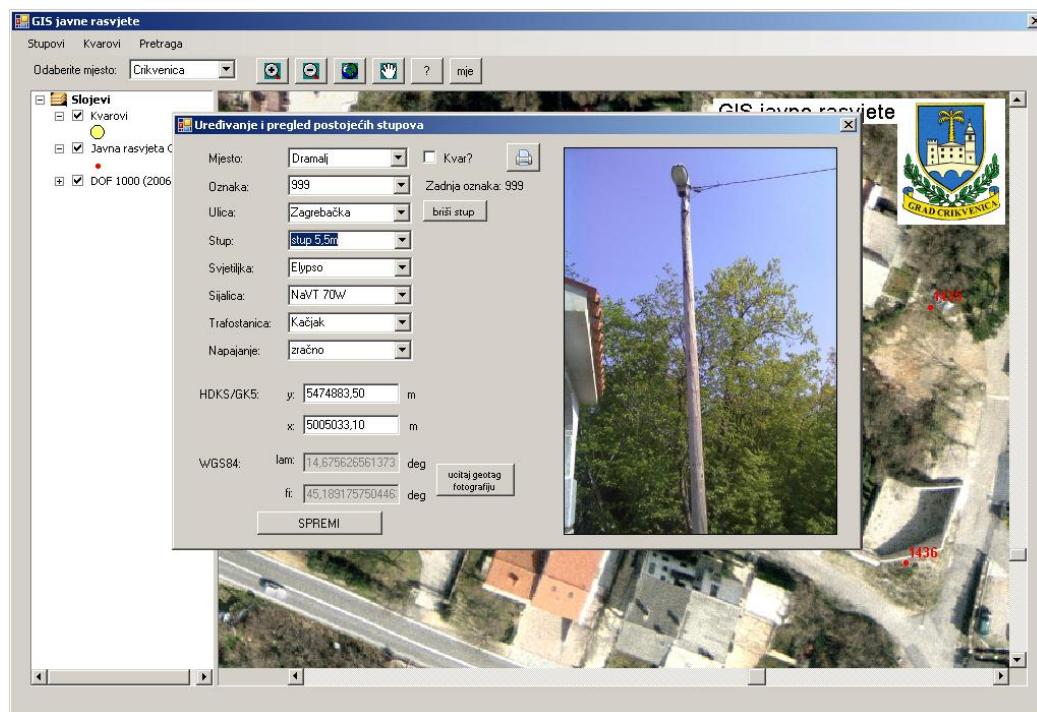
Ovaj dio koda postavlja ZoomIn alat kao trenutni aktivni alat *AxArcReader* kontrole. Gotovo identičan kod vrijedi i za *ZoomOut*, *ZoomPan*, *Identify* te *Measure* alate.

Ovdje ćemo prikazati dio koda za *ZoomExtent* alat:

```
Private Sub btnZoomExtent_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles btnZoomExtent.Click
    'Citanje i postavljanje extent vrijednosti
    Dim xMax, xMin, yMin, yMax As Double
    AxArcReaderControl1.ARPageLayout.FocusARMap.GetFullExtent(xMin, yMin, xMax,
yMax)
    AxArcReaderControl1.ARPageLayout.FocusARMap.SetExtent(xMin, yMin, xMax,
yMax)
    'Refresh prikaza
    AxArcReaderControl1.ARPageLayout.FocusARMap.Refresh()
End Sub
```

## 7.2.4 Forma "stupovi"

Ova forma (slika 22.) se koristi za administriranje objekata iz sustava javne rasvjete. Forma služi za unos novih stupova, uređivanje ili brisanje postojećih, označavanje kvara na određenom stupu, te izradu izvješća o stupu.



Slika 22. Prikaz forme "stupovi"

Odabirom određenog stupa, GIS preglednik se centriira na koordinate tog stupa.

### 7.5.4.1 Unos podataka

Unos atributnih podataka izvodi se preko *ComboBox-eva*, dok se unos položajnih podataka unosi preko *TextBox-eva* ili učitavanjem geotagirane slikovne datoteke.

Prilikom unosa atributnih podataka u *ComboBox-ove* se pozivaju svi mogući podaci već unijeti za to mjesto.

Slijedeći dio koda prikazuje pozivanja svih različitih ulica u *ComboBox* pod nazivom "cmbUlica":

```

'ucitavanje ulica

    conn.Open()

Dim command As New NpgsqlCommand("SELECT DISTINCT ulica FROM " & cmbMjesto.Text,
conn)
    Dim dr As Npgsql.NpgsqlDataReader

    dr = command.ExecuteReader()

    While dr.Read()

        cmbUlica.Items.Add(dr.Item("ulica"))

    End While

    conn.Close()

```

Sličan kod upotrebljava se za učitavanje svih ostalih atributnih podataka u *ComboBox-eve*. Ukoliko želimo možemo unijeti podatak koji do tada nije unijet u bazu. Taj podatak će također prilikom novog pozivanja podataka biti učitan u odgovarajući *ComboBox*.

U nastavku ćemo prikazati kod za unos novih podataka, te kod za uređivanje već postojećih podataka.

Slijedeći dio koda je jednak za unos novih podataka i za ažuriranje već postojećih.

```

Dim ulica As String
Dim svjetiljka As String
Dim sijalica As String
Dim stup As String
Dim napajanje As String
Dim oznaka As String
Dim trafostanica As String
Dim y As String
Dim x As String
Dim lam As String
Dim fi As String
Dim geometrija As String
Dim kvar As Boolean

oznaka = """ & txtOznaka.Text & """
        ulica = """ & cmbUlica.Text & """
svjetiljka = """ & cmbSvjetiljka.Text & """
        sijalica = """ & cmbSijalica.Text & """
        stup = """ & cmbStup.Text & """
        napajanje = """ & cmbNapajanje.Text & """
        trafostanica = """ & cmbTrafostanica.Text & """
y = Replace(txtY.Text, ",", ".")
x = Replace(txtX.Text, ",", ".")
lam = Replace(txtLam.Text, ",", ".")
fi = Replace(txtFi.Text, ",", ".")"

If chkKvar.Checked = True Then
    kvar = True

```

```

    Else
        kvar = False
    End If

'ispitivanje da li se geometrija unosi u projekcijskom koo sustavu ili u wgs84 koo
sustavu

If txtX.Enabled = True Then

    geometrija = " ST_GeomFromText('POINT(" & y & " " & x & ")',99995)"

Else

geometrija = " TRANSFORM(ST_GeomFromText('POINT(" & lam & " " & fi &
")',4326),99995)"

End If

```

Ukoliko unosimo nove podatake nastavni kod će izgledati ovako:

```

Try

conn.Open()

Dim command As New NpgsqlCommand("INSERT INTO " & cmbMjesto.Text & " svjetiljka,
sijalica, stup, napajanje, ts, the_geom, kvar) VALUES (" & oznaka & ", " & ulica &
"," & svjetiljka & ", " & sijalica & "," & stup & "," & napajanje & "," &
trafostanica & ", " & geometrija & ", " & kvar & ")", conn)

Dim dr As Npgsql.NpgsqlDataReader

dr = command.ExecuteReader()

conn.Close()

MsgBox("Zapis je dodan u bazu")

Catch

conn.Close()

MsgBox(Err.Description)

End Try

```

Ukoliko pak ažuriramo postojeći zapis u bazi, nastavni kod će izgledati ovako:

```

Try

conn.Open()

Dim command As New NpgsqlCommand("UPDATE " & cmbMjesto.Text & " SET ulica = "

```

```

& ulica & ", svjetiljka = " & svjetiljka & ", sijalica = " & sijalica & ",
stup = " & stup & ", napajanje = " & napajanje & ", ts =" & trafostanica &
", the_geom = " & geometrija & ", kvar= " & kvar & " WHERE oznaka = '" &
cmbOznaka.Text & "'", conn)

    Dim dr As Npgsql.NpgsqlDataReader

    dr = command.ExecuteReader()

    conn.Close()

    MsgBox("Zapis je azuriran u bazi")

Catch

    conn.Close()

    MsgBox(Err.Description)

End Try

```

#### 7.5.4.2 Označavanje kvara

Na ovoj formi nalazi se *CheckBox* kojim dodajemo ili uklanjamo kvar sa stupa (postavljamo vrijednost polja "kvar" u bazi podataka na *true* ili *false* vrijednost) (slika 23.).



Slika 23. Označavanje kvara

### 7.5.4.3 Izrada izvješća

Također se na formi nalazi i gumb za izradu izvješća o stupu javne rasvjete (slika 24.)



Slika 24. Gumb za izradu izvješća o stupu

Pritiskom na gumb poziva se forma "izvjesce" u kojoj se preko html stranice u web pregledniku prikazuje izvješće o stupu. U nastavku ćemo prikazati kod za izradu izvješća u html obliku.

```
Private Sub btnIspis_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles btnIspis.Click
    Dim izlazna_datoteka As String
    Try
        izlazna_datoteka = "C:\GIS_JR\izvjesce.html"
        IO.File.WriteAllText(izlazna_datoteka, "<html>")
        IO.File.AppendAllText(izlazna_datoteka, "<head>")
        IO.File.AppendAllText(izlazna_datoteka, "<meta http-equiv=""Content-Language"" content=""hr"">")
        IO.File.AppendAllText(izlazna_datoteka, "<meta http-equiv=""Content-Type"" content=""text/html; charset=UTF-8"">")
        IO.File.AppendAllText(izlazna_datoteka, "<title>GIS javne rasvjete</title>")
        IO.File.AppendAllText(izlazna_datoteka, "</head>")
```

```

IO.File.AppendAllText(izlazna_datoteka, "<link rel=""stylesheet"" type=""text/css"" href=""stilovi.css"" />")
IO.File.AppendAllText(izlazna_datoteka, "<body>")
IO.File.AppendAllText(izlazna_datoteka, "<h1>- GIS javne rasvjete - </h1>")
IO.File.AppendAllText(izlazna_datoteka, "<p align=""center"">")
IO.File.AppendAllText(izlazna_datoteka, "<img border=""0"" src=""crikvenica.gif"" width=""106"" height=""124"" align=""center"">")
IO.File.AppendAllText(izlazna_datoteka, "</p>")
IO.File.AppendAllText(izlazna_datoteka, "<div align=""center"">")
IO.File.AppendAllText(izlazna_datoteka, "<table>")
IO.File.AppendAllText(izlazna_datoteka, "<tr>")
IO.File.AppendAllText(izlazna_datoteka, "<th>Mjesto:</th>")
IO.File.AppendAllText(izlazna_datoteka, "<td>" & cmbMjesto.Text &
"</p></td>")
IO.File.AppendAllText(izlazna_datoteka, "</tr>")
IO.File.AppendAllText(izlazna_datoteka, "<tr>")
IO.File.AppendAllText(izlazna_datoteka, "<th>Oznaka stupa:</th>")
IO.File.AppendAllText(izlazna_datoteka, "<td>" & cmbOznaka.Text &
"</td>")
IO.File.AppendAllText(izlazna_datoteka, "</tr>")
IO.File.AppendAllText(izlazna_datoteka, "<tr>")
IO.File.AppendAllText(izlazna_datoteka, "<th>Fotografija:</th>")
IO.File.AppendAllText(izlazna_datoteka, "<td><img border=""0"" src=""Slike/"" & cmbMjesto.Text & "/" & cmbOznaka.Text & ".jpg"" width=""150"" height=""200"" align=""center""></td>")
IO.File.AppendAllText(izlazna_datoteka, "</tr>")
IO.File.AppendAllText(izlazna_datoteka, "<tr>")
IO.File.AppendAllText(izlazna_datoteka, "<tr>")
IO.File.AppendAllText(izlazna_datoteka, "<th>Ulica:</th>")
IO.File.AppendAllText(izlazna_datoteka, "<td>" & cmbUlica.Text &
"</td>")
IO.File.AppendAllText(izlazna_datoteka, "</tr>")
IO.File.AppendAllText(izlazna_datoteka, "<tr>")
IO.File.AppendAllText(izlazna_datoteka, "<th>Vrsta stupa:</th>")
IO.File.AppendAllText(izlazna_datoteka, "<td>" & cmbStup.Text &
"</td>")
IO.File.AppendAllText(izlazna_datoteka, "</tr>")
IO.File.AppendAllText(izlazna_datoteka, "<tr>")
IO.File.AppendAllText(izlazna_datoteka, "<th>Svjetiljka:</th>")
IO.File.AppendAllText(izlazna_datoteka, "<td>" & cmbSvetiljka.Text &
"</td>")
IO.File.AppendAllText(izlazna_datoteka, "</tr>")
IO.File.AppendAllText(izlazna_datoteka, "<th>Sijalica:</th>")
IO.File.AppendAllText(izlazna_datoteka, "<td>" & cmbSijalica.Text &
"</td>")
IO.File.AppendAllText(izlazna_datoteka, "</tr>")
IO.File.AppendAllText(izlazna_datoteka, "<tr>")
IO.File.AppendAllText(izlazna_datoteka, "<th>Trafostanica:</th>")
IO.File.AppendAllText(izlazna_datoteka, "<td>" & cmbTrafostanica.Text &
"</td>")
IO.File.AppendAllText(izlazna_datoteka, "</tr>")
IO.File.AppendAllText(izlazna_datoteka, "<tr>")
IO.File.AppendAllText(izlazna_datoteka, "<th>Napajanje:</th>")
IO.File.AppendAllText(izlazna_datoteka, "<td>" & cmbNapajanje.Text &
"</td>")
IO.File.AppendAllText(izlazna_datoteka, "</tr>")
IO.File.AppendAllText(izlazna_datoteka, "<tr>")
IO.File.AppendAllText(izlazna_datoteka, "<th>Kvar?:</th>")
```

```

If chkKvar.Checked = True Then
    IO.File.AppendAllText(izlazna_datoteka, "<td>DA</td>")
Else
    IO.File.AppendAllText(izlazna_datoteka, "<td>NE</td>")
End If

IO.File.AppendAllText(izlazna_datoteka, "</tr>")
IO.File.AppendAllText(izlazna_datoteka, "<tr>")
IO.File.AppendAllText(izlazna_datoteka, "<th>Opis kvara:</th>")
IO.File.AppendAllText(izlazna_datoteka, "<td>" & lblOpisKvara.text &
"</td>")
IO.File.AppendAllText(izlazna_datoteka, "</tr>")
IO.File.AppendAllText(izlazna_datoteka, "</table>")
IO.File.AppendAllText(izlazna_datoteka, "<p>Izvješće kreirano " &
Today & " u " & TimeOfDay & "</p>")
IO.File.AppendAllText(izlazna_datoteka, "</body>")
IO.File.AppendAllText(izlazna_datoteka, "</html>")
izvjesce.Show()

Catch
    MsgBox(Err.Description)
End Try
End Sub

```



Slika 25. Izvješće o stupu

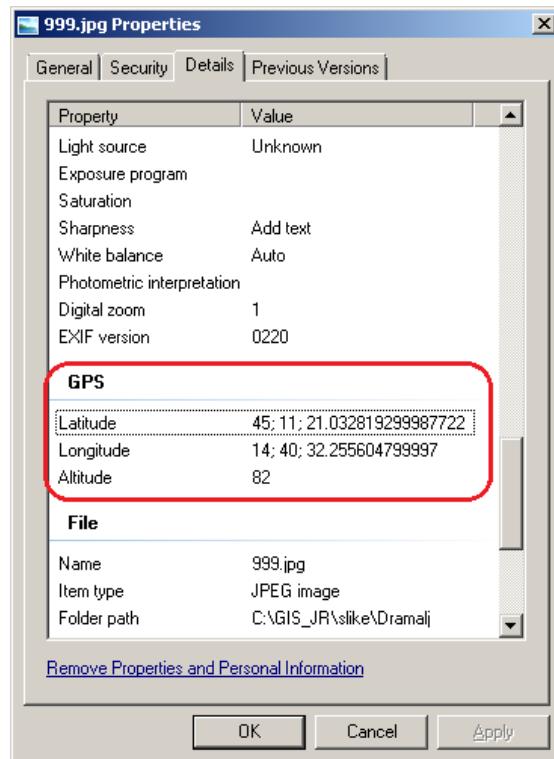
U nastavku ćemo opisati unos položajnih podataka stupova koristeći geotagirane fotografije.

#### 7.5.4.4 Geotagging

*Geotagging* ili označavanje zemljopisnih mesta je postupak dodavanja geografske identifikacije, meta podataka na različite medije kao što su fotografije, video, web stranice, itd. Sastoje se od geografske dužine i širine, a ponekad uključuje nadmorsku visinu, udaljenost i azimut. Slike su jedan od najstarijih vrsta medija za korištenje označavanja zemljopisnih mesta. Određeni formati poput npr. JPEG formata, omogućavaju ugrađivanje geografskih informacija unutar slike koje se tako spremaju zajedno sa fotografijama. Mobilni telefoni s ugrađenim fotoaparatima i GPS navigacijom su napravili *geotagging* fotografija više nego jednostavnim preko mogućnosti brzog dodavanja GPS informacija. Također, veliki broj digitalnih foto aparata ima ugrađenu ovu opciju. S druge strane, podaci koji se mogu dobiti iz označavanja zemljopisnih mesta predstavljaju mogući sigurnosni rizik za vlasnike fotografija, posebice kada se objavljuju na nekoj od društvenih mreža. Stoga treba voditi računa kada i kako te informacije zadržati, a kada i kako ih ukloniti. [URL 4]

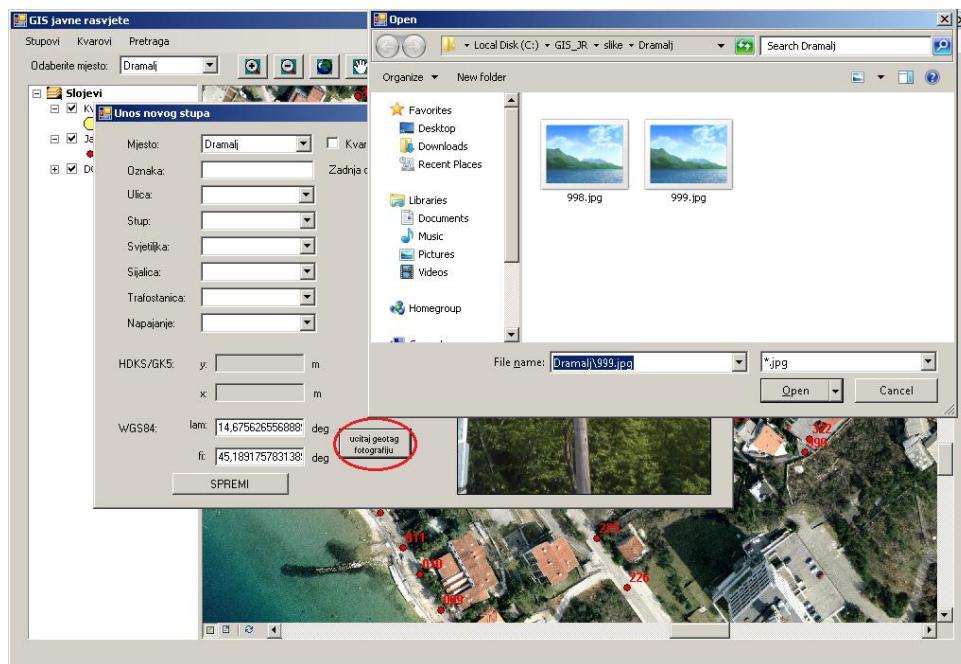
Jedan od načina prikupljanja podataka o stupovima javne rasvjete mogao bi biti korištenjem fotoaparata s GPS uređajem ili korištenjem mobitela koji ima mogućnost geotagiranja fotografija.

Na ovaj način mogli bi u isto vrijeme dobiti sliku i poziciju stupa. Međutim bitno je naglasiti to da pozicija stupa ne bi bila u potpunosti točna iz razloga što bi geotagiranjem dobili položaj stajališta fotoaparata, a ne točnu poziciju stupa. No, ukoliko bi tako određena točnost položaja stupa (cca. unutar 10 m) bila prihvatljiva, geotagiranje bi tada moglo biti jedna vrlo efikasna metoda prikupljanja podataka. Za točnije određivanje položaja stupa bilo bi potrebno koristiti točniji GPS uređaj, te fotoaparat (ukoliko želimo prikupljati slike).



Slika 26. Primjer geotag informacije

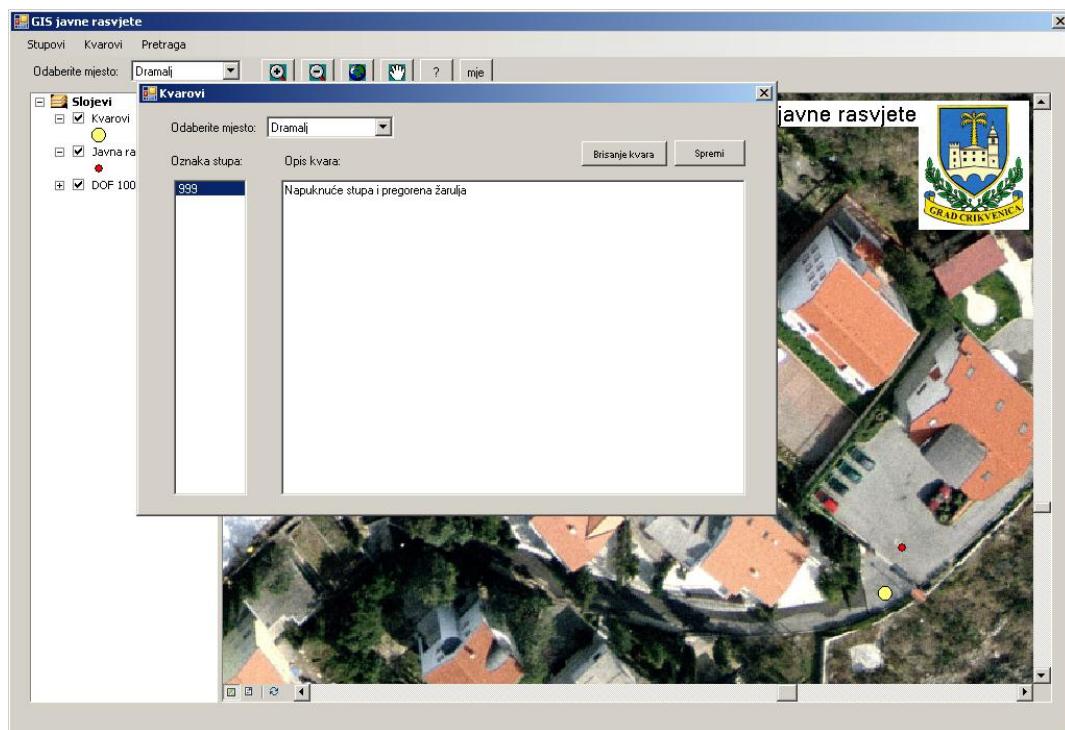
U okviru izrade aplikacije razrađena je mogućnost učitavanja geotagiranih slika (slika 27).



Slika 27. Primjer učitavanja geotagirane fotografije

### 7.5.5 Forma "kvarovi"

Forma "kvarovi" služi za opisivanje i brisanje trenutačnih aktivnih kvarova. Sam unos kvara izvodi se preko forme za uređivanje podataka o stupovima (forma "stupovi").



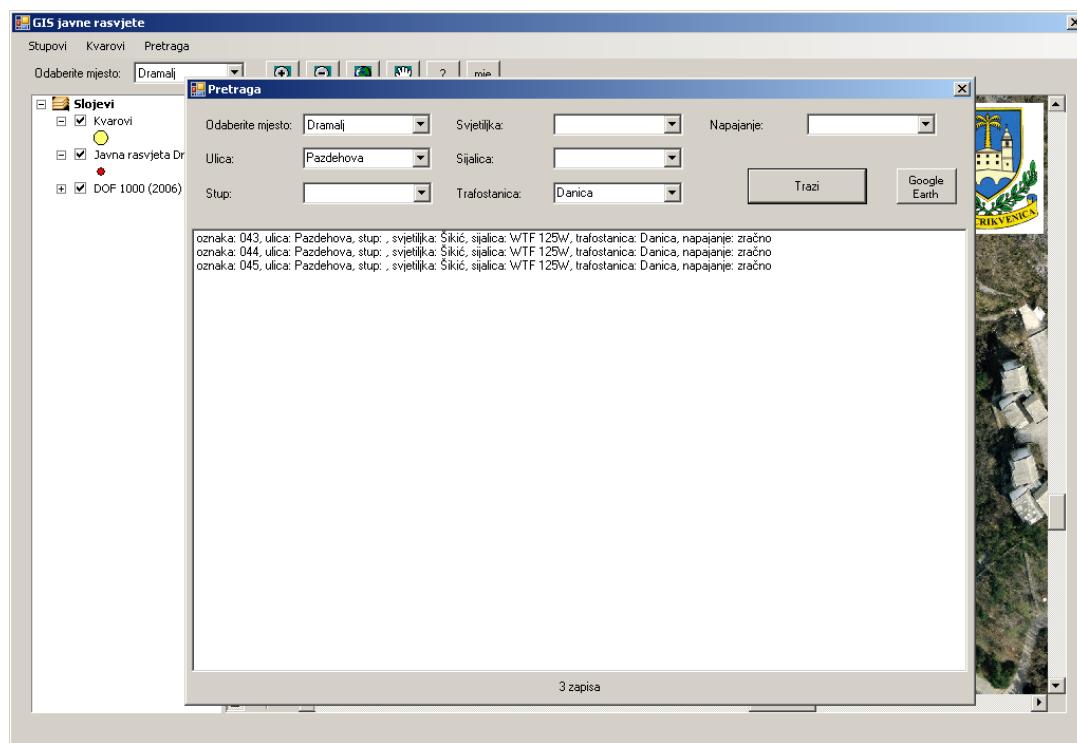
Slika 28. Primjer unosa opisa kvara

Ova forma sastoji se od *ComboBox-a* kojim odabiremo mjesto za koje želimo pregledavati, uređivati ili brisati kvarove, *ListBox-a* u koji se učitavaju stupovi s kvarovima, te *TextBox-a* u koji se upisuje opis kvara. Forma ima još dva gumba, gumb "brisanje kvara" kojim u potpunosti brišemo kvar sa određenog stupa, te gumb "spremi" kojim spremamo promjene u opisu kvara.

## 7.5.6 Forma "pretraga"

Ova forma omogućuje nam pretraživanje stupova po svim atributima. Sastoji se od *ComboBox-eva* pomoću kojih odabiremo attribute po kojima želimo pretraživati podatke.

Rezultati se ispisuju u *ListBox-u*. Klikom na svaki od rezultata pretrage prozor GIS preglednika se centririra na koordinate odabranog stupa.



Slika 29. Izgled forme za pretraživanje

Ovoj formi dodana je mogućnost zapisa rezultata pretrage u \*.kml datoteku , te automatsko pokretanje *Google Earth* aplikacije.

### 7.5.6.1 Google Earth

*Google Earth* je besplatni preglednik geografskih podataka. *Google Earth* prikazuje satelitske snimke Zemljine površine različite rezolucije, omogućujući korisnicima pregledavanje okomito na površinu Zemlje ili pod određenim kutom. Prikaz je stvoren od mnogo različitih satelitskih slika koje nisu iz stvarnog vremena, već su naknadno spojene. Većina površine Zemlje (osim nekih otoka) prekrivena je snimcima rezolucije od najmanje 15 metara. Neki dijelovi su pokriveni rezolucijom od čak 15 cm. Program omogućava pretraživanje po adresama, unos koordinata ili jednostavno korištenje miša za pretraživanje određene lokacije.

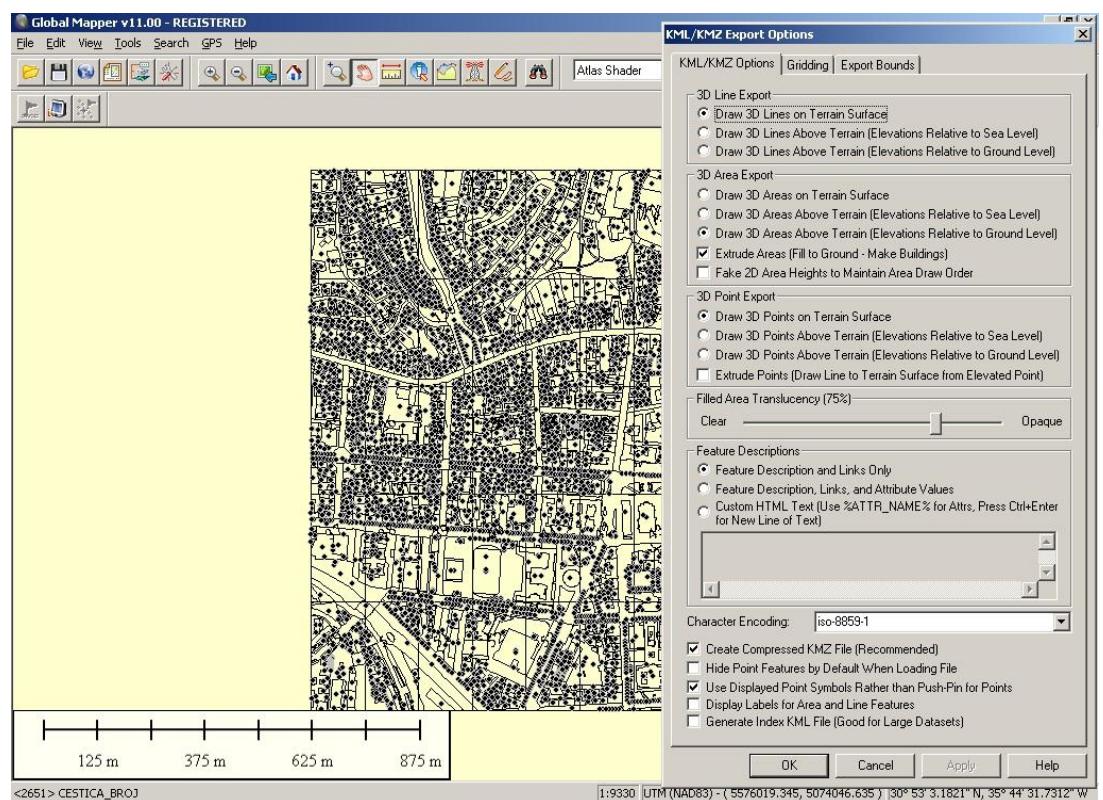
Za velike dijelove površine Zemlje dostupne su samo dvodimenzionalne slike, iz gotovo vertikalnih fotografija. Za ostale dijelove Zemljine površine dostupne su 3D slike terena i zgrada. *Google Earth* koristi digitalni model visina (DEM) prikupljen NASA-inom misijom *Shuttle Radar Topography Mission (SRTM)*.

Mnogo ljudi koristi aplikaciju kako bi dodali vlastite podatke. *Google Earth* omogućuje preklapanje svih vrsta slikovnih datoteka s površinom zemlje, te je također i WMS (engl. Web Map Service) klijent. *Google Earth* podržava upravljanje trodimenzionalnim geoprostornim podacima koristeći Keyhole Markup Language (KML).

*Google Earth* ima široku grupu korisnika, a s obzirom na stalnu nadogradnju i poboljšanja postoji i mogućnost korištenja u profesionalne svrhe. Pokazalo se da bi *Google Earth* mogao postati izuzetno koristan i za GIS timove koji izrađuju različite tipove tematskih podataka za javnu uporabu. Razlozi su očigledni: *Google Earth* je besplatan, te ima izuzetno popularno i profesionalno sučelje, kvantiteta i kvaliteta GIS podloga se ne može usporediti ni sa jednim pojedinačnim GIS-om u svijetu. [8]

### 7.5.6.1.1 Dodavanje vektorskih slojeva u Google Earth

*Google Earth* koristi isključivo \*.kml datoteke. Postoji nekoliko različitih programa za konverziju drugih popularnih datoteka u \*.kml. Jedan od takvih je i *Global Mapper*, razvijen od strane *Global Mapper Software LLC*. Ovaj program nudi konverziju između različitih tipova podataka među kojima je i \*.kml. Korištenjem ovog programa moguće je geodetima popularne \*.dwg i \*.dxf datoteke pretvoriti u \*.kml pri čemu je nužno za svaku ulaznu datoteku definirati koordinatni sustav (TOWGS). Ovo je otrebno iz razloga da bi se konverzija u \*.kml format koji koristi LATLON WGS84 (epsg: 4326) koordinatni sustav mogla izvršiti. Slika 30. prikazuje konverziju \*.dxf datoteke u \*.kml datoteku.

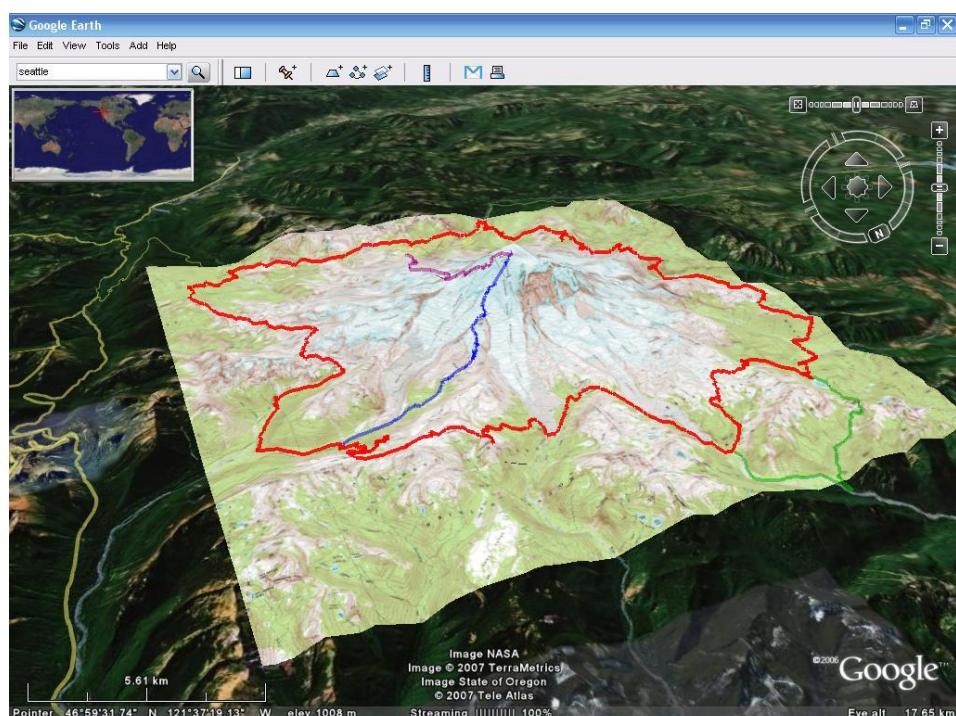


Slika 30. Konverzija \*.dxf datoteke u \*.kml datotoeku

### 7.5.6.1.2 Dodavanje rasterskih slojeva u Google Earth

Pridodavanje rasterskih slojeva u *Google Earth* je nešto složenije i potrebno je pripremiti rastersku kartu, odnosno sliku, u GIF, TIFF, PNG ili JPEG formatu. Slikovnoj datoteci treba biti pridodana tzv. *Header* datoteka, u ovom slučaju \*.kml datoteka. Preporuka je da obje datoteke budu u istome direktoriju i da imaju isti naziv.

Ukoliko želimo da rasteri budu dostupni na Internetu, potrebno je slikovnu datoteku postaviti na neki poslužitelj, te definirati točan URL unutar \*.kml datoteke. [8] Primjer dodavanja rasterske datoteke u *Google Earth* prikazuje slika 31.



Slika 31. Dodavanje rasterskog sloja u Google Earth

### 7.5.6.2 Izrada \*.kml datoteke

U nastavku ćemo prikazati kod za izradu \*.kml datoteke s rezultatima pretrage.

```
izlazna_datoteka = "C:\GIS_JR\pretraga.kml"

IO.File.WriteAllText(izlazna_datoteka, "<?xml version=""1.0"""
encoding=""UTF-8""?>")
IO.File.AppendAllText(izlazna_datoteka, "<kml
xmlns=""http://www.opengis.net/kml/2.2"">")
IO.File.AppendAllText(izlazna_datoteka, "<Document>")

conn.Open()
Dim command As New NpgsqlCommand("SELECT oznaka, ulica, stup, svjetiljka, sijalica,
ts, napajanje, askml(the_geom) as kml FROM " & cmbMjesto.Text & " WHERE " & ulica &
" AND " & stup & " AND " & svjetiljka & " AND " & sijalica & " AND " & trafostanica
& " AND " & napajanje & " ORDER BY oznaka ASC", conn)

Dim dr As Npgsql.NpgsqlDataReader

dr = command.ExecuteReader()

While dr.Read()

IO.File.AppendAllText(izlazna_datoteka, "<Placemark>")
IO.File.AppendAllText(izlazna_datoteka, "<name>" & dr.Item("oznaka") & "</name>")
IO.File.AppendAllText(izlazna_datoteka, "<description>")
IO.File.AppendAllText(izlazna_datoteka, "<![CDATA[")
IO.File.AppendAllText(izlazna_datoteka, "<p>" & "ulica: " & dr.Item("ulica") &
"</p>")
IO.File.AppendAllText(izlazna_datoteka, "<p>" & "stup: " & dr.Item("stup") & "</p>")
IO.File.AppendAllText(izlazna_datoteka, "<p>" & "svjetiljka: " &
dr.Item("svjetiljka") & "</p>")
IO.File.AppendAllText(izlazna_datoteka, "<p>" & "sijalica: " & dr.Item("sijalica") &
"</p>")
IO.File.AppendAllText(izlazna_datoteka, "<p>" & "ts: " & dr.Item("ts") & "</p>")
IO.File.AppendAllText(izlazna_datoteka, "<p>" & "napajanje: " & dr.Item("napajanje") &
"</p>")
IO.File.AppendAllText(izlazna_datoteka, "]]>")
IO.File.AppendAllText(izlazna_datoteka, "</description>")
IO.File.AppendAllText(izlazna_datoteka, dr.Item("kml"))
IO.File.AppendAllText(izlazna_datoteka, "</Placemark>")

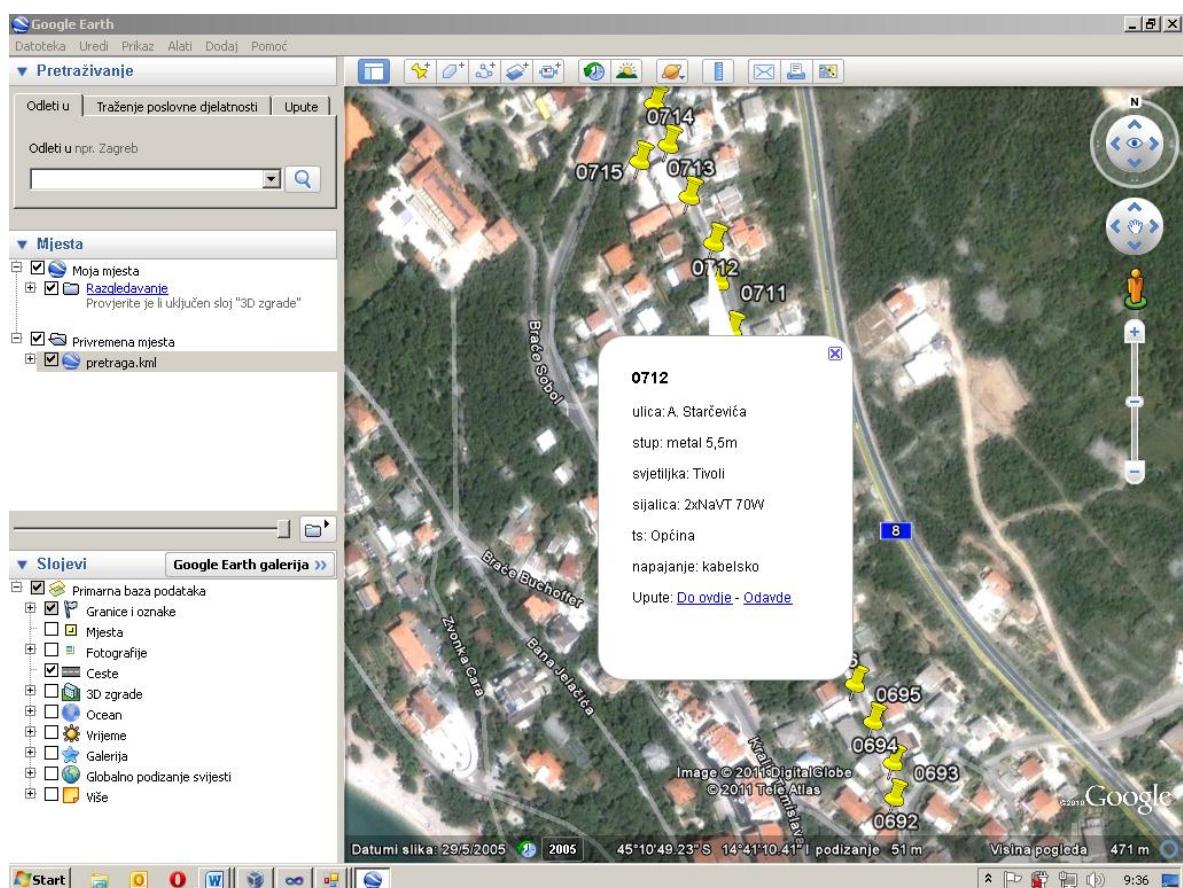
End While

conn.Close()

IO.File.AppendAllText(izlazna_datoteka, "</Document>")
IO.File.AppendAllText(izlazna_datoteka, "</kml>")
```

Nakon izrade \*.kml datoteke, pokrećemo izrađenu \*.kml datoteku s *Google Earth* aplikacijom (slika 32).

```
Private Sub Button1_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles btnGEarth.Click
    Try
        Process.Start("C:\GIS_JR\pretraga.kml")
    Catch
        MsgBox(Err.Description)
    End Try
End Sub
```



Slika 32. Rezultati pretrage preklopjeni s Google Earth snimkom

## 8. Zaključak

Upravljanje sustavima kao što je npr. sustav gradske javne rasvjete bez računalne podrške gotovo je nezamislivo. U takvim sustavima postoji veliki broj objekata koje je potrebno administrirati, te bi administriranje u analognom obliku ili koristeći neprikladnu računalnu aplikaciju bilo dugotrajno i prije svega skupo.

Iako je razvoj računalne tehnologije izazvao veliku revoluciju u geoinformatici, te geografskim informacijskim sustavima, postojeći GIS programi ponekad su nepraktični ili preskupi za upravljanje određenim skupovima geoprostornih podataka. Iz navedenog razloga ponekad je potrebno izraditi vlastitu aplikaciju za tu namjenu.

Aplikacija za upravljanje sustavom javne rasvjete, izrađena u okviru ovog diplomskog rada, u potpunosti je izrađena besplatnim alatima za razvoj aplikacija. Iz čega proizlaze neke dobre ali i neke loše strane izrađene aplikacije.

Izrada aplikacije je po pitanju vizualizacije prostornih podataka bila poprilično ograničena mogućnostima ArcGIS AxArcReader kontrole. Ukoliko bi željeli izraditi fleksibilniju i mogućnostima napredniju aplikaciju, bilo bi potrebno koristiti ArcGIS Engine razvojni alat. Ovaj alat nije besplatan, te bi prvo trebalo nabaviti odgovarajuću licencu. Po pitanju atributnih podataka gotovo nije bilo ograničenja za razvijanje aplikacije.

U konačnici je dobivena besplatna aplikacija za upravljanje svim objektima iz sustava gradske javne rasvjete. Velika prednost aplikacije je što su podaci pohranjeni u prostornoj bazi podataka, te je njima moguće pristupiti i iz drugih GIS aplikacija ukoliko se za time ukaže potreba.

## LITERATURA

1. Gajski D., (2006): Geografski informacijski sustavi. Slajdovi s predavanja. Geodetski fakultet Sveučilišta u Zagrebu, Zagreb.
2. Pribičević B., Medak D. (2003): Geodezija u građevinarstvu. Uџbenik. Građevinski fakultet Sveučilišta u Rijeci, Rijeka.
3. Galathi Stephen R. (2006): *Geographic Information Systems Demystified*, Artech House.
4. Paul A. Longley, Michael F. Goodchild, David J. Maguire, David W. Rhind (2005): *Geographical Information Systems and Science*, John Wiley & Sons Ltd.
5. Gojko Bukvić (1995): PASCAL TURBO programiranje, Školska knjiga, Zagreb
6. Bob Booth and Andy Mitchell (2001): *Getting Started with ArcGIS*, ESRI.
7. Tim Patrick (2009): *Programming Visual Basic 2008*, O'Reilly Media
8. Malarić I (2009): WebGIS arheološkog nalazišta Crikvenica - "Igralište". Diplomski rad. Geodetski fakultet Sveučilišta u Zagrebu, Zagreb.
9. Milovac B. (2004): Microsoft Visual Studio .Net. Seminarski rad. Fakultet elektrotehnike i računarstva Sveučilišta u Zagrebu, Zagreb.
10. Viher M. (2007): Časopis Mreža, veljača, članak "Novi pogled na svijet"

URL 1: Open Geospatial Consortium, <http://www.opengeospatial.org/ogc>, 14.06.2011

URL 2: ArcGIS, <http://en.wikipedia.org/wiki/ArcGIS>, 18.06.2011

URL 3: Google Earth, [http://hr.wikipedia.org/wiki/Google\\_Earth](http://hr.wikipedia.org/wiki/Google_Earth), 16.06.2011

URL 4: Geotagging, <http://en.wikipedia.org/wiki/Geotagging>, 20.06.2011

URL 5: AxArcReader,  
[http://proceedings.esri.com/library/userconf/serug07/papers/state\\_local/1019%20\\_cust\\_om\\_arcreader.pdf](http://proceedings.esri.com/library/userconf/serug07/papers/state_local/1019%20_cust_om_arcreader.pdf), 20.06.2011

URL 6: Global Mapper, <http://www.globalmapper.com/>, 13.06.2011

URL 7: KML Reference, <http://code.google.com/intl/hr-HR/apis/kml/documentation/kmlreference.html>, 15.06.2011

## **Popis slika i tablica**

Slika 1. Jedan od prvih GIS-eva .....	12
Slika 2. Jedan od prvih GIS-eva .....	12
Slika 3. Položaji mjesta na kojima su pronađene umrle osobe, te položaji pumpa za vodu.....	13
Slika 4. Slojevi u GIS-u.....	16
Slika 5. Vektorski i rasterski tip podataka.....	17
Slika 6. Prikaz linije u vektorskem i rasterskom obliku .....	17
Slika 7. Odnos rezolucije i broja piksela .....	21
Slika 8. Klasični spektar sa sedam boja.....	22
Slika 9. Uporaba cijelog niza spektralnih boja je neophodna kako bi različite ciljeve pravilno interpretirali na slici dobivenoj daljinskim istraživanjima .....	22
Slika 10. Razlika između zapisa vektorskog i rasterskog modela podataka .....	24
Tablica 1. Prikaz podataka dobivenih u *.xls datoteci.....	25
Slika 11. Primjer SQL skripte za unos podataka u bazu podataka .....	33
Slika 12. Početna stranica Visual Studioa .....	37
Slika 13. Toolbox prozor Visual Studioa .....	38
Slika 14. Server explorer prozor Visual Studioa.....	39
Slika 15. Solution explorer prozor Visual Studioa .....	39
Slika 16. Properties prozor Visual Studioa.....	40
Tablica 2. Prikaz nekih osnovnih komponenti u Visual Studiu 2010.....	43
Slika 17. Učitane DOF podloge u ArcEditor .....	46
Slika 18. Postavke za vezu sa bazom podataka .....	46
Slika 19. Izrada upita za prihvaćanje podataka iz baze .....	47
Slika 20. Kreiranje *.pmf datoteke korištenjem Publisher ekstenzije.....	48
Slika 21. Prikaz glavne forme aplikacije.....	49
Slika 22. Prikaz forme "stupovi" .....	51
Slika 23. Označavanje kvara .....	54
Slika 24. Gumb za izradu izvješća o stupu .....	55
Slika 25. Izvješće o stupu .....	57
Slika 26. Primjer geotag informacije .....	59
Slika 27. Primjer učitavanja geotagirane fotografije .....	59

Slika 28. Primjer unosa opisa kvara.....	60
Slika 29. Izgled forme za pretraživanje.....	61
Slika 30. Konverzija *.dxf datoteke u *.kml datotoeku.....	63
Slika 31. Dodavanje rasterskog sloja u Google Earth .....	64
Slika 32. Rezultati pretrage preklopljeni s Google Earth snimkom .....	66

## PRILOZI

- Prilog 1 . Diplomski rad i korišteni materijali pohranjeni na CD mediju

Sadržaj CD medija:

<i>Naziv datoteke:</i>	<i>Opis:</i>
1. dsubat_diplomski.doc	Tekst diplomskog rada.
2. dsubat_diplomski.pdf	Tekst diplomskog rada.
3. podaci/crikvenica.xls	Izvorni podaci o stupovima u *.xls formatu.
4. podaci/dramalj.xls	
5. podaci/jadranovo.xls	
6. podaci/selce.xls	
7. gis_javne_rasvjete_vb_dot_net_2010.zip	Kompletni Visual Basic 2010 projekt.
8. javna_rasvjeta.backup	Backup baze podataka (PostgreSQL) .

# ŽIVOTOPIS

## EUROPEAN CURRICULUM VITAE FORMAT



### OSOBNI PODACI

Ime i prezime	<b>Dino Šubat</b>
Adresa	Zagrebačka 1a, 51265 Dramalj
Telefon	051 242 364, 098 361 400
Telefaks	051 788 075
Elektronička pošta, Web adresa	<a href="mailto:dsubat@geof.hr">dsubat@geof.hr</a>
Državljanstvo	Hrvatsko
Datum rođenja	26. svibnja 1987.

### RADNO ISKUSTVO

- Datum (od – do)      1.7.– 1.9 (svake godine počevši od 2006 do 2008)  
1.7 2009. do danas  
Gea d.o.o. Zagrebačka 1a, Dramalj  
Izrada računalnih aplikacija za potrebe tvrtke  
Geodetskko snimanje i iskolčavanje terena

### ŠKOLOVANJE

Datum	1994. – 2002.
Mjesto	Crikvenica
Ustanova	Osnovna škola Vladimira Nazora
Zvanje	
Datum	2002. - 2006.
Mjesto	Crikvenica
Ustanova	Opća gimnazija, dr. Antuna Barca
Zvanje	
Datum	2006. – 2009.
Mjesto	Zagreb
Ustanova	Geodetski fakultet, Sveučilište u Zagrebu

Zvanje	Sveučilišni prvostupnik, inženjer geodezije i geoinformatike, univ. bacc. ing. geod. et geoinf.
Datum	2009. - 2011.
Mjesto	Zagreb
Ustanova	Geodetski fakultet, Sveučilište u Zagrebu
Zvanje	Magistar inženjer geodezije i geoinformatike, mag. ing. geod. et. geoinf.

#### OSOBNE VJEŠTINE I KOMPETENCIJE

Materinji jezik	Hrvatski jezik
-----------------	----------------

Strani jezici

Jezik	Engleski, njemački jezik
Govori	+
Piše	+
Čita	+

#### TEHNIČKE VJEŠTINE I KOMPETENCIJE

Obrada teksta: Microsoft Word, UltraEdit  
 Tablični proračuni: Microsoft Excel, OpenOffice Calc  
 Prezentacijski alati: Microsoft PowerPoint  
 Obrada slike: Adobe Photoshop, Corel Draw  
 CAD alati: Autodesk AutoCAD (R14-2012, Overlay, Raster Design, Map, Civil 3D)  
 GIS alati: Intergraph Geomedia Professional, ArcGIS ArcEditor, ArcGIS Publisher  
 Baze podataka: Microsoft Access, PostgreSQL, PostGIS  
 Programiranje: HTML, CSS, PHP, Java Script, Visual Basic, Visual C#

#### UMJETNIČKE VJEŠTINE I KOMPETENCIJE

Sviranje klavira.

#### VOZAČKA DOZVOLA

DA - B kategorija