Poslovna aplikacija za agenciju za nekretnine nad relacijskom bazom podataka - Clarion

Boban, Antonio

Undergraduate thesis / Završni rad

2019

Degree Grantor / Ustanova koja je dodijelila akademski / stručni stupanj: **University of Rijeka / Sveučilište u Rijeci**

Permanent link / Trajna poveznica: https://urn.nsk.hr/urn:nbn:hr:195:007762

Rights / Prava: In copyright / Zaštićeno autorskim pravom.

Download date / Datum preuzimanja: **2024-08-08**

Sveučilište u Rijeci
Fakultet informatike i digitalnih tehnologija

Repository / Repozitorij:

Repository of the University of Rijeka, Faculty of Informatics and Digital Technologies - INFORI Repository

zir.nsk.hr

DIGITALNA KNJIŽNICA

DIGITALNI AKADEMSKI ARHIVI I REPOZITORIJI

SVEUČILIŠTE U RIJECI
ODJEL ZA INFORMATIKU
Preddiplomski jednopredmetni studij informatike

Antonio Boban

Poslovna aplikacija za agenciju za nekretnine nad relacijskom bazom podataka - Clarion

Završni rad

Mentor: dr. sc. Danijela Jakšić

U Rijeci, Rujan 2019.

Sadržaj

Sadržaj 2
Sažetak 4
Ključne riječi 4
Uvod 5
Model entiteti-veze 6
Relacijski model 10
Izrada rječnika 11
Izrada aplikacije 15
Gotova aplikacija 29
Zaključak 34
Literatura 35
Popis slika 36

ODJEL ZA INFORMATIKU
SVEUČILIŠTE U RIJECI

Rijeka, 7.6.2019

Zadatak za završni rad

Pristupnik: Antonio Boban

Naziv završnog rada: Poslovna aplikacija za agenciju za nekretnine nad relacijskom bazom podataka - Clarion

Naziv završnog rada na eng. jeziku: Real estate agency business application with relational database - Clarion

Sadržaj zadatka:

BP predstavlja kolekciju podataka, ograničenja i operacija koji reprezentiraju neke aspekte realnoga svijeta. Dakle, BP je model neke aplikacijske domene. Od studenta se očekuje da kreira poslovnu aplikaciju za obradu određenih poslovnih dokumenata vezanih uz rad agencije za nekretnine. Aplikacija mora raditi nad relacijskom BP u navedenoj aplikacijskoj domeni (agencija za nekretnine). Potrebno je napraviti model podataka i odgovarajuću BP u programskom alatu Clarion (8, 9 ili 10).

Mentor

dr. sc. Danijela Jakšić

Voditelj za završne radove

doc. dr. sc. Miran Pobar

Zadatak preuzet:

13.06.2019

(potpis pristupnika)

Sažetak

U ovom završnom radu prikazana je izrada poslovne aplikacije za agenciju za nekretnine nad relacijskom bazom podataka u programskom alatu Clarion. Agencije za nekretnine koriste ovakvu aplikaciju kao bazu podataka. U aplikaciji je moguće izvršavati četiri osnovne operacije (CRUD), a to su pregled, dodavanje, brisanje i izmjena zapisa. Model podataka izrađen je u aplikaciji Visio, dok su rječnik i sama aplikacija izrađeni u Clarionu. U uvodu ovog završnog rada detaljnije je opisana i predstavljena izrađena aplikacija.

Ključne riječi

Clarion, Nekretnine, Model, Atribut, CRUD, Aplikacija, Entitet, Veza, Ključ

Uvod

U ovom završnom radu prikazat ću način i proces izrade poslovne aplikacije za agenciju za nekretnine. U današnje vrijeme svaka poslovna organizacija ima i vlastitu poslovnu aplikaciju koja im služi za što jednostavniji, pristupačniji, brži i sigurniji pristup bazi podataka.

Pristupom poslovnoj aplikaciji, odnosno bazi podataka, možemo dohvatiti podatke vezane za agente koji rade u agenciji, kupce i prodavače nekretnina te nekretnine koje agencija ima u ponudi. Također, možemo dohvatiti podatke vezane uz izdate račune i usluge koje je agencija odradila za svoje klijente.

Prilikom izrade ove aplikacije koristio sam RAD (Rapid Application Development) alat Clarion 10. RAD alati su danas jako popularni jer nam omogućuju jednostavniji, brži i sigurniji način izrade aplikacija. Također, korišteno je znanje stečeno na kolegijima Baze podataka, Uvod u programsko inženjerstvo, Modeliranje podataka i mnogih drugih.

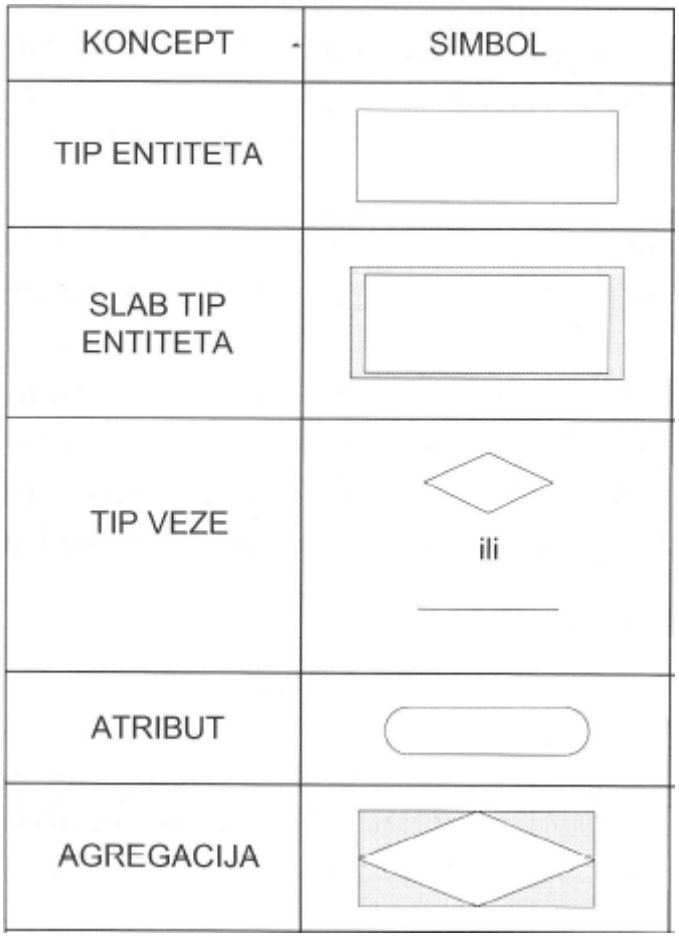
Prilikom izrade poslovne aplikacije najvažnije je dobro poznavanje sustava za koji se poslovna aplikacija izrađuje, u ovom slučaju način poslovanja agencije za nekretnine. Odlično poznavanje sustava nam uvelike olakšava posao jer ćemo u tom slučaju moći kvalitetno napraviti model entiteti-veze i relacijski model, uz pomoć kojih ćemo kasnije u alatu Clarion napraviti rječnik aplikacije i samu aplikaciju. Ukoliko su modeli entiteti-veze i relacijski model te rječnik napravljeni bez greške, sama izrada aplikacije koja preostaje je poprilično jednostavan proces.

Model entiteti-veze

Metoda entiteti – veze (skraćeno EV) je grafički prikaz međusobno povezanih grupa podataka promatranog sustava. EV je semantički bogata metoda za modeliranje podataka jer raspolaže ljudski bliskim konceptima. EV se odlikuje prirodnošću opisa, a njezini koncepti su bliski korisniku, pa je shema modela podataka laka za razumijevanje i komunikaciju korisnika i projektanta [1].

DEV je grafički prikaz modela podataka sustava, metodom EV [1].

Model podataka metodom EV gradi se upotrebom grafičkih simbola. Osnovni simboli za gradnju DEV-a dani su na slici 1 [1].



Slika 1 Grafički oblik koncepata strukture EV metode [1]

Entitet (engl. entity) je pojam u poslovanju o kome se zahtijeva čuvanje nekih podataka [1].

Jak tip entiteta (ili samo tip entiteta, engl. strong entity) je tip entiteta koji ima vlastiti primarni ključ i nije ovisan o drugim tipovima entiteta u modelu podataka [1].

Slab tip entiteta (engl. weak entity) je tip entiteta koji je na neki način ovisan o nekom drugome tipu entiteta, a ta se ovisnost prikazuje specijalnim tipom veze među jakim i slabim tipom entiteta [1].

Agregacija (mješoviti tip entiteta, agregirani tip entiteta) je apstrakcija u kojoj se tip veze između dvaju ili više tipova entiteta tretira kao novi tip entiteta [1].

Izradu aplikacije započinjemo izradom modela entiteti – veze. U nastavku slijedi detaljniji opis modela entiteti – veze napravljen za poslovnu aplikaciju za agenciju za nekretnine, koji je putem DEV-a prikazan na slici 2.

Model sadrži ukupno devet tablica, od čega je sedam jaki tip entiteta i po jedna tablica slabog tipa entiteta i agregacije.

Jaki tip entiteta AGENT sadrži atribute OIB agenta, ime agenta, prezime agenta, mob agenta, email agenta i datum zaposlenja. Atribut OIB agenta je primarni ključ. Povezan je sa još dva jaka tipa entiteta KUPAC i PRODAVATELJ te sa agregacijom POSREDNIČKI. Veza sa entitetom KUPAC ima brojnosti (0,M) sa strane entiteta AGENT i (1,1) sa strane entiteta KUPAC. Razlog ove brojnosti je taj što jedan agent može zastupati više kupaca, dok jednog kupca zastupa samo jedan agent. Veza sa entitetom PRODAVATELJ ima brojnosti (0,M) sa strane entiteta AGENT i (1,1) sa strane entiteta PRODAVATELJ. Razlog ove brojnosti je isti kao i za vezu sa entitetom KUPAC. Veza sa agregacijom POSREDNIČKI ima brojnost (0,M) sa strane entiteta AGENT i (1,1) sa strane agregacije POSREDNIČKI. Razlog ove brojnosti je taj što jedan posrednički ugovor može potpisati samo jedan agent, dok jedan agent može potpisati više posredničkih ugovora.

Jaki tip entiteta KUPAC sadrži atribute OIB kupca, naziv kupca, mob kupca, adresa kupca, opis potraživanja, poštanski broj, OIB agenta. Atribut OIB kupca je primarni ključ, dok su atributi poštanski broj i OIB agenta vanjski ključevi. Povezan je sa još dva jaka tipa entiteta AGENT i MJESTO, te sa agregacijom POSREDNIČKI. Veza sa entitetom AGENT je prethodno opisana. Veza sa entitetom MJESTO ima brojnosti (1,1) sa strane entiteta KUPAC i (0,M) sa strane entiteta MJESTO. Razlog ove brojnosti je taj što jedan kupac može nalaziti u samo jednom mjestu dok se u jednom mjestu može nalaziti više kupaca. Veza sa agregacijom POSREDNIČKI ima brojnost (0,M). Razlog ove brojnosti je u tome što jedan kupac može potpisati više posredničkih ugovora, dok jedan posrednički ugovor potpisuje samo jedan kupac.

Jaki tip entiteta PRODAVATELJ sadrži atribute OIB prodavatelja, naziv prodavatelja, mob prodavatelja, adresa prodavatelja, poštanski broj, OIB agenta. Atribut OIB prodavatelja je primarni ključ, dok su atributi poštanski broj i OIB agenta vanjski ključevi. Povezan je sa još tri jaka tipa entiteta AGENT, MJESTO i NEKRETNINA. Veza sa entitetom AGENT je prethodno opisana. Veza sa entitetom MJESTO ima brojnosti (1,1) sa strane entiteta PRODAVATELJ i (0,M) sa strane entiteta MJESTO. Razlog ove brojnosti je isti kao i za vezu između entiteta MJESTO i entiteta KUPAC. Veza sa entitetom NEKRETNINA ima brojnosti (0,M) sa strane entiteta PRODAVATELJ i (1,1) sa strane entiteta NEKRETNINA. Razlog ove brojnosti je taj što jedan prodavatelj može prodavati više nekretnina dok jednu nekretninu može prodavati samo jedan prodavatelj.

Jaki tip entiteta NEKRETNINA sadrži atribute ID nekretnine, vrsta nekretnine, adresa nekretnine, cijena nekretnine, najam, posebne napomene, balkon, kvadratura, parkirno mjesto, drvarnica, garaža, kat, prilaz za invalide, pogled, orijentacija, broj soba, dizalo, grijanje, OIB

prodavatelja i poštanski broj. Atribut ID nekretnine je primarni ključ, dok su atributi poštanski broj i OIB prodavatelja vanjski ključevi. Povezan je sa još tri jaka tipa entiteta PRODAVATELJ, MJESTO i USLUGA te agregacijom POSREDNIČKI. Veza sa entitetom PRODAVATELJ je opisana gore. Veza sa entitetom MJESTO ima brojnosti (1,1) sa strane entiteta NEKRETNINA i (0,M) sa strane entiteta MJESTO. Razlog ove brojnosti je taj što jedna nekretnina se može nalaziti točno u jednom mjestu, dok u jednom mjestu može biti više nekretnina. Veza sa entitetom USLUGA ima brojnosti (0,M) sa strane entiteta NEKRETNINA i (1,1) sa strane entiteta USLUGA. Razlog ove brojnosti je taj što jedna usluga može biti napravljena za točno jednu nekretninu, dok jedna nekretnina može biti uključena u više različitih usluga. Veza sa agregacijom POSREDNIČKI ima brojnost (0,M). Razlog ove brojnosti je taj što se posrednički ugovor potpisuje za točno jednu nekretninu, dok jedna nekretnina može biti na više posredničkih ugovora.

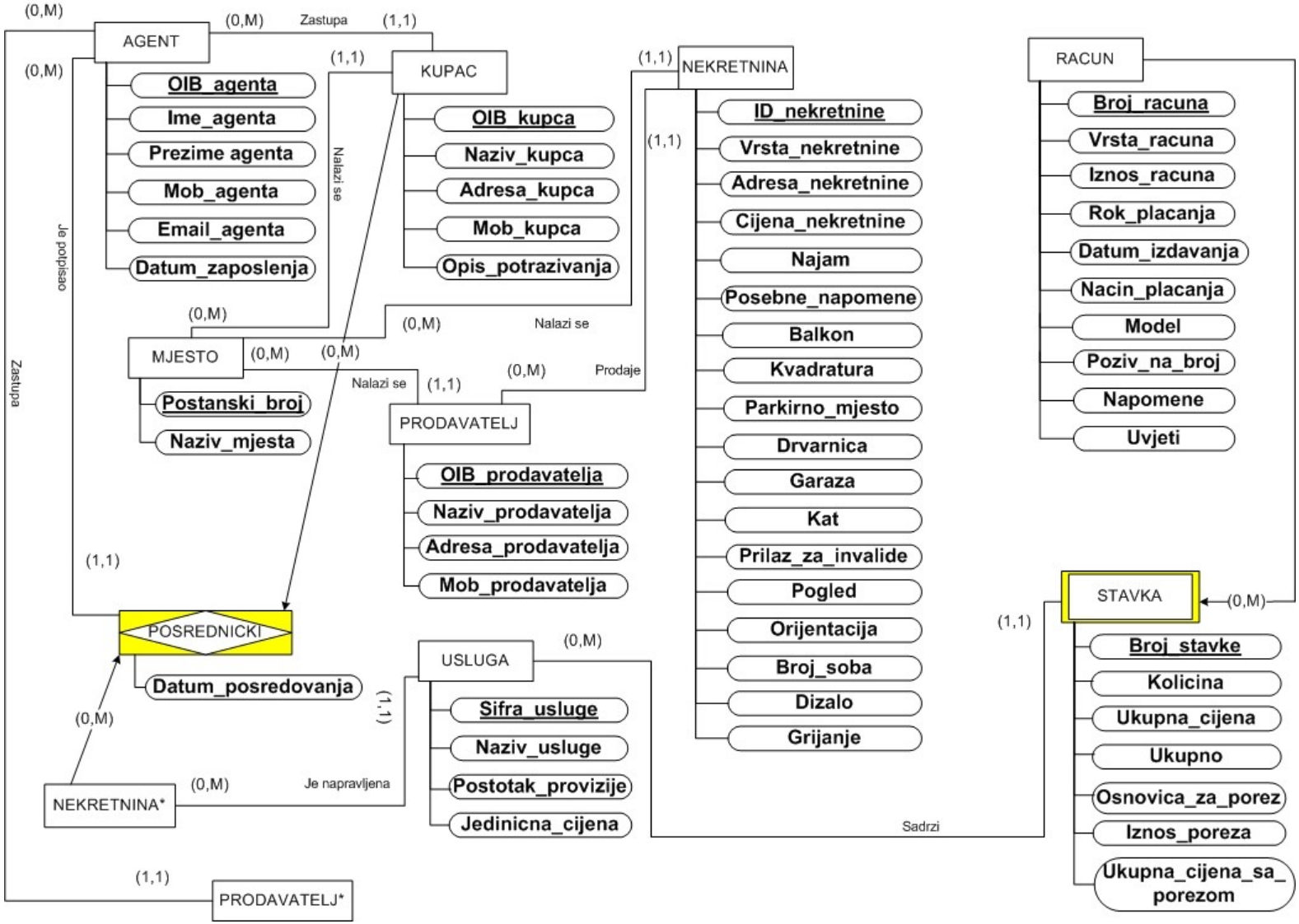
Jaki tip entiteta MJESTO sadrži atribute poštanski broj i naziv mjesta. Atribut poštanski broj je primarni ključ. Povezan je još s tri jaka tipa entiteta KUPAC, PRODAVATELJ i NEKRETNINA. Sve veze su već opisane.

Jaki tip entiteta USLUGA sadrži atribute šifra usluge, naziv usluge, postotak provizije, jedinična cijena, ID nekretnine. Atribut šifra usluge je primarni ključ, dok je atribut ID nekretnine vanjski ključ. Povezan je s još jednim jakim tipom entiteta NEKRETNINE i entitetom STAVKA koji je slabi tip entiteta RAČUN. Veza sa entitetom NEKRETNINE je već opisana. Veza sa entitetom STAVKA ima brojnosti (0,M) sa strane entiteta USLUGA i (1,1) sa strane entiteta STAVKA. Razlog ove brojnosti je taj što jedna usluga može biti na više stavki, dok jedna stavka može sadržavati samo jednu uslugu.

Jaki tip entiteta RAČUN sadrži atribute broj računa, vrsta računa, iznos računa, rok plaćanja, datum izdavanja, način plaćanja, model, poziv na broj, napomene, uvjeti. Atribut broj računa je primarni ključ. Povezan je s jednim slabim tipom entiteta STAVKA. Veza sa slabim tipom entiteta STAVKA ima brojnost (0,M). Razlog ove brojnosti je taj što jedan račun može sadržavati više stavki, dok jedna stavka može biti na samo jednom računu.

Slabi tip entiteta STAVKA sadrži atribute broj stavke, količina, ukupna cijena, osnovica za porez, iznos poreza, ukupna cijena sa porezom, šifra usluge, broj računa. Primarni ključ je kombinacija dvaju atributa broj računa i broj stavke, dok je šifra usluge vanjski ključ. Povezan je s još dva jaka tipa entiteta RAČUN i USLUGA. Veze između njih su već opisane.

Agregacija POSREDNIČKI sadrži atribute datum posredovanja, OIB kupca, OIB agenta i ID nekretnine. Primarni ključ je kombinacija dvaju atributa OIB kupca i ID nekretnine. Povezan je s još tri jaka tipa entiteta KUPAC, AGENT i NEKRETNINA. Sve veze su već opisane. Posrednički ugovor se potpisuje prilikom pokazivanja nekretnine kupcu od strane agenta.


AGENT
OIB_agenta
Ime_agenta
Prezime agenta
Mob_agenta
Email_agenta
Datum_zaposlenja
KUPAC
OIB_kupca
Naziv_kupca
Adresa_kupca
Mob_kupca
Opis_potrazivanja
NEKRETNINA
ID_nekretnine
Vrsta_nekretnine
Adresa_nekretnine
Cijena_nekretnine
Najam
Posebne_napomene
Balkon
Kvadratura
Parkirno_mjesto
Drvarnica
Garaza
Kat
Prilaz_za_invalide
Pogled
Orijentacija
Broj_soba
Dizalo
Grijanje
RACUN
Broj_racuna
Vrsta_racuna
Iznos_racuna
Rok_placanja
Datum_izdavanja
Nacin_placanja
Model
Poziv_na_broj
Napomene
Uvjeti
MJESTO
Postanski_broj
Naziv_mjesta
PRODAVATELJ
OIB_prodavatelja
Naziv_prodavatelja
Adresa_prodavatelja
Mob_prodavatelja
POSREDNICKI
Datum_posredovanja
USLUGA
Sifra_usluge
Naziv_usluge
Postotak_provizije
Jedinicna_cijena
STAVKA
Broj_stavke
Kolicina
Ukupna_cijena
Ukupno
Osnovica_za_porez
Iznos_poreza
Ukupna_cijena_sa_porezom
NEKRETNINA*
PRODAVATELJ*
Zastupa
Je potpisao
Nalazi se
Prodaje
Je napravljena
Sadrzi
(0,M)
(1,1)

Slika 2 DEV

Relacijski model

MJESTO (Poštanski broj, naziv mjesta)

AGENT (OIB agenta, ime agenta, prezime agenta, mob agenta, email agenta, datum zaposlenja)

KUPAC (OIB kupca, naziv kupca, mob kupca, adresa kupca, opis potraživanja, Poštanski broj, OIB agenta)

PRODAVATELJ (OIB prodavatelja, naziv prodavatelja, mob prodavatelja, adresa prodavatelja, Poštanski broj, OIB agenta)

NEKRETNINA (ID nekretnine, vrsta nekretnine, adresa nekretnine, cijena nekretnine, najam, posebne napomene, balkon, kvadratura, parkirno mjesto, drvarnica, garaža, kat, prilaz za invalide, pogled, orijentacija, broj soba, dizalo, grijanje, OIB prodavatelja, Poštanski broj)

RAČUN (Broj računa, vrsta računa, iznos računa, rok plaćanja, datum izdavanja, način plaćanja, model, poziv na broj, napomene, uvjeti)

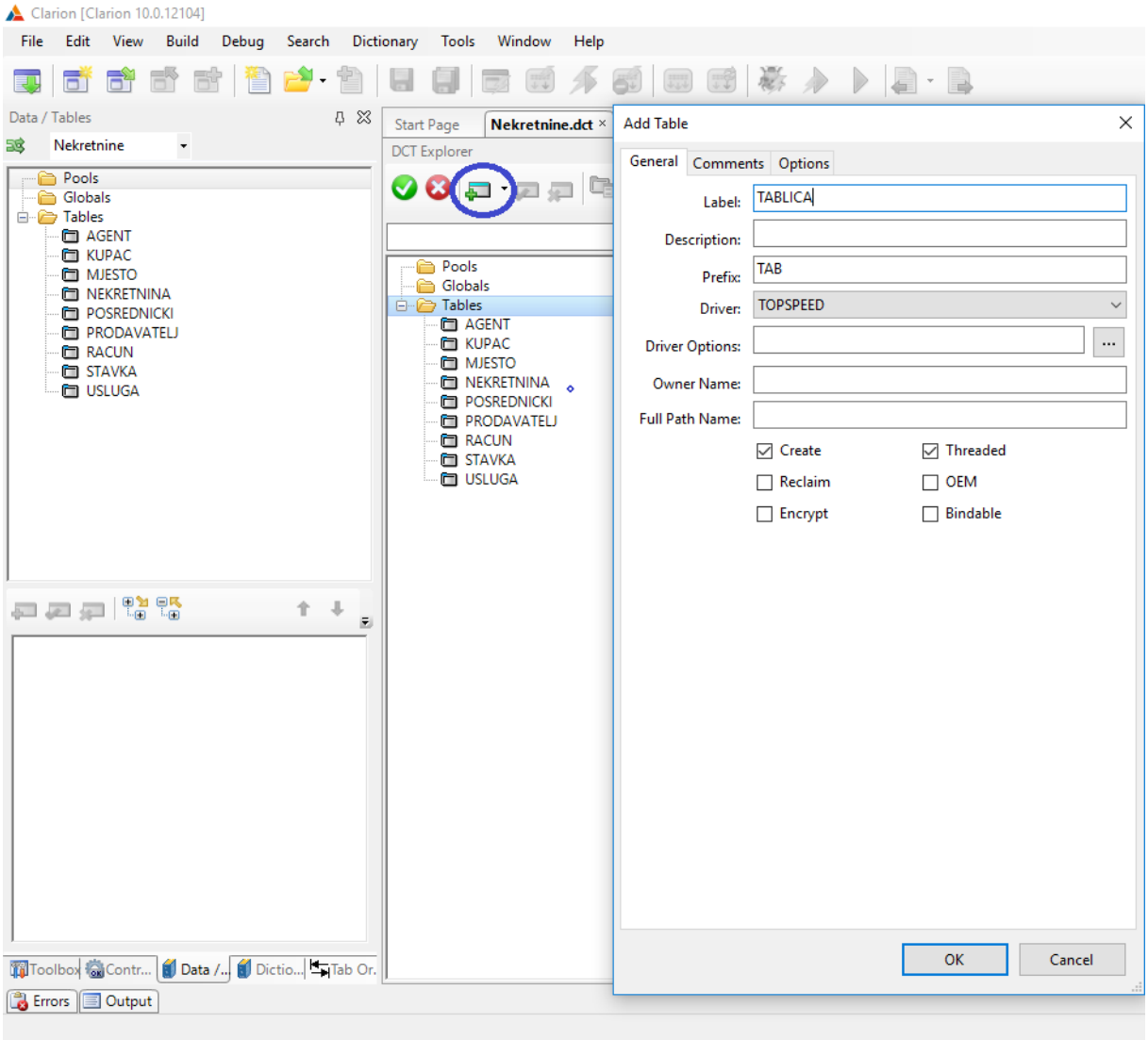
USLUGA (Šifra usluge, naziv usluge, postotak provizije, jedinična cijena, ID nekretnine)

STAVKA (Broj računa, Broj stavke, količina, ukupna cijena, osnovica za porez, iznos poreza, ukupna cijena sa porezom, Šifra usluge)

POSREDNIČKI (OIB kupca, ID nekretnine, datum posredovanja, OIB agenta)

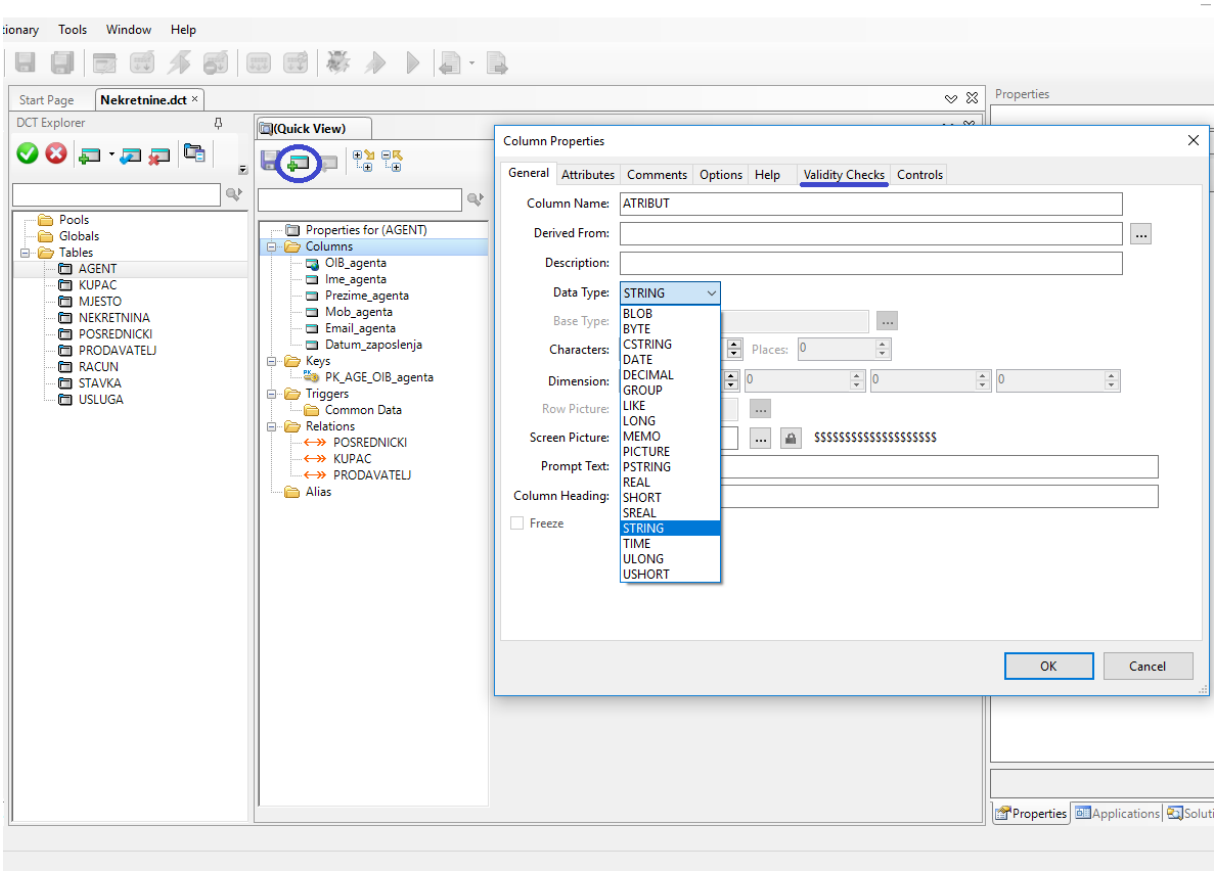
Izrada rječnika

Nakon što smo detaljno upoznali način poslovanja poslovnog sustava, napravili model entiteti-veze i relacijski model, počinjemo s izradom aplikacije, odnosno s njezinim glavnim dijelom rječnikom.



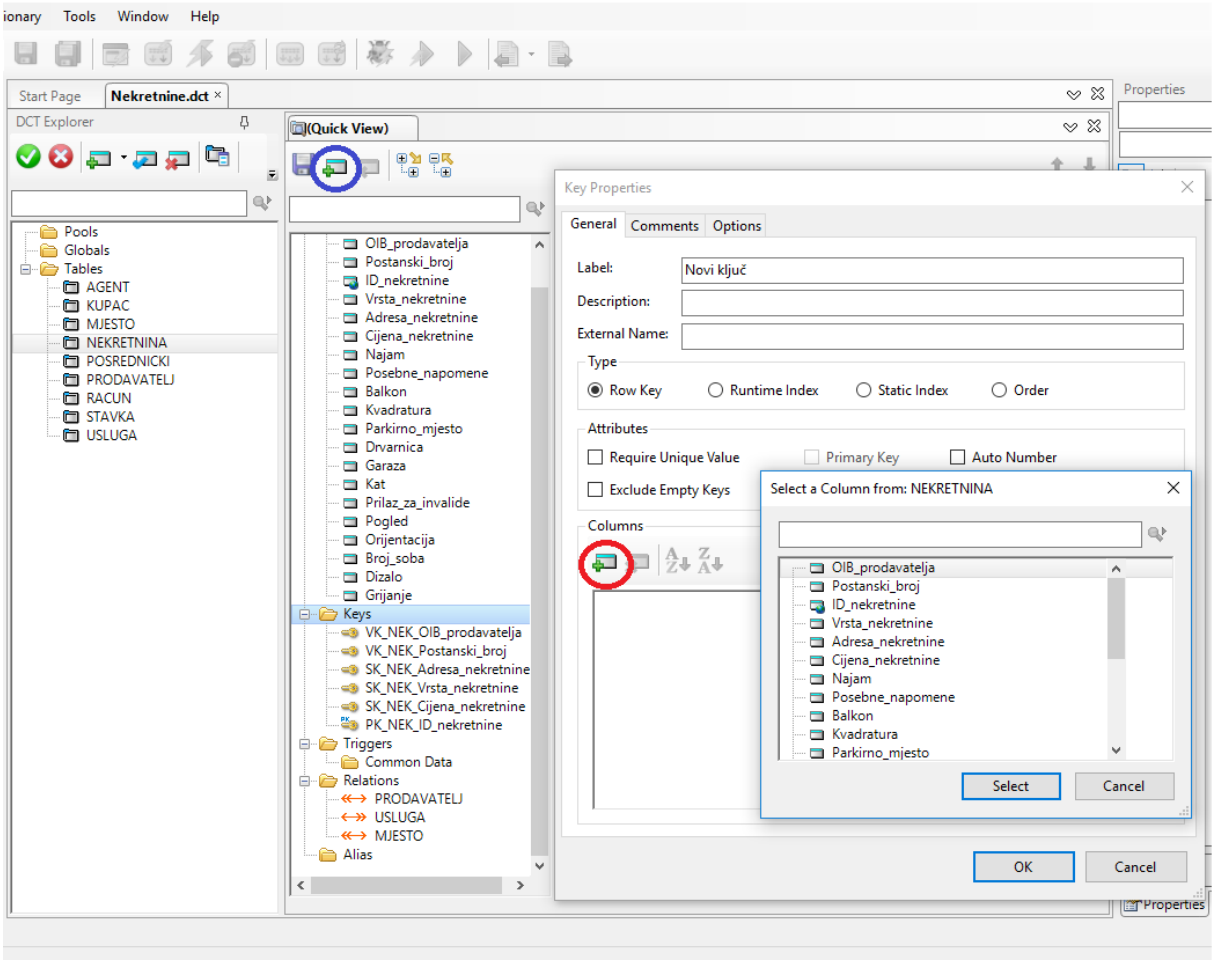
Slika 3 Izrada tablice

Prilikom izrade rječnika prvo dodajemo tablice koje smo prethodno definirali u modelima. Klikom na zaokruženu ikonu na slici 3 otvara nam se novi prozor „Add table“, uz pomoć kojeg dodajemo novu tablicu. U tom prozoru definiramo naziv tablice i njezinu skraćenicu koja će nam kasnije poslužit sa bolje raspoznavanje ključeva i atributa. Na slici 3 možemo vidjeti i sve tablice korištene za izradu poslovne aplikacije za agenciju za nekretnine.



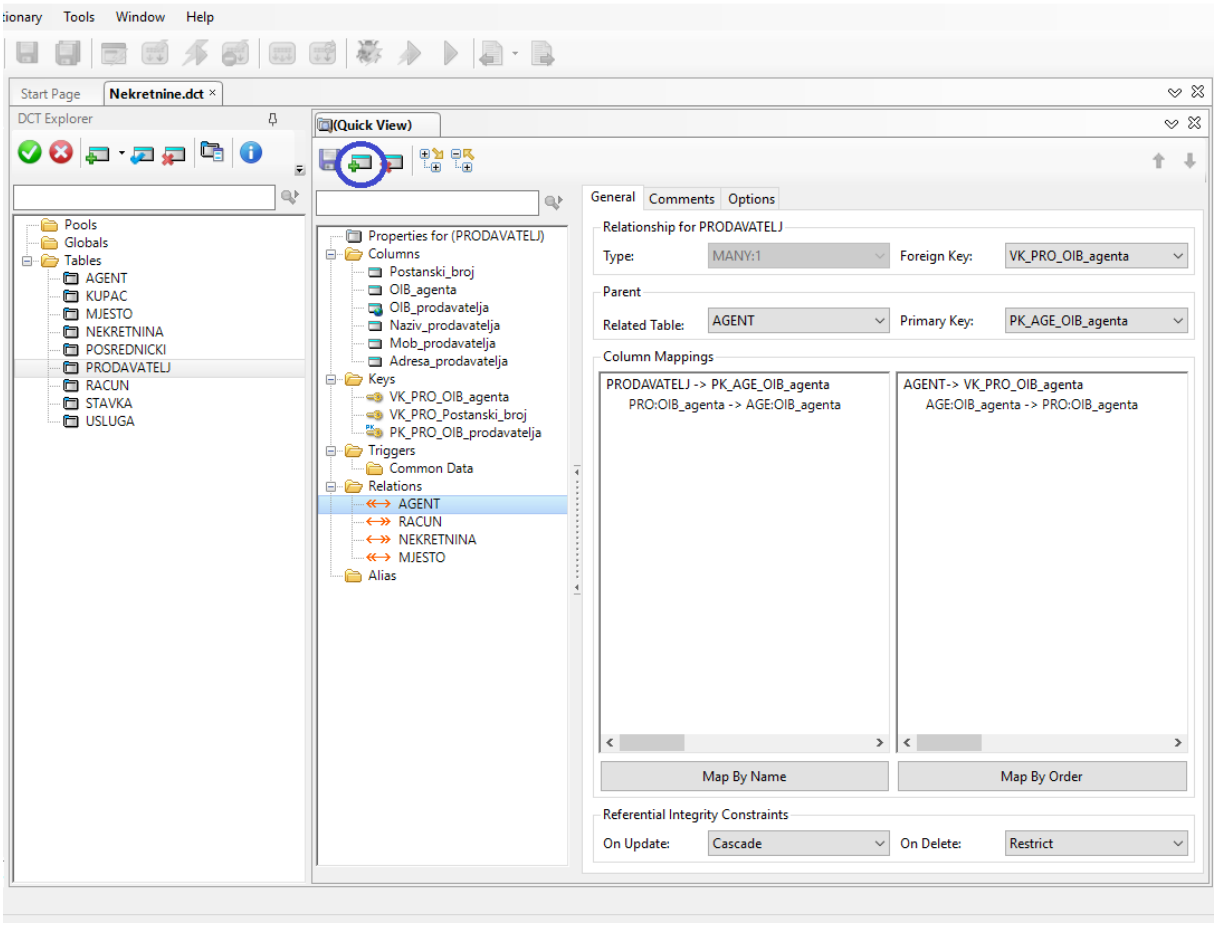
Slika 4 Dodavanje atributa

Nakon što smo definirali sve tablice koje će naš rječnik imati, krećemo s dodavanjem atributa za svaku pojedinu tablicu. Odabirom pojedine tablice otvara nam se „Properties“ za odabranu tablicu. Klikom na zaokruženu ikonu na slici 4 otvara nam se novi prozor „Column Properties“ uz pomoć kojeg dodajemo atribute u tablicu. Određujemo naziv i „data type“ atributa te ga dodajemo u tablicu. Ukoliko atribut koji smo dodali nije obavezan za unos, pod „Validity Checks“ (podcrtani dio na slici 4) odabiremo „No Checks“, ukoliko je obavezan odabiremo „Cannot be Zero or Blank“. Također, atribut može biti i iz neke druge tablice, njega dodajemo uz pomoć opcije „Derived From“. Na slici 4 možemo vidjeti sve atribute dodane u tablicu AGENT.



Slika 5 Dodavanje ključeva

Sljedeći korak u izradi rječnika je definiranje ključeva. Definirat ćemo tri vrste ključeva a to su primarni ključ, vanjski ključ i ključ za sortiranje. Primarni ključ je najvažniji ključ i svaka tablica ga mora imati, on je jedinstven i zbog njega se svaki redak u tablici razlikuje. Vanjski ključevi i ključevi za sortiranje nisu obavezni. Dodavanje ključeva započinje klikom na plavo zaokruženu ikonu na slici 5. Otvara nam se novi prozor „Key Propertis“ te definiramo naziv ključa. Ukoliko je ključ primarni pod „Attributes“ odabiremo opcije „Require Unique Value“ i „Primary Key“, ukoliko nije pod „Attributes“ ne odabiremo nijednu opciju. Nakon toga klikom na crveno zaokruženu ikonu na slici 5 otvara nam se prozor koji nam nudi sve atribute definirane u odabranoj tablici, odabiremo jedan ili više atributa (u slučaju da je ključ složen) te oni postaju ključ. Na slici 5 također možemo vidjeti sve ključeve definirane za tablicu NEKRETNINE.

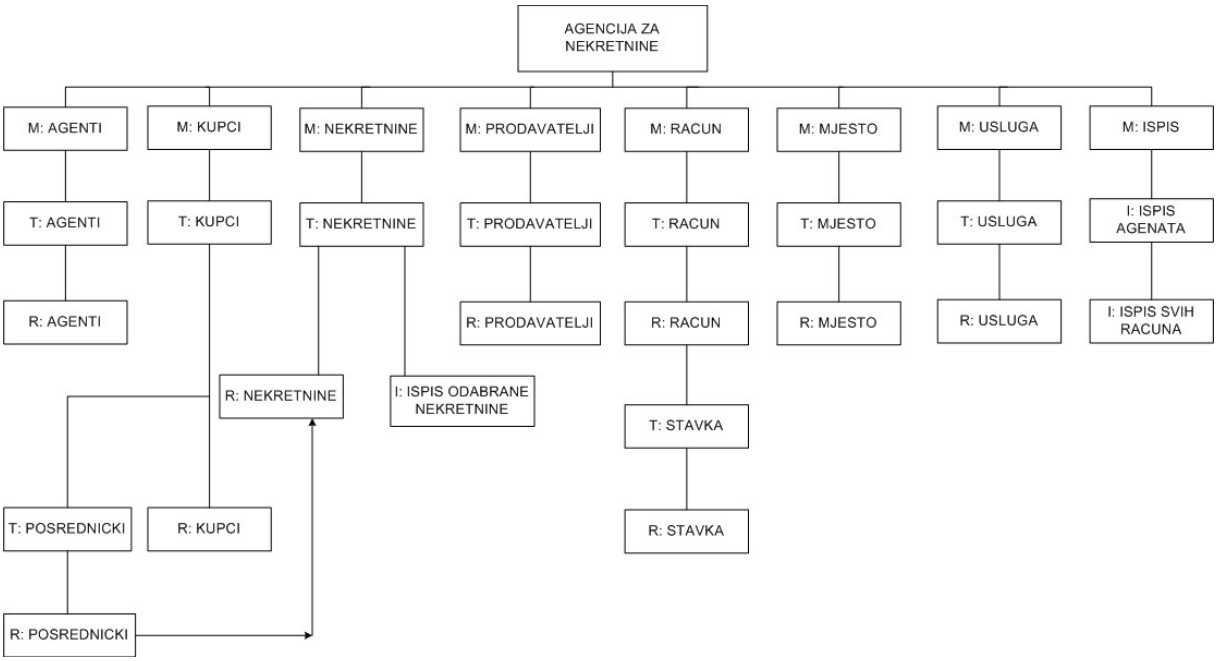


Slika 6 Dodavanje veza

Zadnja stvar koju moramo napraviti da bi rječnik bio gotov, je dodavanje veza između tablica. Veze i brojnosti su definirane u modelima. Klikom na zaokruženu ikonu na slici 6 otvorit će nam se novi prozor „Relationship properties“. Određujemo brojnosti, tablice koje povezujemo i ključeve. Biramo jednu od opcija „Map By Name“ ili „Map By Order“. Ukoliko je veza između dva jaka tipa entiteta „On Update“ će nam bit postavljen na „Cascade“, a „On Delete“ na „Restrict“. Ukoliko povezujemo jedan jak tip entiteta i jedan slabi tip ili agregaciju, tada će nam obje opcije biti postavljene na „Cascade“. Na slici 6 vidimo primjer izrađene veze između tablica PRODAVATELJ i AGENT, također vidimo i sve ostale veze tablice PRODAVATELJ.

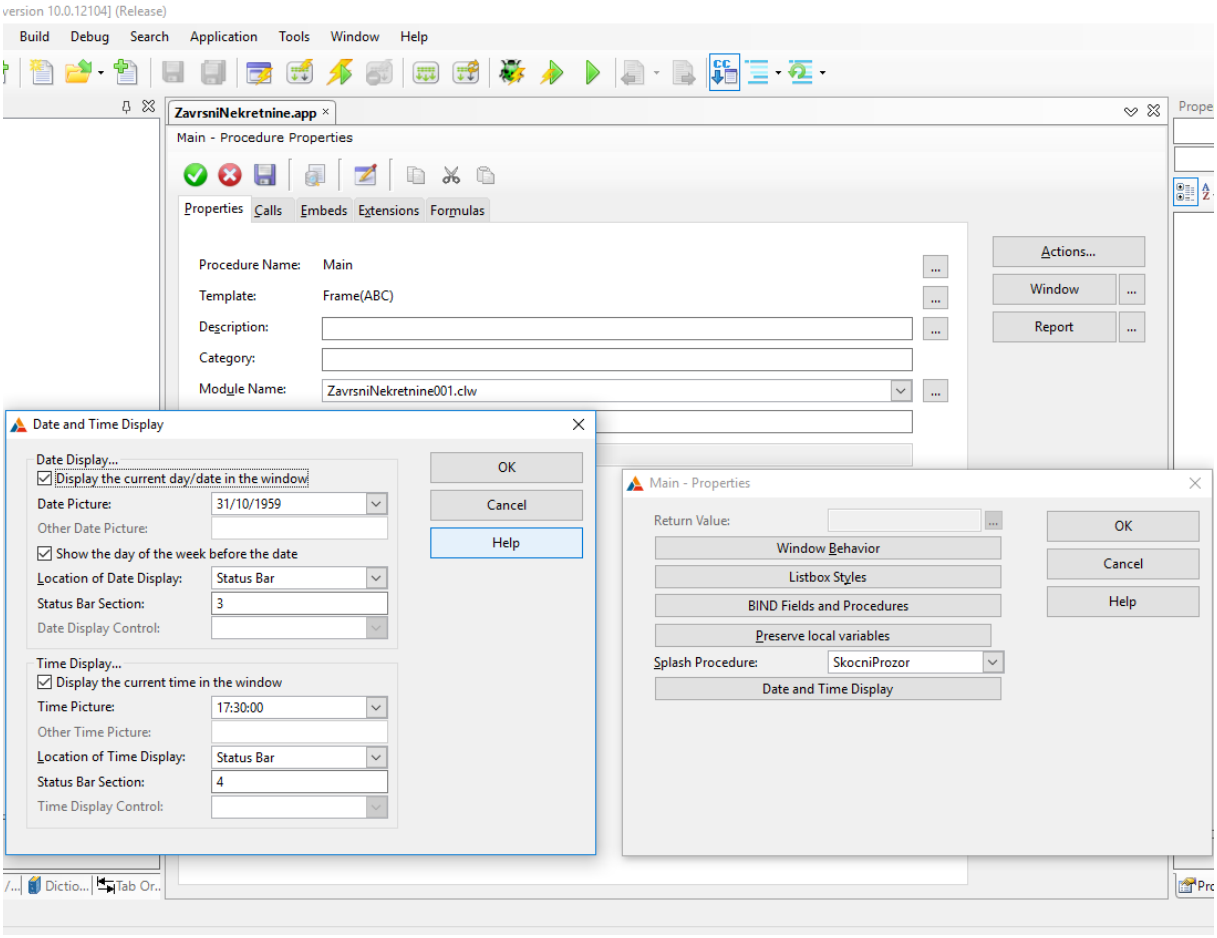
Izrada aplikacije

Prije same izrade aplikacije dobra je praksa izraditi dijagram programskog proizvoda ili skraćeno DPP. DPP nam služi za prikaz strukture aplikacije, odnosno kako su prozori povezani unutar same aplikacije. Prozori unutar DPP-a označeni su slovima M, T, R i I. Slovo M označava meni/izbornik, slovo T nam označava popis procedure, slovo R nam označava procedure za ažuriranje i slovo I nam označava izvještaje. Na slici 7 možemo vidjeti DPP izrađen za poslovnu aplikaciju za agenciju za nekretnine.



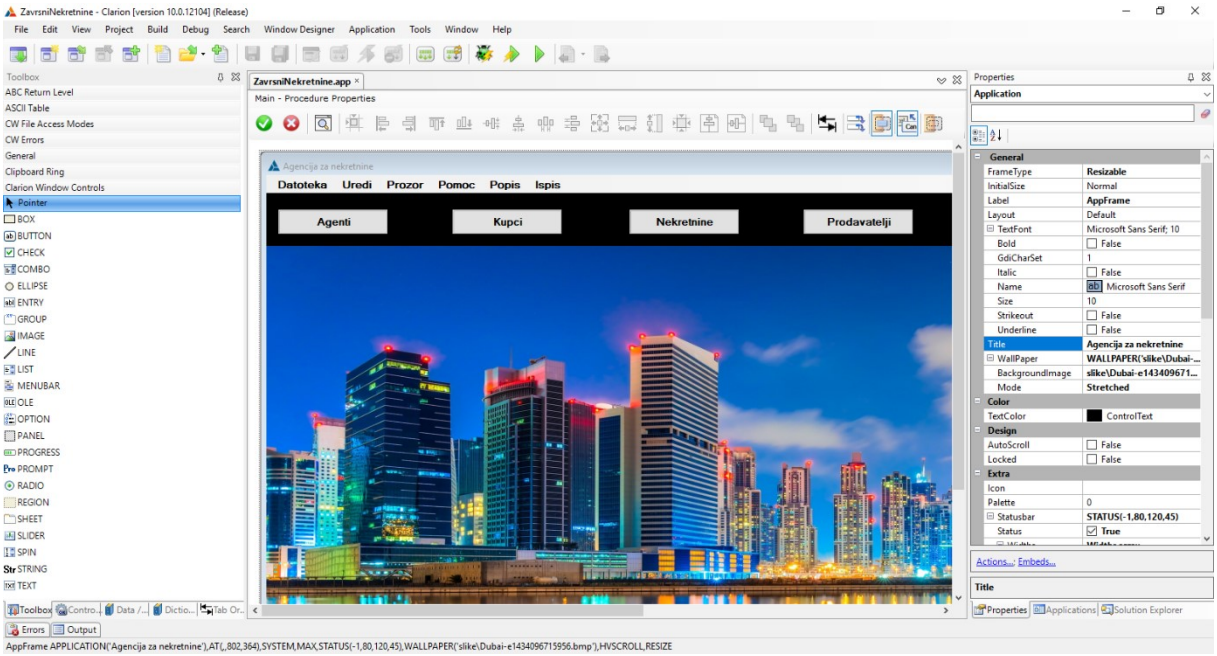
Slika 7 DPP

Nakon što smo izradili rječnik i DPP, slijedi izrada aplikacije. Prvi korak je odabir predloška za naš glavni izbornik, u ovom slučaju odabran je „Default MDI Frame“.



Slika 8 Glavni izbornik – „Actions“

Odabirom predloška otvara nam se „Procedure Properties“. Klikom na gumb „Actions“ možemo definirati „Splash Procedure“ te datum i vrijeme koje će se prikazivati u donjem desnom uglu aplikacije.

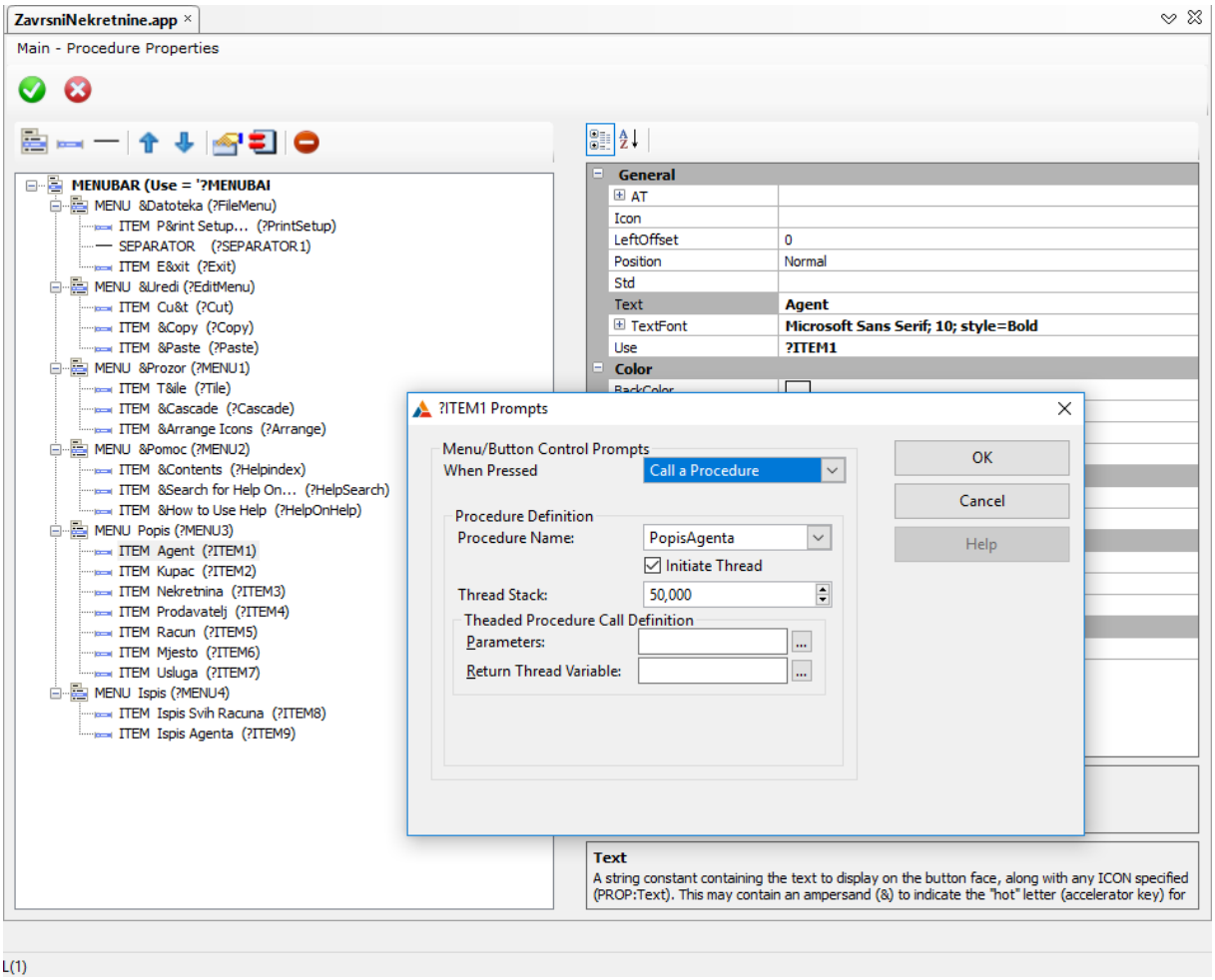


Slika 9 Glavni izbornik – „Window“

Klikom na dugme „Window“ otvara nam se izbornik kao na slici 9. Naziv i pozadinu postavljamo u „Properties“. Slijedi uređivanje menija.

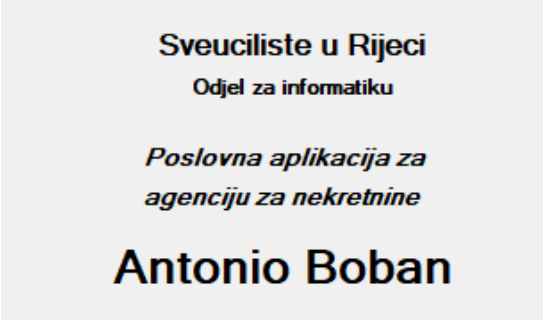
Desnim klikom na meni odabirmo opciju „Edit menu“. Dodajemo novi meni koji smo nazvali „Popis“ u koji dodajemo po jedan „Item“ za svaki jaki tip entiteta. Desnim kllikom na pojedinačni „Item“ i odabirom opcije „Actions“ otvara nam se prozor kao na slici 10. Za opciju „When Pressed“ iz padajućeg izbornika odabiremo „Call a Procedure“, odabiremo naziv procedure te aktiviramo opciju „Initiate Thread“. Na slici 10 možemo vidjeti uređeni meni „Popis“ te sve njegove „Iteme“. Povratkom na glavni prozor možemo vidjeti da se pored već zadanih menija pojavio i naš novi meni „Popis“, a pritiskom na njega otvorit će nam se padajući izbornik u kojem ćemo moći odabrati jednu od procedura Popis Agenta, Popis Nekretnina, Popis Kupaca itd.

Da ne bismo samo preko menija mogli doći do procedura odnosno popisa, na ekran iz „Toolboxa“ ubacujemo „Toolbar“ te ga pozicioniramo ispod menija. Na „Toolbar“, također iz „Toolboxa“, dodajemo po jedan „Button“ za svaki „Item“. Desnim klikom na „Button“ te odabirom opcije „Actions“, otvara nam se isti prozor kao na slici 10. Opcije koje moramo odabrati u ovom slučaju su identične, jedina razlika je što sad već imamo definirane procedure.



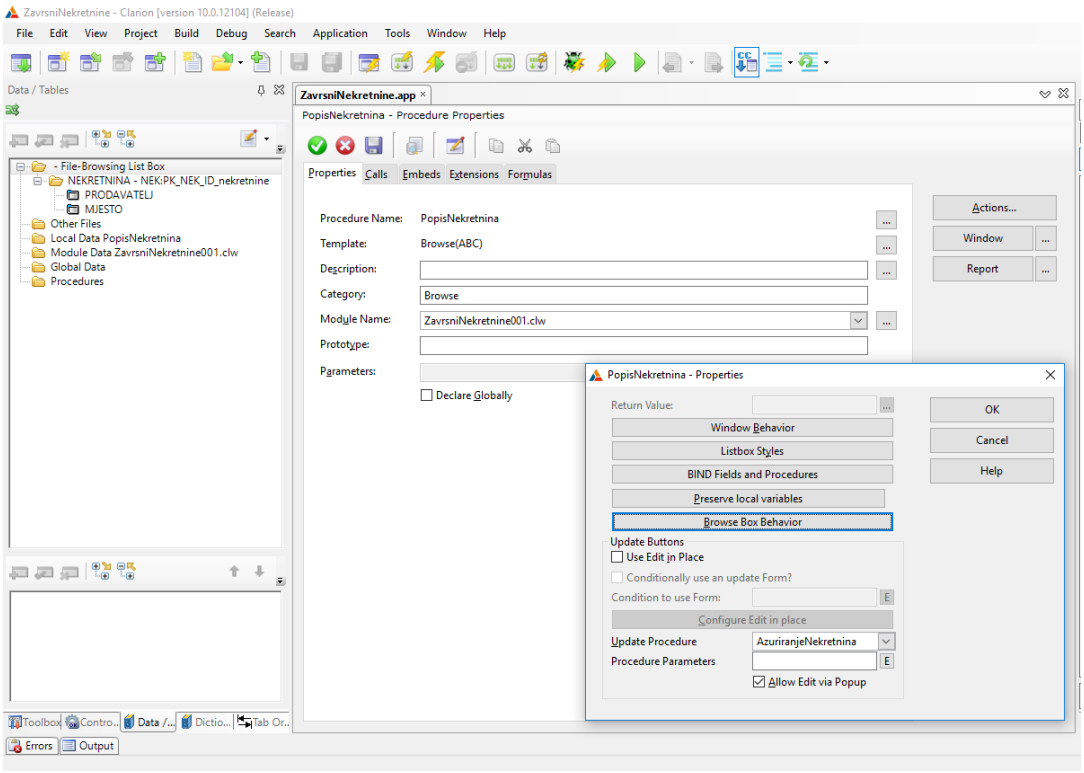
Slika 10 Uređivanje menija

Vraćamo se na početno stablo, u kojem sad imamo sve procedure koje smo prethodno definirali. Prvo ćemo uredit naš skočni prozor. Skočni prozor je prozor koji se pojavljuje prilikom pokretanja aplikacije u trajanju od pet sekundi. Odabrani predložak za skočni prozor je „Default Splash Window“. Vrijeme trajanja skočnog prozora možemo postaviti u „Actions“. Odabirom gumba „Window“ možemo urediti naš skočni prozor. Na slici 11 možemo vidjeti skočni prozor kakav se pojavljuje u aplikaciji.



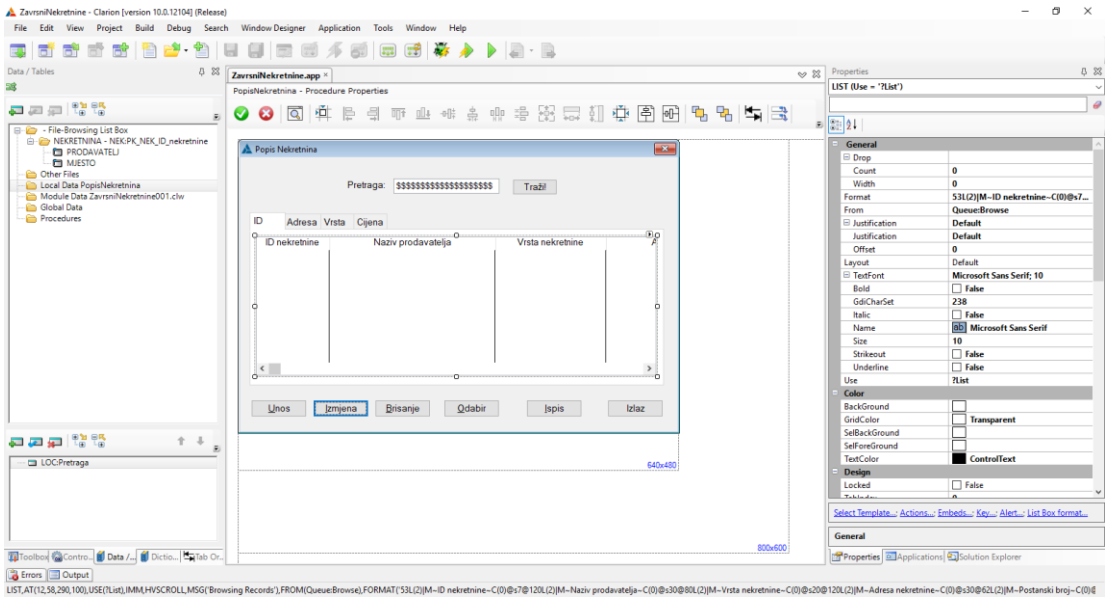
Slika 11 Skočni prozor

Sljedeći korak je uređivanje procedura. Odabrani predložak za sve „Popis“ procedure je „Browse with Update and Select“. Klikom na gumb „Actions“ otvara nam se novi prozor u kojem postavljamo „Update Procedure“, odnosno proceduru u kojoj ćemo kasnije dodavati i izmjenjivati podatke. Ukoliko želimo podatke dodavati i izmjenjivati na licu mjesta, bez da nam se otvara novi prozor, aktivirat ćemo opciju „Use Edit in Place“.



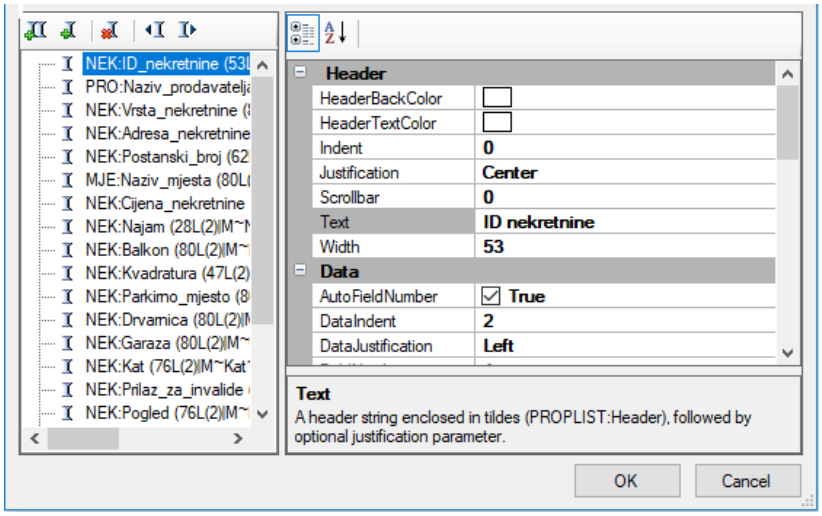
Slika 12 Popis – „Actions“

Odabiremo opciju „Window“. Otvara nam se prozor kao na slici 13.



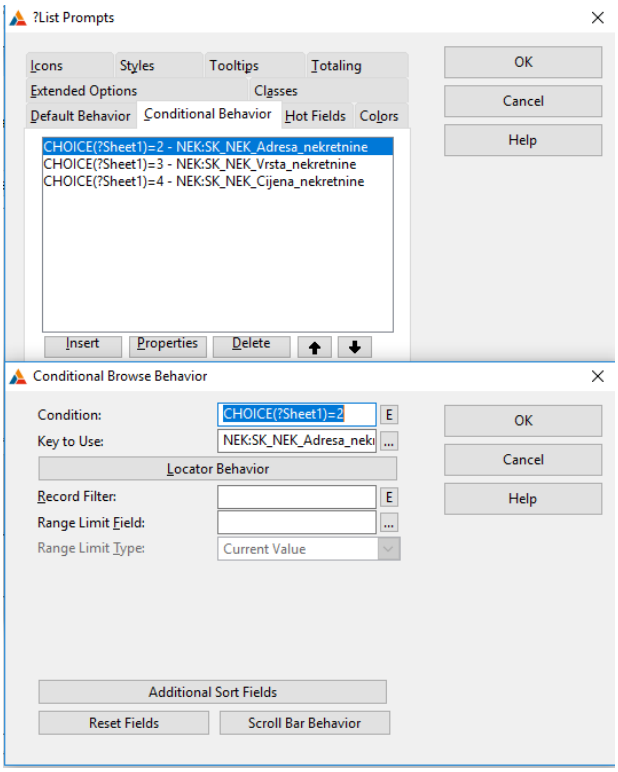
Slika 13 Popis – „Window“

Preimenovanje radimo u „Properties“. Odlazimo u „Data/Tables“ te odabiremo tablicu s kojom mislimo raditi i njezin primarni ključ. U ovom primjeru to će biti tablica NEKRETNINE. Desnim klikom na „Listbox“ otvara nam se novi prozor u kojem odabiremo opciju „ListBox format“ te nam se otvara prozor kao na slici 14. U njemu dodajemo atribute koje želimo vidjeti na našem „Listboxu“.



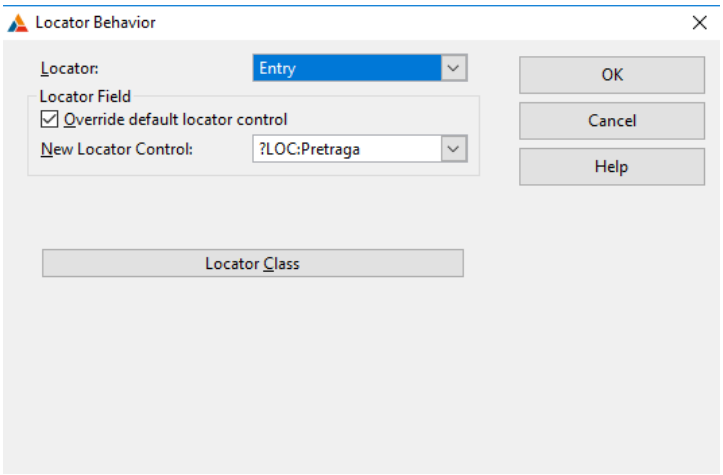
Slika 14 „ListBox format“

Prilikom pregleda podataka moguće je i sortirati podatke. To smo postigli uz pomoć „Sheet“ opcije iz „Toolboxa“ te dodavanjem novih „Tab“-ova. Desnim klikom na „Listbox“ otvara nam se novi prozor, odabiremo opciju „Actions“ te pod „Conditional Behavior“ biramo opciju „Insert“. Otvara nam se novi prozor u kojem pod „Condition“ biramo „Tab“, dok pod „Key to Use“ biramo jedan od ključeva za sortiranje koji smo prethodno napravili u rječniku. Primjer izrade uvjeta možemo vidjeti na slici 15. Prvi „Tab“ uvijek sortira po primarnom ključu tako da u ovom slučaju mi moramo napraviti još tri uvjeta za sortiranje pošto smo ih toliko i predvidjeli u rječniku.



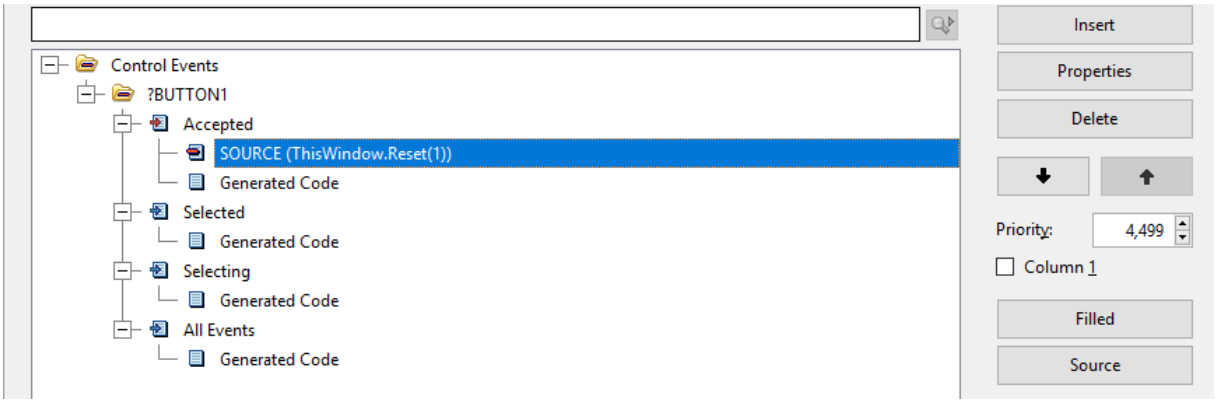
Slika 15 Sortiranje po Tabovima

Također, moguće je i pretraživanje podataka. To smo postigli uz pomoć jedne lokalne varijable koju smo nazvali „Pretraga“. Pretraživati možemo ovisno o trenutno odabranom „Tab“-u. Desnim klikom na „Listbox“ odabiremo opciju „Actions“ te u opcijama „Default Behavior“ i „Conditional Behavior“ svagdje gdje se pojavljuje „Locator Behavior“ činimo ono što je prikazano na slici 16.



Slika 16 „Locator Behavior“

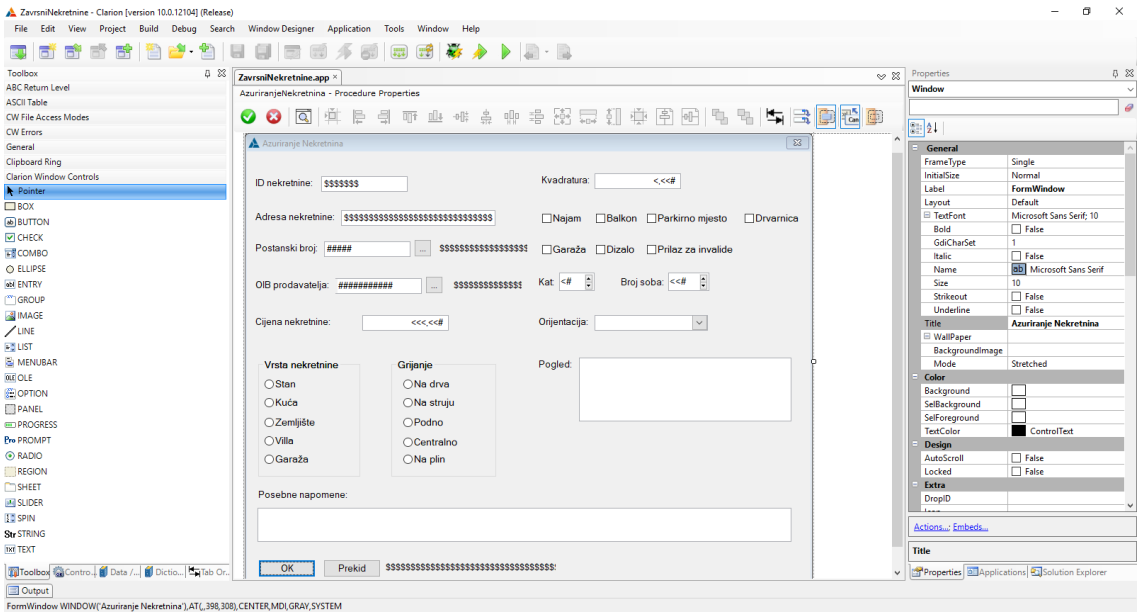
Dodali smo i jedan gumb na kojem piše „Traži!“ koji nam služi za pokretanje i resetiranje pretrage. Desnim klikom na gumb otvara nam se prozor u kojem odabiremo opciju „Embeds“, te u polje „Accepted“ upisujemo liniju koda prikazanu na slici 17.



Slika 17 Gumb za pretraživanje

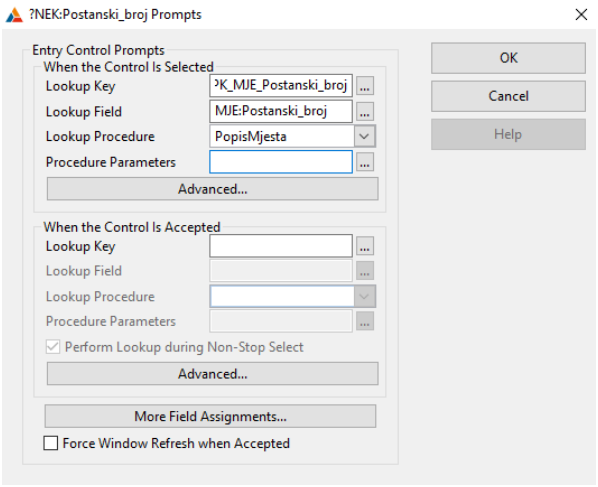
Sve ostale „Popis“ procedure napravljene su po istom principu koji je prethodno opisan.

Slijedi nam uređivanje procedura za ažuriranje podataka. Odabrani predložak za sve procedure za ažuriranje podataka je „FORM(add/edit/delete)“. Klikom na gumb „Window“ otvara nam se prozor kao na slici 18.



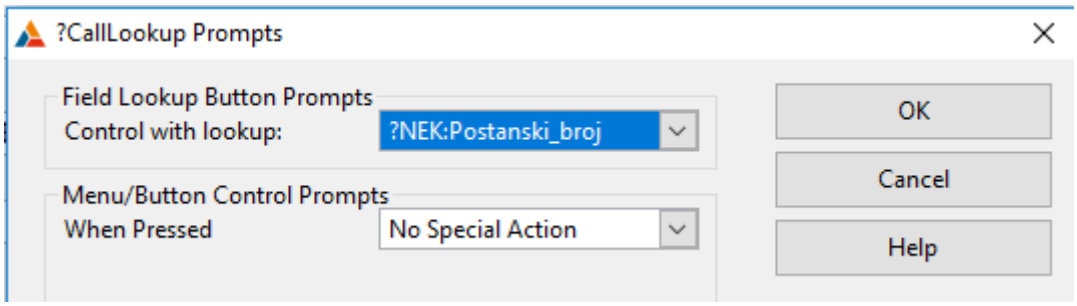
Slika 18 Ažuriranje – „Window“

Prvi korak je odabir u „Data/Tables“ tablice, te ubacivanje svih atributa u glavni prozor. Na slici 18 možemo vidjeti da su korištene razne vrste polja za unos i odabir. Korištene opcije za unos u ovoj aplikaciji su „Text“, „Radio“, „Spin“, „Check“ i „List“. Ukoliko vrijednost jednog od atributa se nalazi u nekoj drugoj tablici kao npr. „Poštanski broj“ čija se vrijednost nalazi u tablici MJESTO ili „OIB prodavatelja“ čija se vrijednost nalazi u tablici PRODAVATELJ, odabirom tog polja za unos otvorit će nam se popis iz kojeg ćemo odabrati vrijednost koju želimo unijeti. To smo postigli na način da smo odredili „Lookup“ proceduru, kao na slici 19.



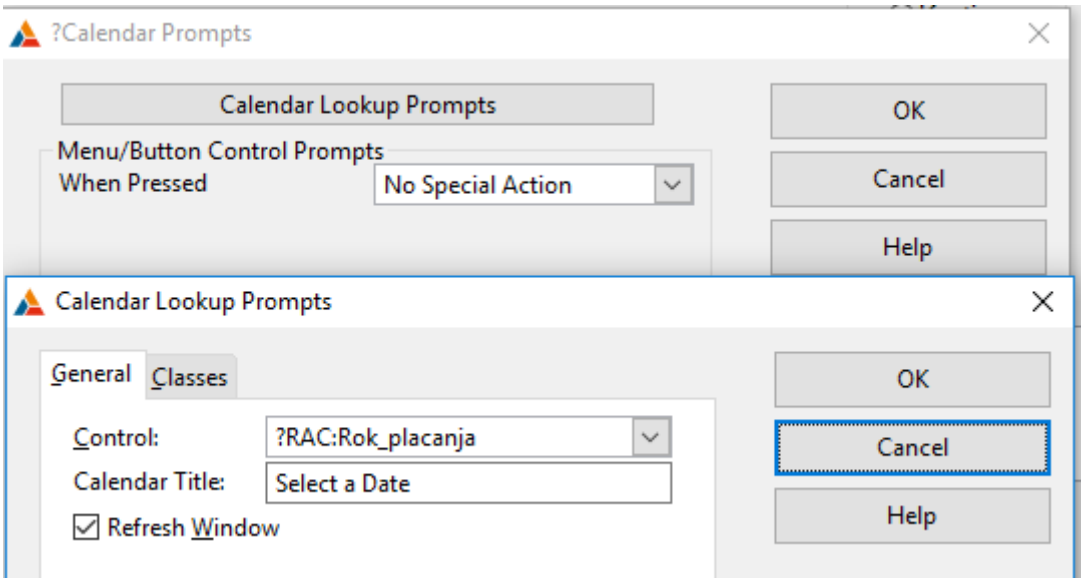
Slika 19 „Lookup“ procedura

Također, iz „Control Templates“ dodan je i „Field Lookup Button“ uz pomoć kojeg također možemo doći do popisa iz kojeg moramo odabrati jednu od vrijednosti koju će poprimiti naš atribut.



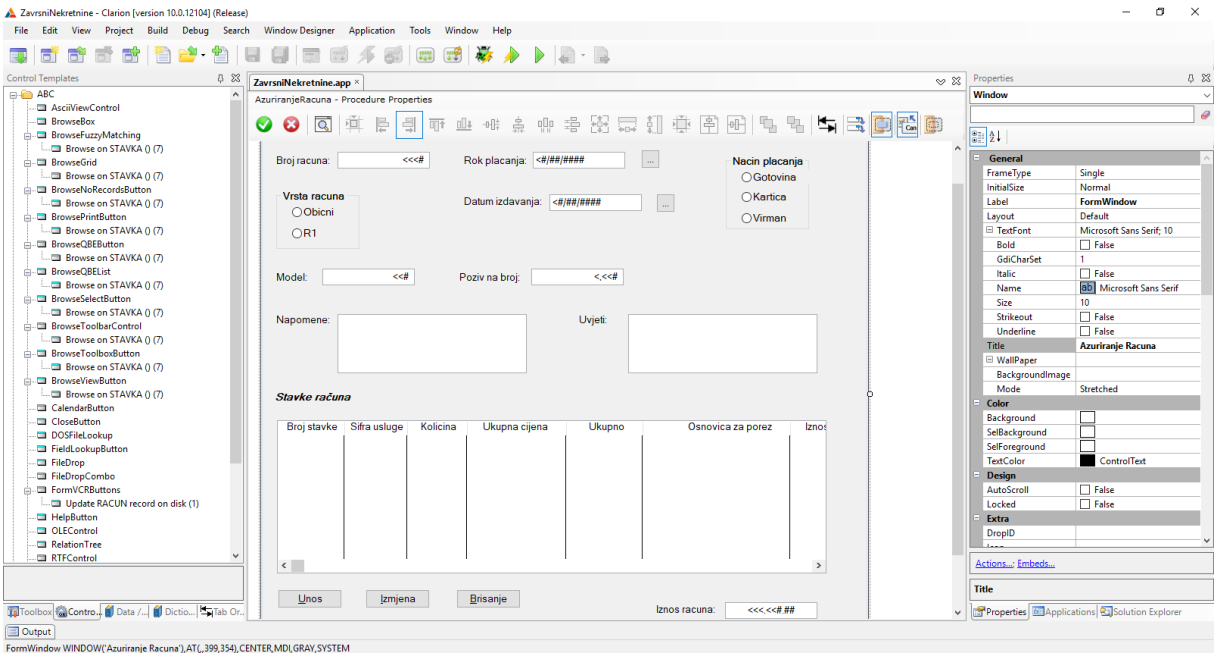
Slika 20 „Lookup Button“

U slučaju kada se od nas traži unos datuma, osim ručnog unosa najbolja praksa je dodavanje „Calendar Button“ iz „Control Templates“, gdje nam se klikom na isti otvara kalendar iz kojeg izaberemo datum.



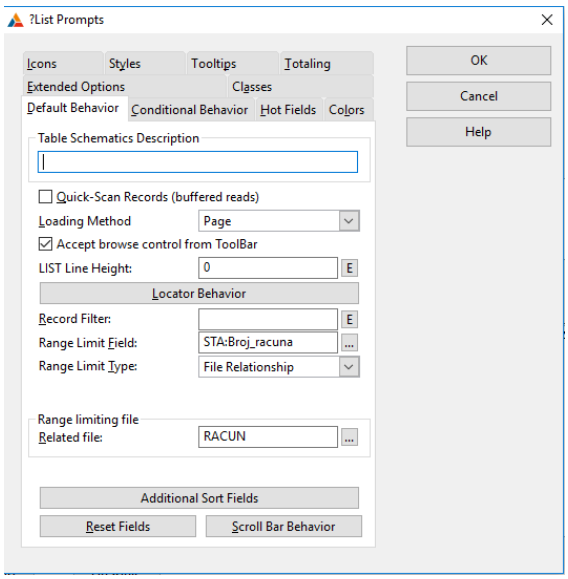
Slika 21 „Calendar Button“

U slučaju kada naš model ima slabe tipove entiteta ili agregacije, moramo dodati novi „Listbox“. Na slici 22 možemo vidjeti dodan novi „Listbox“ koji se nalazi u proceduri „Ažuriranje Računa“, on će nam poslužiti za pregled svih podataka tablice STAVKA. Tablica STAVKA je slabi tip entiteta tablice RAČUN i zbog toga smo novi „Listbox“ dodali u jednu od procedura tablice RAČUN.



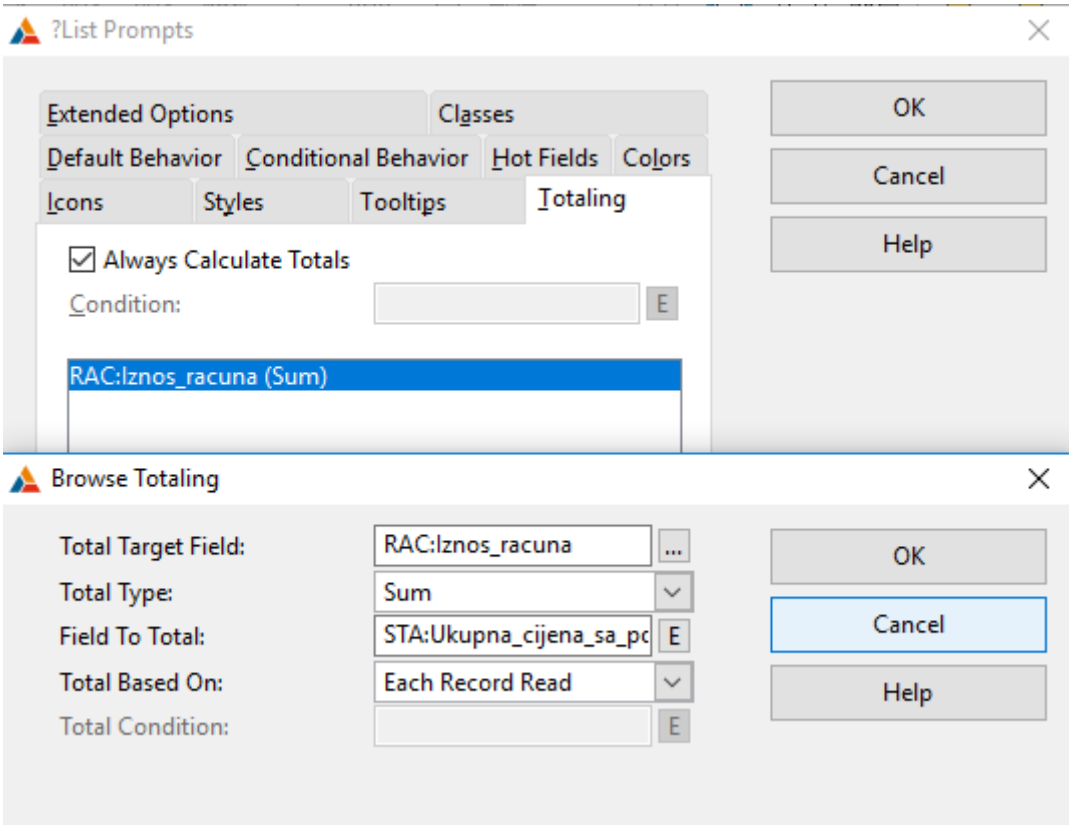
Slika 22 Dodavanje novog „Listbox“-a

Način dodavanja novog „Listbox“-a je poprilično jednostavan. Iz „Control Templates“ izaberemo „BrowseBox“, te ga povučemo na ekran. Nakon što smo to napravili moramo u „Data/Tables“ odrediti tablicu za koju ćemo uređivati naš novi „Listbox“. Odabirom opcije „List Box format“ uređujemo naš „Listbox“. U slučaju kada dodajemo novi „Browsebox“ moramo dodati i odgovarajuće gumbe. Njih isto možemo pronaći u „Control Templates“ pod nazivom „Browse Update Buttons“, povlačenjem ih dodajemo na ekran. Desnim klikom na jedan od dugmadi i odabirom opcije „Actions“ otvara nam se novi prozor u kojem određujemo „Update Procedure“, odnosno proceduru za ažuriranje podataka.



Slika 23 „Range Limit Field“

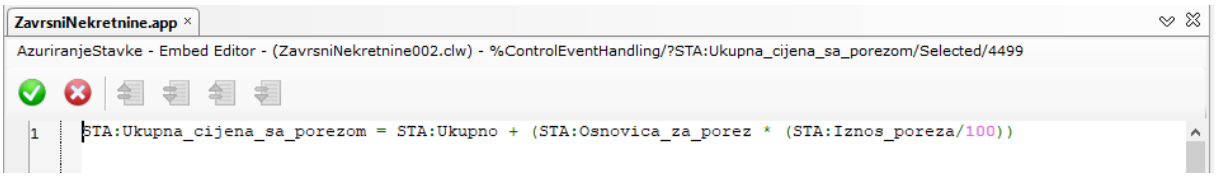
Ukoliko želimo da nam se na „Listbox“-u „Popis Stavki“ prikazuju točno određeni podaci, odnosno podaci koji su vezani za točno određeni račun, moramo odrediti „Range Limit Field“. Desnim klikom na „Listbox“ i odabirom opcije „Actions“ otvara nam se prozor gdje to možemo učiniti. Primjer određivanja „Range Limit Field“-a prikazan je na slici 23.



Slika 24 „Totaling“

Atribut „Iznos računa“ tablice RAČUN je zbroj svih stavki za taj račun. Taj atribut ne upisujemo mi nego se vrijednost automatski upisuje. Desnim klikom na „Listbox“ i odabirom opcije „Actions“ otvara nam se prozor gdje to možemo učiniti. Primjer možemo vidjeti na slici 24.

Također, možemo dodati i vlastite operacije koje izračunavaju vrijednost nekog atributa. Na slici 25 možemo vidjeti primjer napisane operacije za računanje sveukupnog cijene sa porezom.



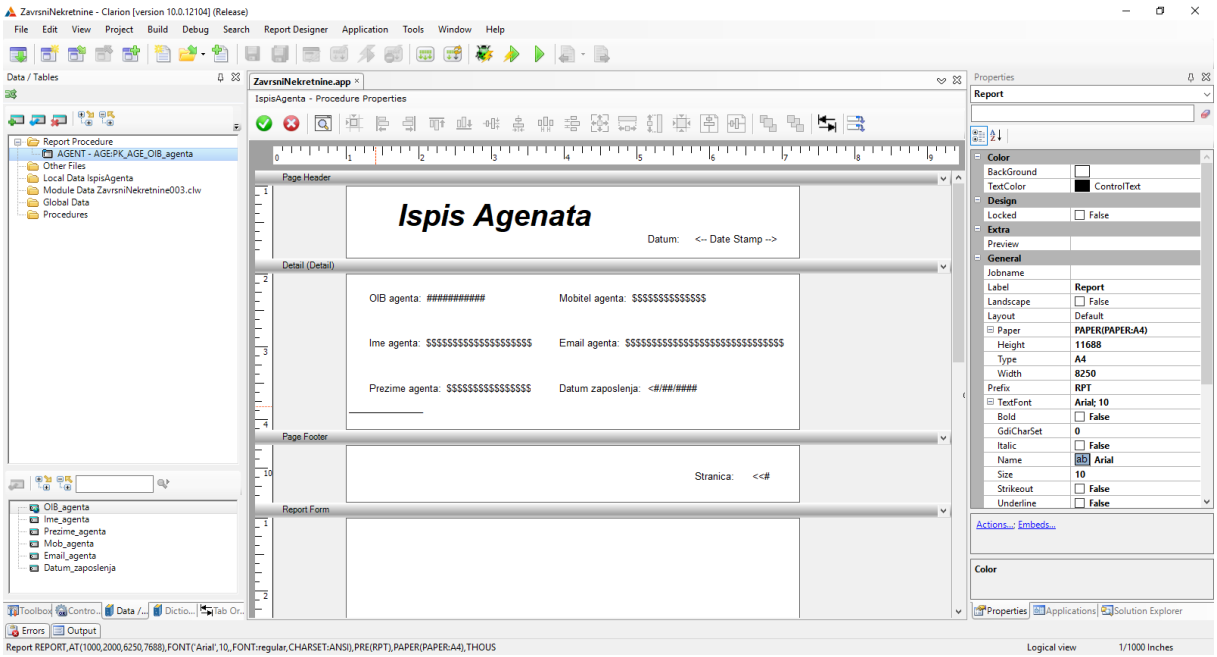
Slika 25 Računanje sveukupne cijene

Slika 26 IF petlja

Slika 26 nam prikazuje upotrebu IF petlje. Ukoliko za atribut „Rok plaćanja“ koji se nalazi u tablici RAČUN odaberemo datum koji je stariji od današnjeg, prilikom unosa javit će nam se obavijest da smo unijeli stari datum za rok plaćanja.

Sve procedure za ažuriranje podataka napravljene su po principu koji je prethodno opisan.

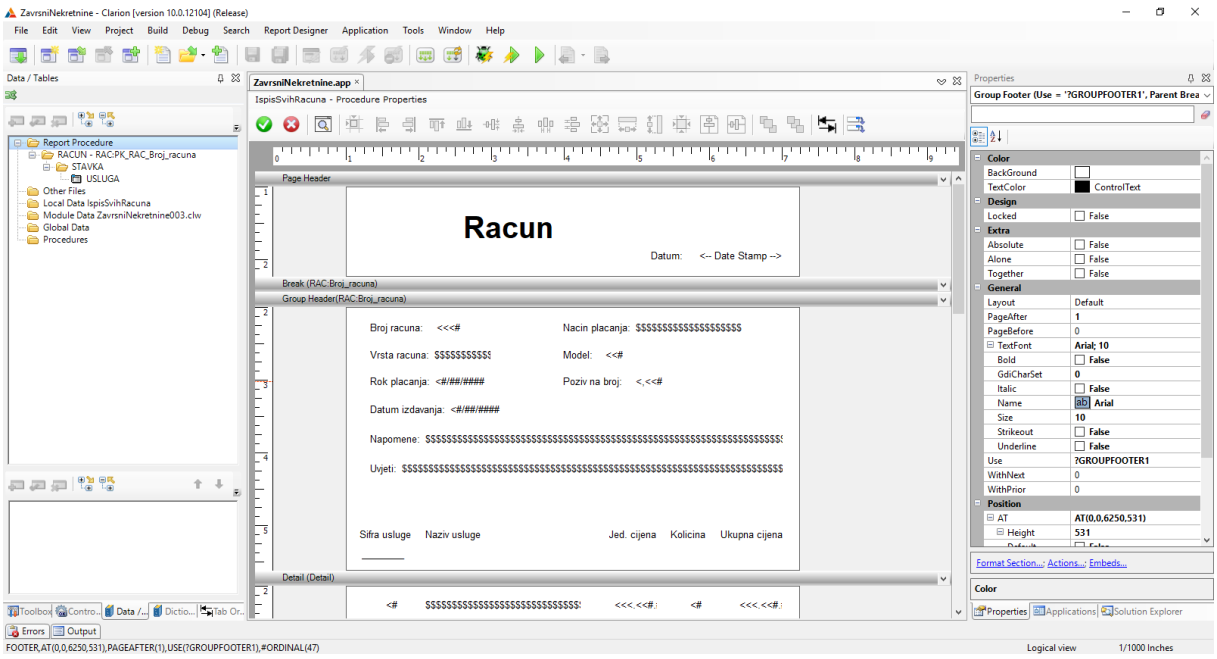
Slijedi nam izrada izvještaja. Izradu izvještaja započinjemo izradom menija koji smo nazvali „Ispis“. Postupak je isti kao i kada smo radili meni „Popis“, također na slici 10 možemo vidjeti uređeni meni „Ispis“ i sve njegove „Iteme“. Postoje tri vrste izvještaja, a oni su jednostavni izvještaj, složeni izvještaj i izvještaj po gumbu.



Slika 27 Jednostavno izvješće

Izrada jednostavnog izvještaja, kako mu samo ime kaže, je poprilično jednostavan proces. Naše jednostavno izvješće sastoji se od jedne tablice AGENT i ispisuje nam sve podatke o svakom agentu koji se trenutno nalazi u bazi. Za sve izvještaje u ovom radu korišten je predložak „A4 - Portrait“. Klikom na gumb „Report“ otvara nam se prozor za uređivanje našeg izvještaja kao na slici 27. Najprije odaberemo u „Data/Tables“ tablicu za koju radimo izvještaj i njezin primarni ključ. Za broj stranice i datum iz „Control Templates“-a koristimo opcije „Report Page Number“ i „Report Date Stamp“. Nakon toga u prozor ubacujemo atribute za koje želimo da budu ispisani u izvještaju, te uz pomoć „Toolbox“-a uređujemo naš izvještaj.

Složeno izvješće je nešto kompliciranije za izradu od jednostavnog izvješća. Princip izrade je isti, jedina je razlika što se složeno izvješće sastoji od više hijerarhijski povezanih tablica. Naše složeno izvješće sastoji se od tablica RAČUN, STAVKA, USLUGA.

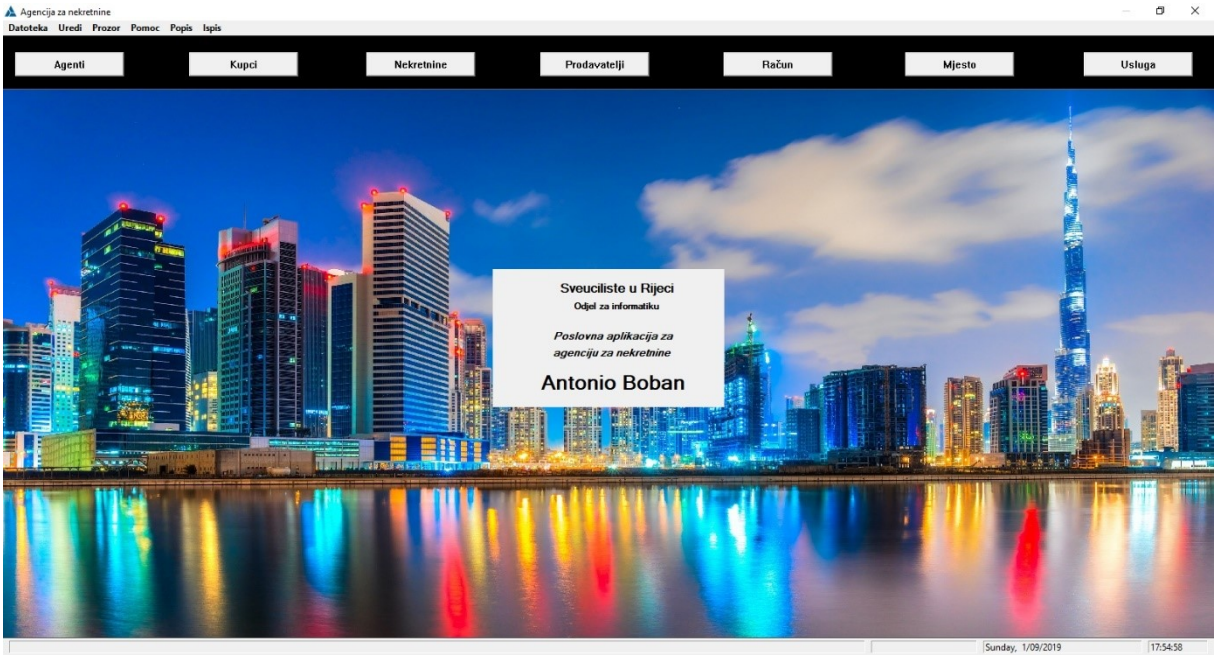


Slika 28 Složeno izvješće

Izvješće po gumbu može biti ili jednostavno ili složeno izvješće. U ovom radu izvješće po gumbu je složeno izvješće, koje će nam ispisivati sve informacije o odabranoj nekretnini. Sastoji se od tri tablice, a to su NEKRETNINA, PRODAVATELJ i MJESTO. Gumb „Ispiši“ nalazi se u „Control Templates“ pod nazivom „Browse Print Button“ kojeg smo povukli na ekran i preimenovali. Njega smo pozicionirali unutar procedure „Popis Nekretnina“ kao što možemo vidjeti na slici 13. Desnim klikom na gumb i odabirom opcije „Actions“ otvara nam se novi prozor u kojem odabiremo željenu proceduru.

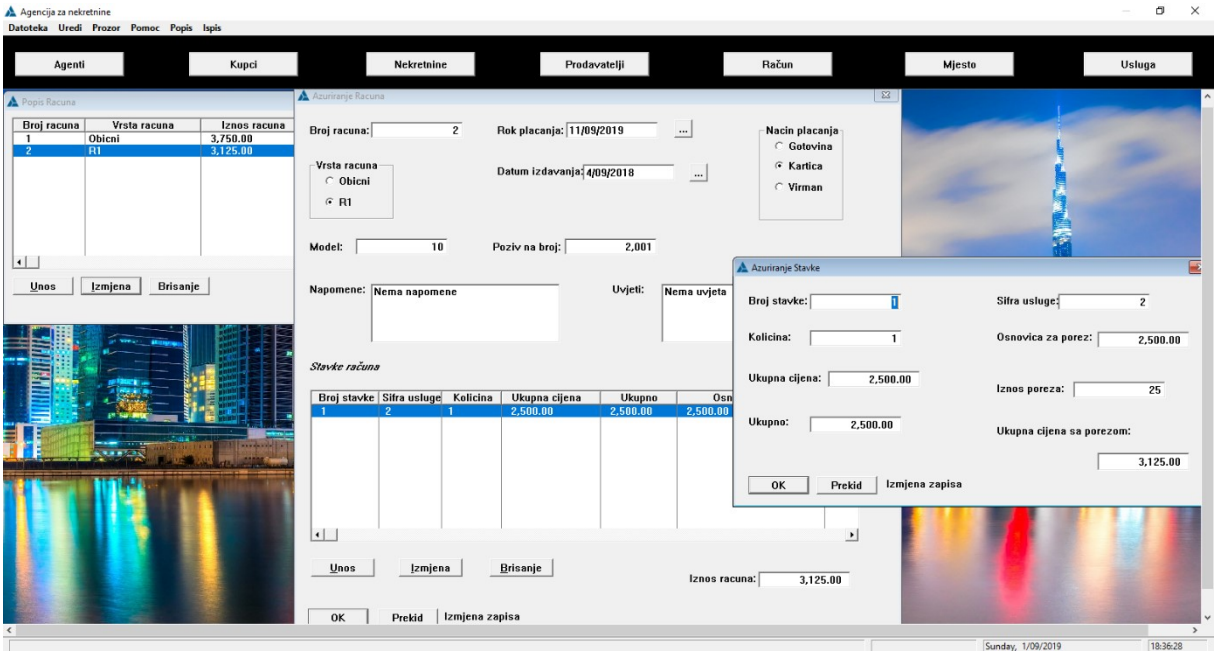
Gotova aplikacija

Prilikom pokretanja aplikacije otvorit će nam se početni prozor kao na slici 29. Ovdje možemo pristupiti svim tablicama, podacima i izvješćima preko izbornika i dugmadi koje smo napravili.



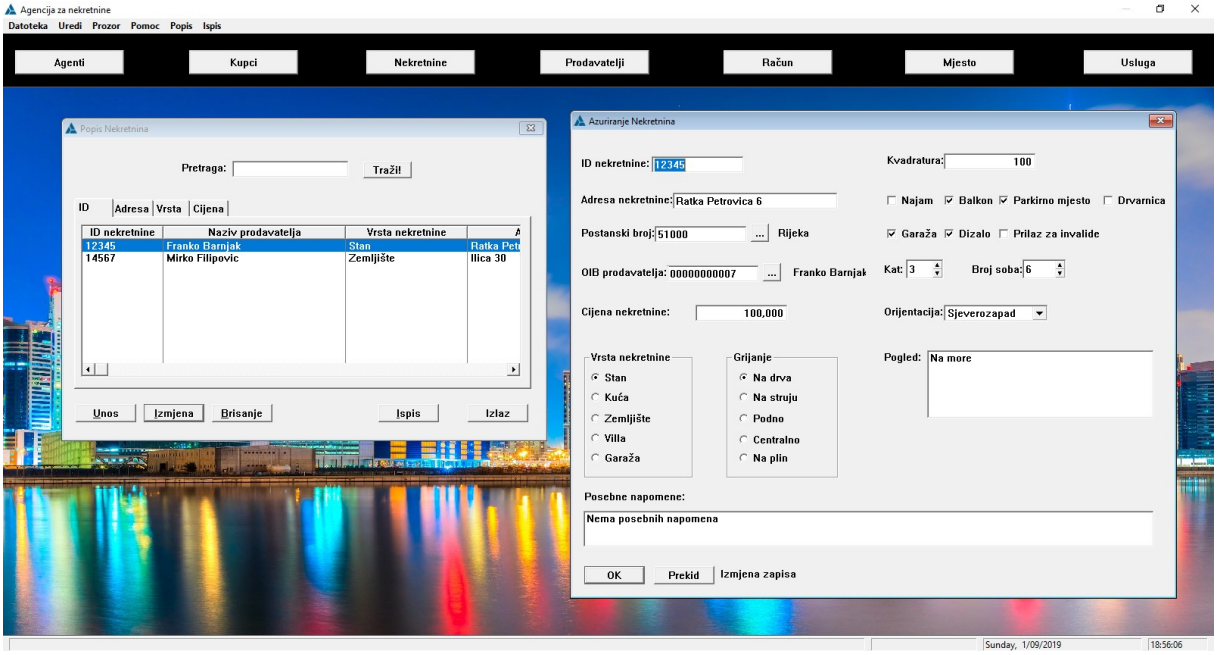
Slika 29 Početni prozor

Klikom na gumb „Račun“ otvara nam se krajnji lijevi prozor na slici 30, imena „Popis računa“. On nam služi za pregled svih računa koji se nalaze u bazi podataka. Klikom na dugme „Unos“ ili „Izmjena“ otvara nam se središnji prozor na slici 30, „Ažuriranje računa“. On nam služi za unos novih i izmjenu već postojećih podataka vezanih za račun. Pošto je „STAVKA“ slabi tip entiteta od jakog tipa entiteta „RAČUN“ u ovaj prozor smo dodali i popis svih stavki vezanih uz odgovarajući račun. Klikom na jedan od dugmadi „Unos“ ili „Izmjena“ otvara nam se novi prozor „Ažuriranje stavki“, na slici 30 krajnji desni. Agregacija „POSREDNIČKI“ riješena je na isti način kao i slabi tip entiteta „STAVKA“, samo se njezin popis nalazi zajedno sa popisom svih kupaca.



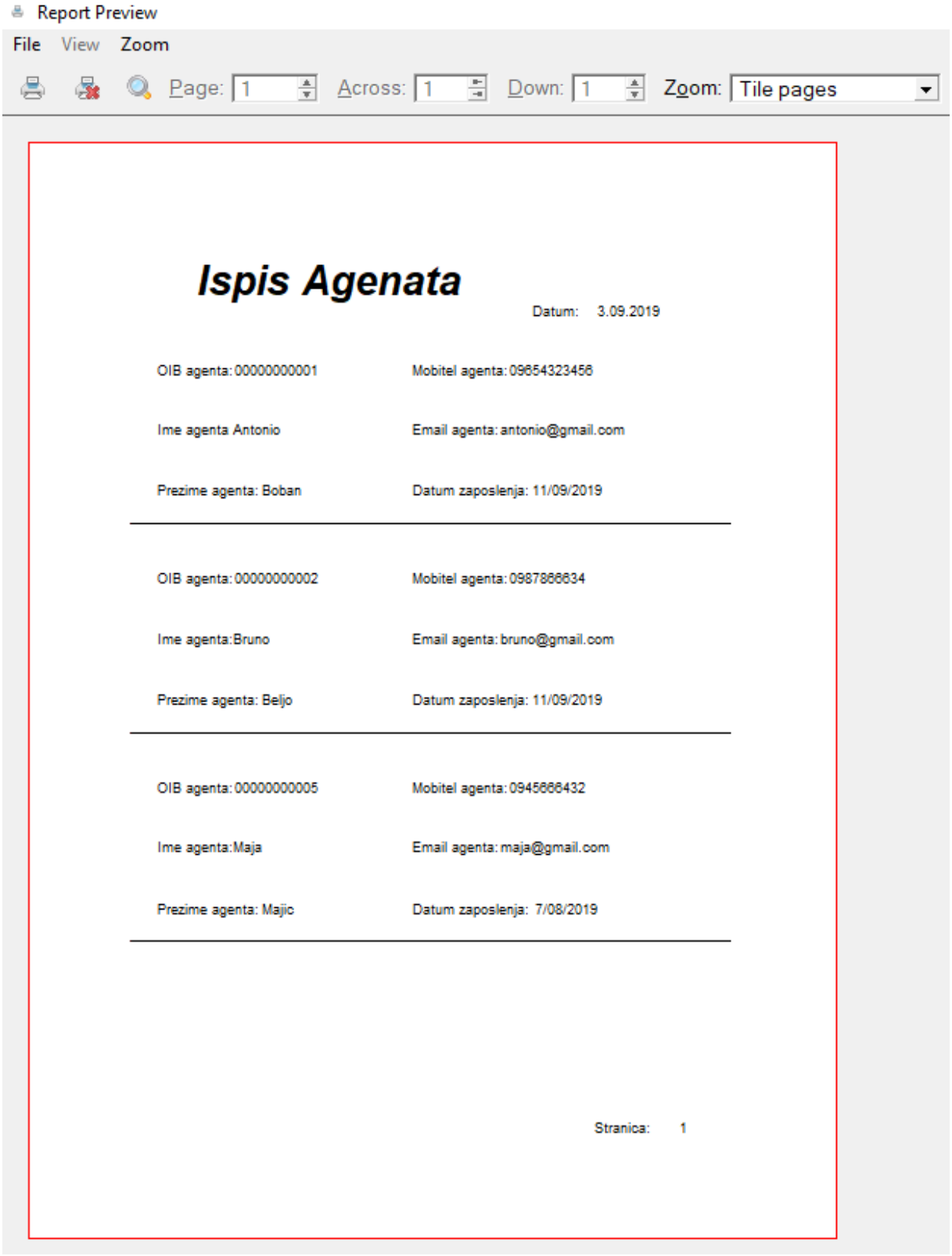
Slika 30 „Račun“ i „Stavka“

Na slici 31 možemo vidjeti ekrane „Popis nekretnina“ i „Ažuriranje nekretnina“. Na ekranu „Popis nekretnina“ možemo vidjeti „Tab“-ove koji nam služe za sortiranje podataka po željenom redoslijedu. Također, možemo vidjeti i tražilicu uz pomoć koje možemo pronaći odgovarajući podatak na brži i jednostavniji način, te gumb „Ispis“ čijim klikom nam se otvara izvještaj koji sadrži sve podatke o odabranoj nekretnini. Na ekranu „Ažuriranje nekretnina“ možemo vidjeti upotrebu raznih vrsta polja.



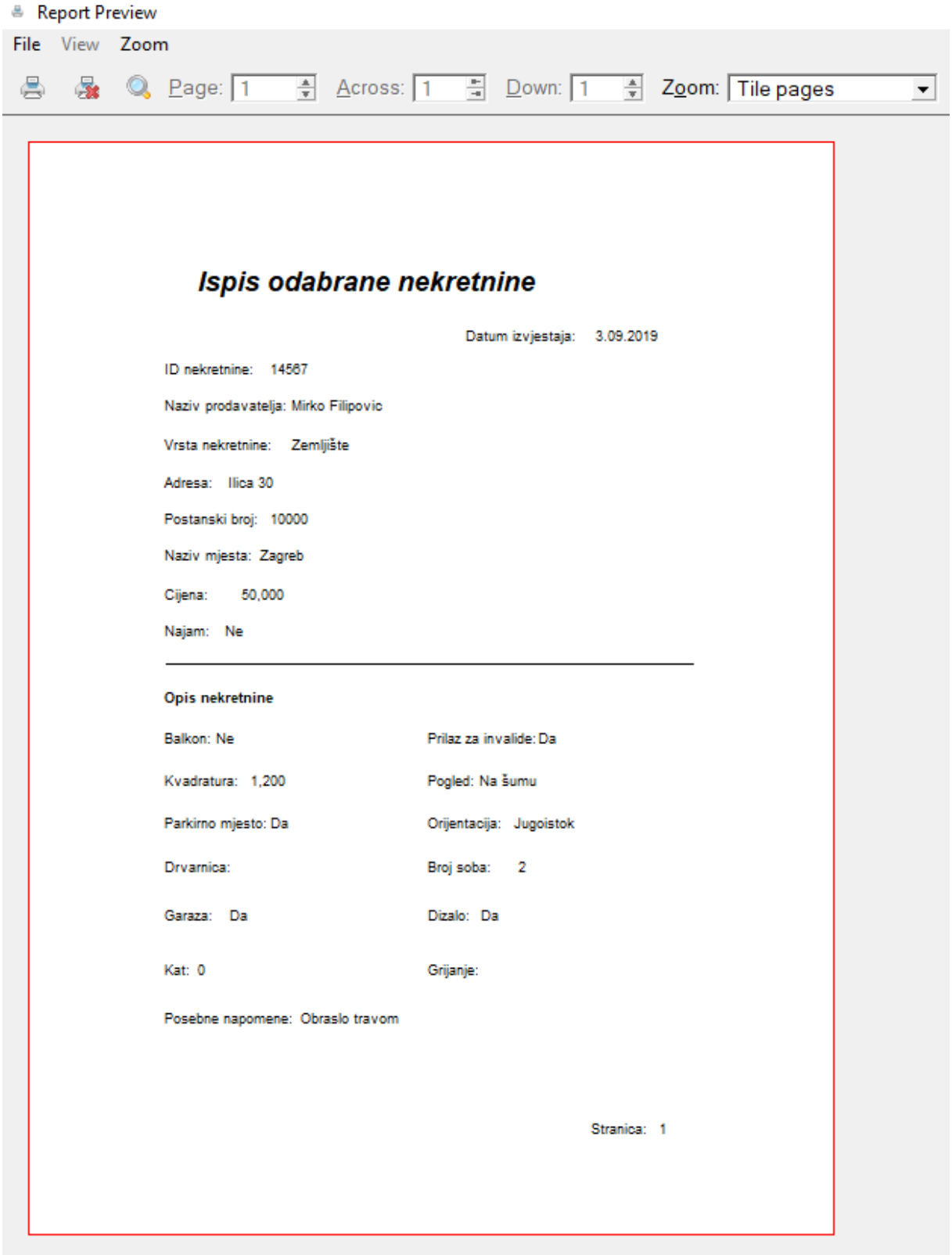
Slika 31 „Nekretnina“

Na slici 32 možemo vidjeti kako nam izgleda jednostavno izvješće koje ispisuje sve agente u bazi podataka.



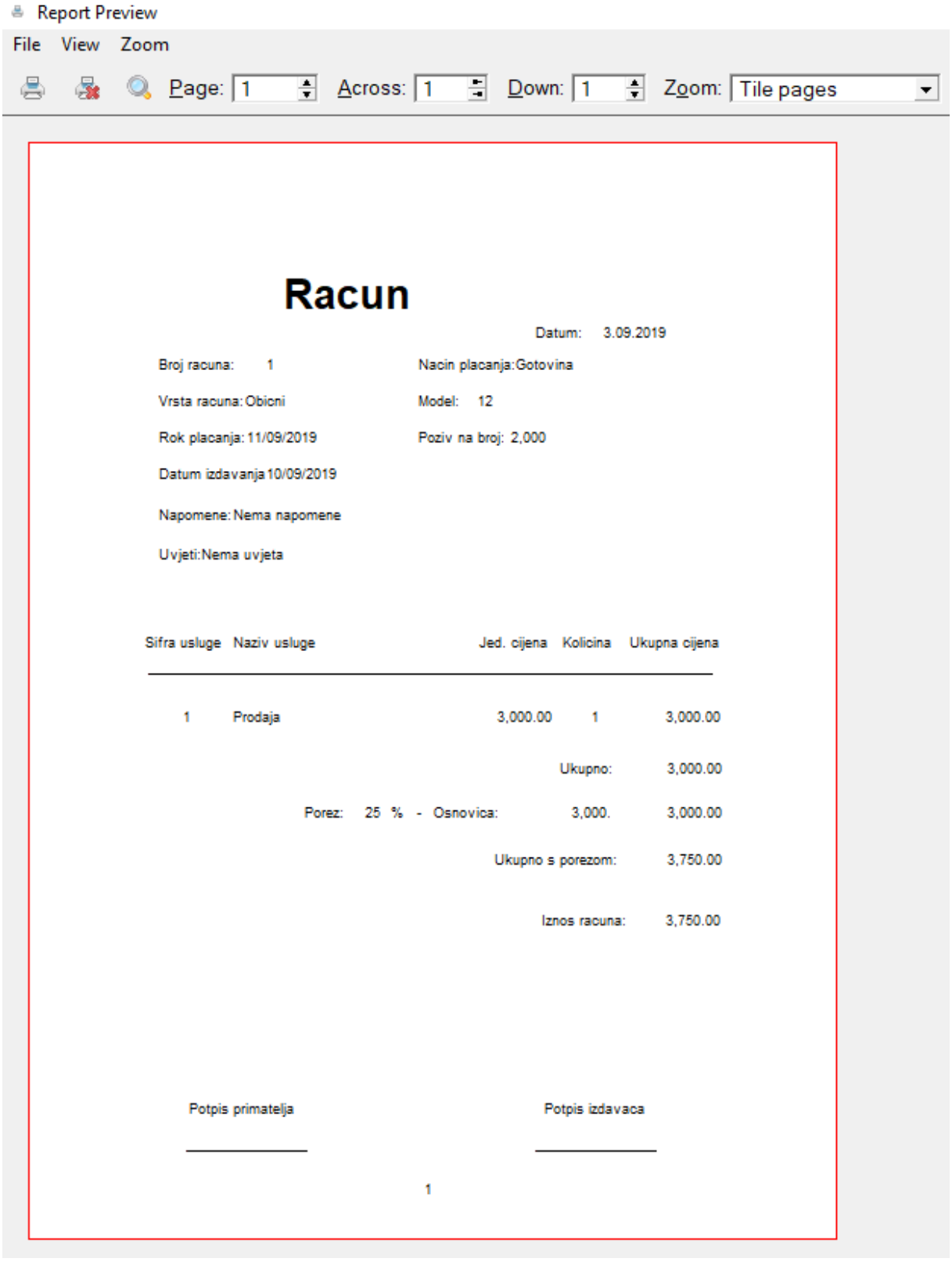
Slika 32 „Ispis Agenata“

Na slici 33 možemo vidjeti kako nam izgleda izvješće „po gumbu“ koje ispisuje odabranu nekretninu u bazi podataka.



Slika 33 „Ispis odabrane nekretnine“

Na slici 34 možemo vidjeti kako nam izgleda složeno izvješće koje ispisuje sve račune unutar baze podataka.



Slika 34 „Račun“

Zaključak

Poslovnu aplikaciju za agenciju za nekretnine izabrao sam iz razloga jer moji roditelji rade kao agenti za nekretnine i zbog toga sam imao dobar uvid u način poslovanja agencije.

Programski alat Clarion kao i svaki program ima svojih nedostataka i prednosti. Jedan od većih nedostataka je često rušenje programa, što novim korisnicima može predstavljati veliki problem prilikom korištenja alata. Meni osobno to nije predstavljalo neki problem, jer ukoliko znate raditi u Clarionu možete rušenja svesti na minimum. Najveća prednost programskog alata Clarion je njegova jednostavnost korištenja, preglednost i funkcionalnost. Korištenjem alata kao što je Clarion uštedimo jako puno vremena prilikom izrade jedne ovakve aplikacije, pošto alat sam generira kod te ga mi ne moramo upisivati ručno.

Cilj ovog završnog rada je bilo stvaranje poslovne aplikacije uz pomoć koje će krajnji korisnici, u ovom slučaju agenti neke agencije za nekretnine, što jednostavnije pregledavati, upisivati, izmjenjivati i brisati podatke iz baze podataka.

Prilikom izrade ovog završnog rada shvatio sam da ključ kvalitetne izrade leži u razgovoru sa krajnjim kupcem i korisnicima. Uz pomoć razgovora možemo saznati sve potrebne informacije vezane uz poslovnu organizaciju za koju se poslovna aplikacija izrađuje, te uz pomoć tih informacija napraviti bazu za svaku poslovnu aplikaciju, a to su modeli entiteti-veze i relacijski model.

U budućnosti mi je plan nadograditi moju poslovnu aplikaciju za agenciju za nekretnine, tako da aplikacija uistinu i postane dio jednog poslovnog sustava.

Literatura

[1] Pavlić Mile, Oblikovanje baze podataka, Odjel za informatiku sveučilišta u Rijeci, Rijeka, 2011.

[2] Clarion - Home, SoftVelocity, http://www.softvelocity.com/

Popis slika

Slika 1 Grafički oblik koncepata strukture EV metode [1] 6
Slika 2 DEV 9
Slika 3 Izrada tablice 11
Slika 4 Dodavanje atributa 12
Slika 5 Dodavanje ključeva 13
Slika 6 Dodavanje veza 14
Slika 7 DPP 15
Slika 8 Glavni izbornik – „Actions“ 16
Slika 9 Glavni izbornik – „Window“ 17
Slika 10 Uređivanje menija 18
Slika 11 Skočni prozor 18
Slika 12 Popis – „Actions“ 19
Slika 13 Popis – „Window“ 19
Slika 14 „ListBox format“ 20
Slika 15 Sortiranje po Tabovima 21
Slika 16 „Locator Behavior“ 21
Slika 17 Gumb za pretraživanje 22
Slika 18 Ažuriranje – „Window“ 22
Slika 19 „Lookup“ procedura 23
Slika 20 „Lookup Button“ 23
Slika 21 „Calendar Button“ 24
Slika 22 Dodavanje novog „Listbox“-a 24
Slika 23 „Range Limit Field“ 25
Slika 24 „Totaling“ 26
Slika 25 Računanje sveukupne cijene 26
Slika 26 IF petlja 27
Slika 27 Jednostavno izvješće 27
Slika 28 Složeno izvješće 28
Slika 29 Početni prozor 29
Slika 30 „Račun“ i „Stavka“ 30
Slika 31 „Nekretnina“ 30
Slika 32 „Ispis Agenata“ 31
Slika 33 „Ispis odabrane nekretnine“ 32
Slika 34 „Račun“ 33