

Agencija za nekretnine - poslovna aplikacija nad relacijskom bazom podataka (Oracle APEX)

Siljan, Vanessa

Undergraduate thesis / Završni rad

2022

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:195:050708>

Rights / Prava: [In copyright](#) / [Zaštićeno autorskim pravom.](#)

Download date / Datum preuzimanja: **2024-05-19**



Sveučilište u Rijeci
**Fakultet informatike
i digitalnih tehnologija**

Repository / Repozitorij:

[Repository of the University of Rijeka, Faculty of Informatics and Digital Technologies - INFORI Repository](#)



Sveučilište u Rijeci – Fakultet informatike i digitalnih tehnologija

Sveučilišni preddiplomski studij informatike

Vanessa Siljan

Agencija za nekretnine - poslovna
aplikacija nad relacijskom bazom
podataka (Oracle APEX)

Završni rad

Mentor: doc. dr. sc. Danijela Jakšić

Rijeka, 11. srpnja 2022.



Rijeka, 14.6.2022.

Zadatak za završni rad

Pristupnik: Vanessa Siljan

Naziv završnog rada: Agencija za nekretnine - poslovna aplikacija nad relacijskom bazom podataka (Oracle APEX)

Naziv završnog rada na eng. jeziku: Real estate agency - business application with relational database (Oracle APEX)

Sadržaj zadatka: Baza podataka (BP) predstavlja kolekciju podataka, ograničenja i operacija koji reprezentiraju neke aspekte realnoga svijeta. Dakle, BP je model neke aplikacijske domene. Cilj rada je izgraditi aplikaciju za potrebe procesa agencije za nekretnine. Aplikacija će biti izgrađena nad relacijskom BP pomoću alata Oracle APEX. Na početku će biti specificirani zahtjevi i potrebne značajke aplikacije. Sljedeći korak bit će dizajn baze podataka, odnosno izrada modela entiteta i veze. Transformacijom modela entiteta i veze konstruirat će se relacijski model. Na temelju relacijskog modela izgradit će se baza podataka. Nakon toga kreće izrada same aplikacije. Aplikacija će raditi nad relacijskom BP u navedenoj aplikacijskoj domeni (agencija za nekretnine).

Mentor

doc. dr. sc. Danijela Jakšić

Voditelj za završne radove

doc. dr. sc. Miran Pobar

Zadatak preuzet: 14.6.2022.

(potpis pristupnika)

Sažetak

Unutar ovog završnog rada detaljno je opisan postupak izrade poslovne aplikacije nad relacijskom bazom podataka koristeći *Oracle Application Express* (Oracle APEX) za agenciju za nekretnine. Cilj je prikazati način korištenja alata Oracle APEX kroz vlastito osmišljen primjer. Prije same izrade aplikacije izrađen je dijagram entiteti – veze koristeći alat Figma. Također, izrađen je odgovarajući relacijski mode koji prati dijagram entiteti – veze. Izrađen je mockup zbog pružanja mogućnosti uvida u detalje, boje, grafiku te dizajn aplikacije. Mockup je izrađen u alatu Figma. Unutar same aplikacije moguće je pregledavati, dodavati, ažurirati i brisati podatke unutar same baze podataka (CRUD).

Ključne riječi

Aplikacija, Oracle APEX, Dijagram, DEV, CRUD, Primarni ključ, Vanjski ključ, Agencija, Baza podataka, Relacija, Atribut, Entitet, Agregacija

Sadržaj

1. Uvod.....	5
2. Model entiteta - veze.....	7
3. Relacijski model.....	13
4. Prikaz mockup-a	14
5. Izrada aplikacije.....	18
5.1. Postavljanje radnog okruženja.....	18
5.2. Kreiranje aplikacije	18
5.3. Izrada baze podataka.....	19
5.4. Izrada stranica aplikacije i regija unutar stranica za prikaz podataka.....	22
5.5. Izrada stranica aplikacije i regija unutar stranica za prikaz, uređivanje i brisanje podataka.....	26
5.5.1. Form view	26
5.5.2. Report with form	30
5.5.3. Master Detail	31
5.6. Izrada navigacije.....	32
5.7. Autentikacija i autorizacija.....	33
6. Prikaz kreirane aplikacije	38
7. Zaključak	46
8. Popis slika.....	47
9. Literatura	49
10. Popis priloga	50

1. Uvod

Unutar ovog završnog rada biti će prikazan proces izrade poslovne aplikacije nad relacijskom bazom podataka za agenciju za nekretnine u alatu Oracle APEX. Poslovna aplikacija „Agencija s nekretninama“ nad relacijskom bazom podataka izrađena u Oracle APEX-u sadržavala bi nekretnine poput kuća, zemljišta i stanova. Cilj ove aplikacije je postići što veći broj zadovoljnih korisnika.

Postojale bi dvije vrste pogleda, za registrirane i neregistrirane korisnike. Neregistrirani korisnici bi mogli pregledavati i filtrirati kuće, zemljišta i stanove. Kuće bi se mogle filtrirati s obzirom na vrstu kuće (npr. prizemnica, dvokatnica, trokatnica,...), a zemljišta bi se mogla filtrirati s obzirom na vrstu zemljišta (npr. građevinsko ili poljoprivredno zemljište). Tijekom pregleda nudili bi se osnovni podaci o pojedinoj kući, zemljištu ili stanu, te agentima i klijentima koji su u bazi podataka. Registrirani korisnici, odnosno agenti, osim prethodno navedenih funkcionalnosti pregledavanja i filtriranja, mogu vršiti ubacivanje, brisanje i uređivanje podataka. Nadalje, agenti bi mogli voditi računa o dogovorenim upitima vezanih uz klijentov obilazak nekretnine. Na taj način se može vidjeti kada koji klijent ima zakazan termin za obilazak određene nekretnine. Primarni korisnici bili bi agenti, odnosno zaposlenici unutar određene agencije za nekretnine, te klijenti koji žele prodati ili kupiti određenu nekretninu.

Za izradu aplikacije korišten je Oracle APEX verzije 22.1.1. koji spada u kategoriju RAD alata (Rapid Application Development tools). 1991. godine James Martin službeno je predstavio metodologiju brzog razvoja aplikacije, iako je takva metodologija osmišljena još 1970.-ih godina. RAD alati koriste agilnu metodu koja omogućava brz i siguran način izrade aplikacije kroz česte iteracije i odobrenja uz stalne povratne informacije klijenta. Također, RAD alati poznati su po: samostalnom generiranju programskog koda, skraćenom vremenu razvoja i ubrzanoj isporuci, kraćem vremenu testiranja, boljem upravljanju rizicima, te poboljšanoj fleksibilnosti i prilagodljivosti. Neke od poznatijih tehnologija koje koriste RAD metodologiju jesu: Kissflow, Bizagi, Appian, Zoho Creator, APEX i mnoge druge.

Oracle Application Express (APEX) je platforma tvrtke Oracle Corporation koja se koristi za razvoj i implementaciju web aplikacija. Oracle APEX započinje sa svojim radom 2000.-e godine te je tada nosio ime FLOWS. Tijekom idućih godina promijenio je nekoliko naziva poput: „Oracle Platform“, „HTML DB“, „Project Marvel“ te 2006. godine dobiva današnji naziv „Application Express“. Navedeni alat prepoznatljiv je po niskoj stopi korištenja programskog koda što se naziva „low code“ pristup (Wikipedia, n.d.). Low code okruženja postaju vrlo popularan način za izradu aplikacija. Američka tvrtka Gartner, koja se bavi tehnoloških istraživanjem, predviđa da će do 2024. godine 65% aplikacija koristiti low-code okruženja tijekom njihovog razvoja (mendix, n.d.). Stoga, se koriste grafička sučelja sa jednostavnom logikom i „drag and drop“ elementima umjesto programiranja. APEX pruža integrirano razvojno okruženje temeljeno na webu koje koristi različite čarobnjake i uređivače svojstava za izradu aplikacija i njihovih stranica. Cilj ovog alata je pojednostaviti proces izrade web aplikacije koje, između ostalog, služi kao frontend poslužitelju baze podataka. Naime, integrirano razvojno okruženje (IDE) je vrlo intuitivno, što uvelike olakšava korištenje alata, bez obzira je li korisnik iskusan programer ili ne. Oracle APEX postoji na Oracle bazi podataka te svi meta podaci, kao i PL/SQL kod i JavaScript

API-ji su pohranjeni u samoj bazi podataka. Dostupan je kao Cloud servis, no postoji mogućnost preuzimanja programa te rada u desktop aplikaciji.

2. Model entiteti - veze

Za izradu modela entiteti-veze korištena je metodologija MIRIS (skraćenica za Metodologija za Razvoj Informacijskog Sustava). Metoda entiteti – veze (eng. *Entity – relationship model*) je jedan od prvih semantički bogatijih metoda za modeliranje podataka, ono je grafički prikaz međusobno povezanih grupa podataka promatranoga sustava¹. Za izradu nama potrebnog modela EV korišteni su koncepti poput jakog tipa entiteta, agregacije, tipa veze te tipa atributa. Dijagram strukture modela naziva se dijagram entiteta i veza, skraćeno DEV.

Entitet (eng. *entity*) je pojam u poslovanju o kome se zahtijeva čuvanje nekih podataka. Tip entiteta (eng. *entity type*) je skup sličnih pojedinačnih entiteta dobiveno procesom klasifikacijske apstrakcije². To je skup entiteta istog tipa. Klasifikacija je apstrakcija u kojoj se slični entiteti proglašavaju zajedničkim skupom nazvanim Tip entiteta.

Jaki tip entiteta (ili samo tip entiteta) je tip entiteta koji ima vlastiti primarni ključ i nije ovisan o drugi tipovima entiteta u modelu podataka. Tip entiteta označava se simbolom pravokutnika. Slabi tip entiteta (eng. *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. Specijalni tip veze je tip veze koji povezuje dva tipa entiteta među čijim entitetima postoji ovisnost. Moguće su različite vrste ovisnosti, a prema njoj definira se vrsta specijalnog tipa veze. Grafički prikaz specijalnog tipa veze prikazuje se romбом ili linijom s dodatkom strelice na liniji od jakog k slabom tipu entiteta.

Agregacija ili mješoviti tip entiteta ili agregirani tip entiteta je apstrakcija u kojoj se tip veze između dvaju ili više tipova entiteta tretira kao novi tip entiteta. Agregacija se grafički prikazuje simbolom pravokutnika unutar kojeg se nalazi simbol romba.

Veza ili asocijacija (eng. *relationship*) je koncept koji predstavlja neku interakciju među entitetima u sustavu, odnosno predstavlja znanje o njihovoj povezanosti. Grafički prikaz je linija ili romb koji sadrže naziv veze.

Atribut entiteta (eng. *attribute*) je neko svojstvo entiteta. Atribut pridružuje entitete vrijednostima i obrnuto. Na primjer: naziv agencije za nekretnine, ime agenta, prezime agenta... Grafički se predstavlja imenovanom linijom između pravokutnika i kruga, odnosno između naziva entiteta i tipa vrijednosti. Vrijednost (eng. *value*) je podataka, element ili elementarni podatak. Tip podataka (eng. *data type*) je oznaka vrste podataka u tipu vrijednosti³. Mogući su slijedeći tipovi podataka: slova, znakovi, datumi, brojevi, alfanumerički (kombinacija brojeva i znakova). Atribut može imati neodređenu vrijednost nekog atributa. Takvu vrijednost nazivamo null vrijednost (eng. *null value*).

¹ Metoda entiteti – veze prvi put je objavljena u Chenovu članku 1976. godine






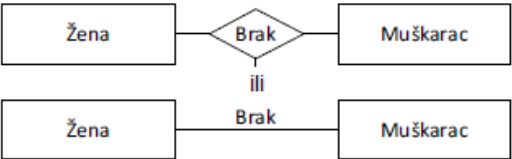

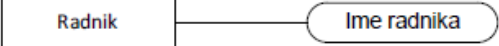
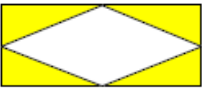

² Klasifikacija označava razvrstavanje na temelju nekog sustava; podjela predmeta ili pojmova na razrede, skupine, rodove, vrste, tipove, prema određenim načelima koja ovise o vrsti građe i svrsi klasifikacije

³ Tip vrijednosti predstavlja skup pojedinačnih vrijednosti i označava se simbolom ovalnog oblika

Za svaki tip entiteta atribut tog tipa entiteta može pridružiti jednom pojavljivanju entiteta vrijednosti iz tipa vrijednosti u sljedećoj količini:

- 0 – nijedan
- 1 – jedan
- Br – cijeli prirodan broj
- M – mnogo, neodređeni cijeli prirodan broj

Brojnost preslikavanja atributa su dva uređena para koja sadrže gornji i donju granicu brojnosti. Uređen par predstavlja zapis dvaju elemenata u uglatim zagradama odvojene zarezom, na primjer (1,M).

KONCEPT	SIMBOL	PRIMJER
TIP ENTITETA		
SLAB TIP ENTITETA		
TIP VEZE		
ATRIBUT		
AGREGACIJA		

Slika 1: Prikaz grafičkih oblika konceptata strukture EV metode, preuzeto iz (Pavlić, 2011.)

Jaki tipovi entiteta jesu slijedeći: ŽUPANIJA, MJESTO, AGENCIJA ZA NEKRETNINE, AGENTI, VRSTA KUĆE, KUĆA, ZEMLJIŠTE, VRSTA ZEMLJIŠTA, STAN, OBILAZAK NEKRETNINE, KLIJENT. JE_ZAPOSLEN predstavlja agregaciju koja se sastoji od AGENCIJE ZA NEKRETNINE i AGENTI.

Jaki tip entiteta ŽUPANIJA sadrži attribute *ID_županije* i *Naziv_županije*. *ID_županije* predstavlja primarni ključ. Ovaj tip entiteta predstavlja sve županije koja imaju mjesta u kojoj se nalaze nekretnine, agencija ili prebivališta klijenta. ŽUPANIJA je povezana s MJESTOM te jedna županija može imati najmanje jedno mjesto, a najviše mnogo mjesta. Jedno mjesto može se nalaziti u točno jednoj županiji.

Jaki tip entiteta MJESTO sadrži attribute *Pošt_broj* i *Naziv_mjesta*. *Pošt_broj* predstavlja poštanski broj mjesta te zbog njegove jedinstvenosti i neredundantnosti je

izabran za primarni ključ. Ovaj tip entiteta predstavlja sva mjesta u kojima se nalaze nekretnine, agencija ili klijentovo prebivalište. MJESTO je uz ŽUPANIJU povezano s AGENCIJOM ZA NEKRETNINE, ZEMLJIŠTEM, STANOM, KUĆOM, KLIJENTOM. Sa strane mjesta svugdje se nalazi uređeni par (0,M) zato što se u jednom mjestu može nalaziti niti jedna ili mnogo kuća, stanova ili zemljišta. Također, u nekom mjestu se može nalaziti niti jedna ili mnogo agencija za nekretnine. U određenom mjestu može se nalaziti niti jedno ili mnogo prebivališta klijenta.

Jaki tip AGENCIJA ZA NEKRETNINE sadrži attribute poput *ID_agencije*, *Naziv_agencije*, *Adresa_agencije*, *Broj_tel_agencije*. *ID_agencije* odabran je za primarni ključ. Ovaj tip entiteta predstavlja agenciju za nekretnine koja oglašava kuće, stanove i zemljišta. AGENCIJA ZA NEKRETNINE se mora nalaziti točno u jednom mjestu zato veza koja povezuje MJESTO i AGENCIJU ZA NEKRETNINE sa strane agencije za nekretnine ima brojnost (1,1) jer jedna agencija se najmanje može nalaziti u jednom mjestu te jedna agencija za nekretnine se najviše smije nalaziti u jednom mjestu. AGENCIJA ZA NEKRETNINE je povezana sa KUĆOM, STANOM i ZEMLJIŠTEM te sa strane agencije za nekretnine brojnost veze je (0,M) iz razloga što jedna agencija za nekretnine može posjedovati niti jednu ili mnogo kuća, zemljišta ili stanova. Također, AGENCIJA ZA NEKRETNINE sudjeluje u agregaciji JE_ZAPOSLEN te je to specijalni tip veze. Sa strane jakog tipa entiteta veza ima brojnost (0,M) zato što agencija za nekretnine može imati zaposlenog niti jednog agenta ili mnogo njih.

Agregacija JE_ZAPOSLEN je nastala iz razloga što jedan agent može biti zaposlen u više agencija za nekretnine te jedna agencija za nekretnine može imati više zaposlenih agenata. Stoga, u agregaciji sudjeluju dva jaka tipa entiteta: AGENCIJA ZA NEKRETNINE i AGENTI. JE_ZAPOSELEN sadrži attribute poput: *Datum_zaposlenja*, *Osnovna_placa*, *Provizija*. S obzirom da je to agregacija, primarni ključ se sastoji od primarnih ključeva tipova entiteta koji sudjeluju u agregaciji, odnosno *ID_agencije* te *ID_agenta*. Ovaj složeni tip entiteta sadrži podatke poput datuma zaposlenja, provizije, osnovne plaće određenog agenta. Sa strane agregacije brojnosti obje veze iznose (1,1).

Jaki tip entiteta AGENTI sadrži slijedeće attribute: *ID_agenta*, *Ime_agenta*, *Prezime_agenta*, *Br_tel_agenta*. *ID_agenta* odabran je za primarni ključ. Ovaj tip entiteta predstavlja sve agente zaposlene u agenciji za nekretnine. Kao što je prethodno navedeno, AGENTI je povezan s agregacijom JE_ZAPOSLEN te je sa strane AGENTI brojnost (0,M) jer jedan agent može biti zaposlen u niti jednoj agenciji za nekretnine ili u mnogo njih. Također, AGENTI je povezan s OBILAZAK NEKRETNINE. Brojnost veze sa strane AGENTI iznosi (0,M) zato što jedan agent može biti zadužen za niti jedan obilazak nekretnine ili za mnogo obilaska.

Jaki tip entiteta STAN sadrži attribute poput: *ID_stana*, *Adresa_stana*, *Broj_soba_stana*, *Površina_stana*, *Parkirno_mjesto*, *Pogled/orijentacija*, *Kat*, *Drvarnica*, *Vrsta_grijanja*, *Godina_izgradnje_zgrade*, *Opis_stana*, *Cijena_stana*. *ID_stana* odabran je za primarni ključ. Ovaj tip entiteta predstavlja sve stanove koji se oglašavaju u agenciji za nekretnine. STAN je povezan s prethodno tipom entiteta MJESTO i AGENCIJA ZA NEKRETNINE. Brojnost veze s AGENCIJOM ZA NEKRETNINE sa strane STANA iznose (1,1) zato što se jedan stan može oglašavati u najmanje jednoj agenciji za nekretnine

i u najviše jednoj agenciji za nekretnine. Što se tiče povezanosti s tipom entiteta MJESTO, jedan stan se može nalaziti u najmanje jednom te u najviše jednom mjestu. Iz navedenog razloga veza ima brojnost (1,1) sa strane tipa entiteta STAN. STAN sudjeluje u vezi s tipom entiteta OBILAZAK NEKRETNINE te je brojnost veze sa strane stana (0,M) zato što jedan stan može nitko ne doći obići ili može mnogo njih doći obići.

Jaki tip entiteta VRSTA ZEMLJIŠTA sadrži attribute *ID_vrste_zemljišta* i *Naziv_vrste_zemljišta*. *ID_vrste_zemljišta* predstavlja primarni ključ. Ovaj tip entiteta predstavlja sve vrste zemljišta koja postoje unutar agencije zbog lakšeg filtriranja zemljišta unutar baze podataka. VRSTA ZEMLJIŠTA sudjeluje u relaciji s jakim tipom entiteta ZEMLJIŠTE te brojnost veze sa strane VRSTE ZEMLJIŠTA iznosi (0,M) zato što može postojati vrsta zemljišta koja ne pripada niti jednom zemljištu koje se trenutno oglašava u agenciji ili može postojati vrsta zemljišta koja pripada mnogim zemljištima.

Jaki tip entiteta ZEMLJIŠTE sadrži attribute *ID_zemljišta*, *Adresa_zemljišta*, *Površina_zemljišta*, *Opis_zemljišta*, *Cijena_zemljišta*. *ID_zemljišta* odabran je za primarni ključ. Ovaj tip entiteta sadrži sva zemljišta koja se oglašavaju unutar agencije za nekretnine. Tip entiteta ZEMLJIŠTE sudjeluje u vezi s jakim tipovima entiteta poput OBILAZAK NEKRETNINE, VRSTA ZEMLJIŠTA, MJESTO, AGENCIJA ZA NEKRETNINE. Što se tiče veza s MJESTO, AGENCIJA ZA NEKRETNINE i VRSTA ZEMLJIŠTA, brojnost sa strane zemljišta iznosi (1,1) zato što se jedno zemljište nalazi u točno jednom mjestu, odnosno u najmanje jednom i najviše jednom mjestu. Jedno zemljište može biti samo jedne vrste zemljišta, pa zemljište može biti najmanje i najviše jedne vrste. Također, jedno zemljište se može oglašavati u najmanje jednoj i najviše jednoj agenciji za nekretnine.

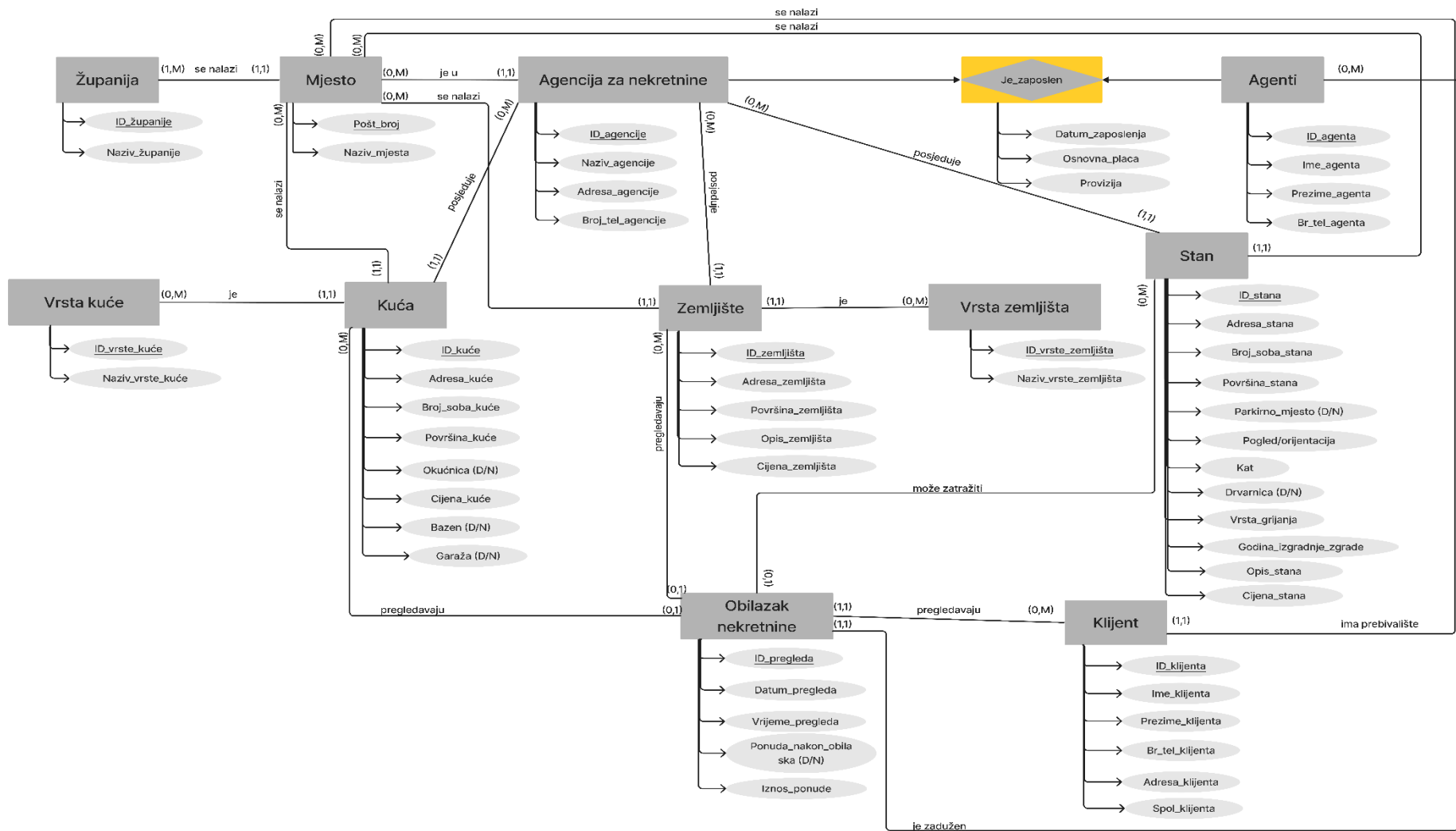
Jaki tip entiteta KUĆA sadrži attribute kao što su: *ID_kuće*, *Adresa_kuće*, *Broj_soba_kuće*, *Površina_kuće*, *Okućnica*, *Cijena_kuće*, *Bazen*, *Garaža*. *ID_kuće* je odabran za primarni ključ. Ovaj tip entiteta sadrži sve kuće koje se oglašavaju unutar agencije za nekretnine. Jaki tip entiteta KUĆA povezan je s tipom entiteta OBILAZAK NEKRETNINE, VRSTA KUĆE, AGENCIJA ZA NEKRETNINE, MJESTO. Veze u kojima sudjeluju MJESTO, AGENCIJA ZA NEKRETNINE i VRSTA KUĆE sa strane KUĆE imaju brojnosti (1,1) iz razloga što jedna kuća se može nalaziti u najmanje jednom i najviše jednom mjestu. Također, jedna kuća se može oglašavati u najmanje jednoj i najviše jednoj agenciji za nekretnine, te jedna kuća može biti točno jedne vrste, odnosno jedna kuća biti najmanje jedne i najviše jedne vrste kuće.

Jaki tip entiteta VRSTA KUĆE sadrži attribute *ID_vrste_kuće* i *Naziv_vrste_kuće*. *ID_vrste_kuće* odabran je za primarni ključ. Ovaj tip entiteta predstavlja sve vrste kuća koje postoje unutar agencije zbog lakšeg filtriranja kuća unutar baze podataka. Ovaj tip entiteta povezan je s prethodno spomenutim tipom entiteta KUĆA te brojnost veze sa strane VRSTA KUĆE iznosi (0,M) zato što može postojati vrsta kuće koja ne pripada niti jednoj kući koja se trenutno oglašava u agenciji ili može postojati vrsta kuće koja pripada mnogim kućama.

Jaki tip entiteta OBILAZAK NEKRETNINE ima attribute: *ID_pregleda*, *Datum_pregleda*, *Vrijeme_pregleda*, *Ponuda_nakon_obilaska*, *Iznos_ponude*. *ID_pregleda* odabran je za primarni ključ. Ovaj tip entiteta predstavlja sve obilaskе nekretnine u kojima klijent zajedno s agentom obilazi određene nekretnine poput stanova, kuća ili zemljišta. OBILAZAK NEKRETNINE sudjeluje u vezi s tipovima entiteta KLIJENT, STAN,

ZEMLJIŠTE, KUĆA te AGENT. Što se tiče veze s tipovima entiteta KLIJENT i AGENT brojnost veze sa strane OBILAZAK NEKRETNINE je (1,1) zato što u jednom određenom terminu najmanje jedan i najviše jedan klijent može ići u obilazak nekretnine. Također, jedan obilazak nekretnine može voditi najmanje jedan i najviše jedan agent. Nadalje, po pitanju veza s tipovima entiteta STAN, ZEMLJIŠTE i KUĆA brojnosti sa strane OBILAZAK NEKRETNINE iznose (0,1) zato što u jednom obilasku nekretnine može sudjelovati niti jedna ili najviše jedna nekretnina kao što je kuća, stan ili zemljište.

Jaki tip entiteta KLIJENT sadrži attribute: *ID_klijenta*, *Ime_klijenta*, *Prezime_klijenta*, *Br_tel_klijenta*, *Adresa_klijenta*, *Spol_klijenta*. *ID_klijenta* odabran je za primarni ključ. Ovaj tip entiteta predstavlja sve klijente koji se nalazi u bazi podataka agencije za nekretnine. KLIJENT sudjeluje u vezi s jakim tipovima entiteta OBILAZAK NEKRETNINE i MJESTO. Što se tiče brojnosti veze s tipom MJESTO, ona iznosi (1,1) sa strane KLIJENTA zato što jedan klijent može imati trenutno prebivalište u najmanje jednom i najviše jednom mjestu. Brojnost veze u kojoj sudjeluje OBILAZAK NEKRETNINE sa strane KLIJENTA iznosi (0,M) zato što jedan klijent može pregledati najmanje niti jednu ili najviše mnogo nekretnina.



Slika 2: Prikaz modela entiteti-veze

3. Relacijski model

Primarni ključ (glavni ključ, engl. *primary key*) jedan je od ključeva tipa entiteta koji je odabran zbog nekoga značajnijeg razloga za zamjenika entiteta toga tipa (za ključ). Jedan od razloga je njegovo često korištenje. Ključ agregacije je skup ključnih atributa komponenti agregacije. Agregacija dobiva ključ od komponenti agregacije. Brojnost ključa tipa agregacije uvijek je (1,1):(1,1) inače to ne bi bio ključ. Da bi neki atribut bio primarni ključ, on mora zadovoljiti dva uvjeta: uvjet jedinstvenosti i uvjet neredundantnosti. Uvjet jedinstvenosti osigurava da vrijednost ključa svake n-torke relacije jedinstveno određuje n-torku, odnosno ne smiju postojati dva retka unutar iste tablice koja imaju sve iste vrijednosti svih atributa koji čine ključ. Uvjet neredundantnosti osigurava da ne postoji niti jedan atribut kao dio ključa koji se smije izostaviti iz ključa, a da se pritom uvjet jedinstvenosti ne gubi. Ako neki skup atributa u promatranoj relaciji nije ključ, ali je ključ u nekoj drugoj relaciji u modelu, onda se on naziva vanjski ključ (engl. *foreign key*). Na temelju prethodno izrađenog modela EV i pravila za prevođenje u relacijski model, dobije se relacijski model prikazan u nastavku.

ŽUPANIJA (ID_županije, Naziv_županije)

MJESTO (Pošt_broj, Naziv mjesta, ID_županije)

AGENCIJA_ZA_NEKRETNINE (ID_agencije, Naziv_agencije, Adresa_agencije, Broj_tel_agencije, Pošt_broj)

AGENTI (ID_agenta, Ime_agenta, Prezime_agenta, Br_tel_agenta)

JE_ZAPOSLEN (ID_agenta, ID_agencije, Datum_zaposlenja, Osnovna_placa, Provizija)

STAN (ID_stana, Adresa_stana, Broj_soba_stana, Površina_stana, Parkirno_mjesto, Pogled/orijentacija, Kat, Drvarnica, Vrsta_grijanja, Godina_izgradnje_zgrade, Opis_stana, Cijena_stana, Pošt_broj, ID_agencije)

VRSTA_ZEMLJIŠTA (ID_vrste_zemljišta, Naziv_vrste_zemljišta)

ZEMLJIŠTE (ID_zemljišta, Adresa_zemljišta, Površina_zemljišta, Opis_zemljišta, Cijena_zemljišta, ID_vrste_zemljišta, ID_agencije, Pošt_broj)

KUĆA (ID_kuće, Adresa_kuće, Broj_soba_kuće, Površina_kuće, Okućnica, Cijena_kuće, Bazen, Garaža, ID_agencije, ID_vrste_kuće, Pošt_broj)

VRSTA_KUĆE (ID_vrste_kuće, Naziv_vrste_kuće)

KLIJENT (ID_klijenta, Ime_klijenta, Prezime_klijenta, Br_tel_klijenta, Adresa_klijenta, Spol_klijenta, Pošt_broj)

OBILAZAK_NEKRETNINE (ID_pregleda, Datum_pregleda, Vrijeme_pregleda, Ponuda_nakon_obilaska, Iznos_ponude, ID_klijenta, ID_agenta, ID_zemljišta, ID_kuće, ID_stana)

4. Prikaz mockup-a

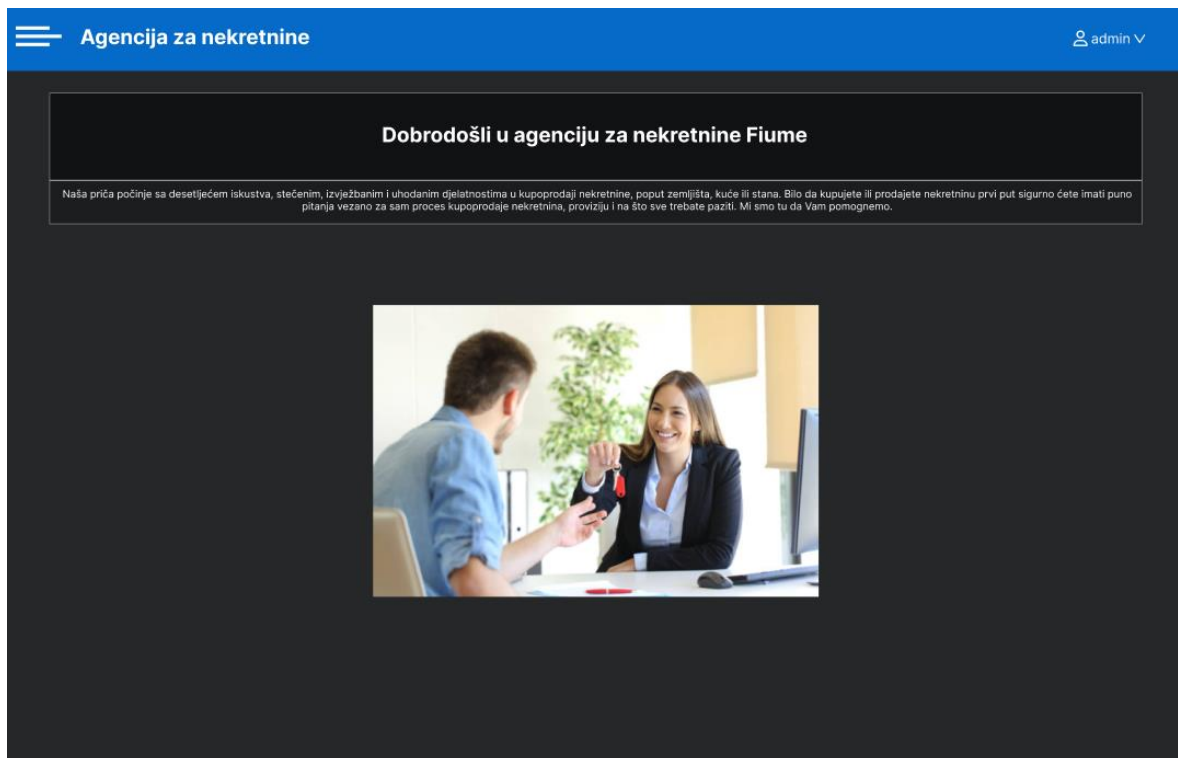
Oduvijek se težilo da se klijentima prezentira što jasnija slika o tome kako će njihova aplikacija izgledati. Takav pristup pomaže klijentima da budu odlučniji prilikom izbora grafičkog rješenja koje im više odgovara. Shvaćanje grafičkog dizajna važno je kod svih proizvođača koji su u interakciji s čovjekom. Grafički dizajn je grana primijenjene umjetnosti koja koristi grafičke elemente kao što su tipografija, boje, geometrijski oblici, ilustracije i fotografije kako bi se vizualnom komunikacijom, brzo i efikasno prenijela neka poruka. Ključno u dizajnu nije samo vidljivi segment, nego i sama interakcija korisnika s proizvodom pa je tako uz dizajn korisničkog sučelja još važniju ulogu dobio dizajn korisničkog iskustva. Za dobro korisničko iskustvo moramo imati i dobar dizajn korisničkog sučelja. Što se prije krene razmišljati o korisničkom iskustvu, sučelju i interakciji manji je rizik da sustav neće odgovarati korisniku. Korisničko sučelje (eng. *user interface* (UI)) i korisničko iskustvo (eng. *user experience* (UX)) usko su povezani.

Korisnici preko korisničkog sučelja mogu čitati informacije, birati akcije ili promatrati fotografije. Kod korisničkog sučelja treba voditi računa o: rasporedu sadržaja, isticanju elemenata, standardizaciji, brzini učitavanja i interaktivnosti. Što se tiče rasporeda sadržaja, korisnicima je lakše skenirati pogledom uredan prostor jer na taj način dobivaju na vremenu. Također, svi ekrani bi trebali imati sličan raspored sadržaja i elemenata. Na taj način korisnik zna što može očekivati. Isticanje elemenata postiže se korištenjem kontrasta u tipografiji, povećanju (ili podebljanju) teksta, promjeni boje nekih elemenata. Po pitanju standardizacije, potrebno je definirati font, gumbe, boje i stil prikaza podataka. Zbog bržeg učitavanja preporuča se optimiziranje veličine slike. Atraktivnost predstavlja kombinaciju grafičkih elemenata kako bi se sadržaj prikazao na zanimljiv i oku ugodan način. Upravo atraktivnost, odnosno atraktivno korisničko sučelje zadržava korisnika na stranici ili u aplikaciji.

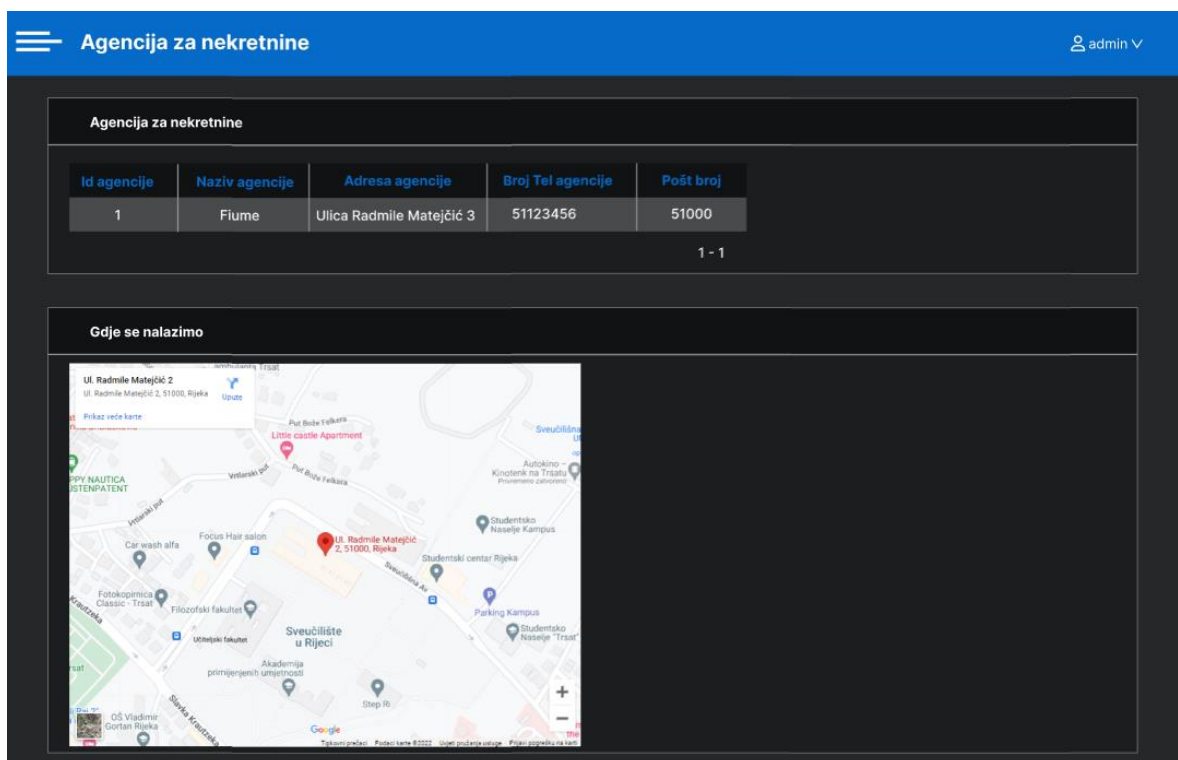
Cilj je kontinuirano provoditi testiranje u svakoj iteraciji dizajna. To znači provoditi testiranje koje kreće s izradom wireframea, te zatim preko mockup-a doći sve do prototipa. Wireframe daje grubu sliku web stranice ili aplikacije koristeći osnovne linije i oblike. Prikazuju se ekrani sa osnovnim elementima koji će se na njima naći. Također, ne vodi se briga o estetici, već se ugrubo žele odrediti potrebni elementi na ekranu. Wireframe se brzo izrađuje, te njegova izrada nije skupa, a ujedno daje dobru podlogu za komunikaciju među članovima tima. Korisnik treba vidjeti wireframe te dati povratnu informaciju dizajneru. Može biti izrađen na papiru ili pomoću nekakvog programa za crtanje poput Figma ili Adobe XD.

Mockup je nacrt koji prikazuje više vizualnih detalja poput boja, tipografije i tome slično. Mockup nastaje na način da se na wireframe dodaju vizualni elementi zanimljivi za korisnika te se dobije bolja slika buduće aplikacije. Uključuje boje, grafiku i tipografiju, prikazuje gumbove i stvarne tekstove, raspored elemenata je uređen i među elementima je definiran razmak, a definirana je i navigacija. Mockup prikazuje bogatstvo detalja projekta o kojima tim može lakše raspravljati. Svi tako bolje razumiju buduću proizvod i saznaju više o njemu. Na taj način smo korak bliže ka konačnom proizvodu i dobar je model na kojemu se rano mogu naći greške i može ih se izbjeći u sljedećoj iteraciji. Razrađuje se raspored stvari na zaslonu, kamo vodi određeni gumb, raspored elemenata na ekranu i slično. Tako se

uočavaju i rješavaju problemi navigacije i rasporeda funkcionalnosti po ekranima. Mockup se izrađuje pomoću softverskog alata, a među popularnijima jesu upravo Figma i Adobe XD. U nastavku je prikazan mockup za aplikaciju Agencija za nekretnine.



Slika 3: Prikaz mockup-a za početnu stranicu



Slika 4: Prikaz mockup-a za stranicu O agenciji

Agencija za nekretnine

admin

Popis županija

Zupanija

Istarska

Pretraži

▼ Popis mjesta

Pošt broj	Naziv mjesta
52100	Pula

1 - 1

Slika 5: Prikaz mockup-a za stranicu Lokacija

Agencija za nekretnine

admin

Interactive grid - obilazak nekretnine

Create

Search: All Text Columns

Go

Actions

Reset

	Datum pregleda	Vrijeme pregleda	Ponuda nakon obilaska	Iznos ponude	Prezime klijenta	Prezime agenta	Adresa zemljišta	Adresa kuće	Adresa stana
	7/9/2022	7/9/2022	Ne		Ivić	Josic		Šišanska ulica 11	
	7/7/2022	7/7/2022	Da	150	Horvat	Franić			Kaštanjer 54
	7/9/2022	7/9/2022			Ivić	Franić			Kaštanjer 54
	7/14/2022	7/14/2022			Lukić	Anić		Vahvidalska ulica 50	
	7/19/2022	7/19/2022	Da	10000	Nikolić	Franić		Šišanska ulica 11	

1 rows selected

Total 5

Slika 6: Prikaz mockup-a za stranicu Obilasci

Agencija za nekretnine admin

Master detail - Agenti Reset Create

Search...

Agenti Edit

Ime agenta	Ana
Prezime agenta	Anić
Broj telefona agenta	98123456

Je zaposlen +

	Id agencije	Datum zaposlenja	Osnovna plaća	Provizija
	1	2/13/2020	9000	20

1 - 1

Slika 7: Prikaz mockup-a za stranicu Agenti

Agencija za nekretnine admin

Pregled korisnika

Username	IsAdministrator
ADMIN	Y
GUEST	N
VANESSA	Y

1 - 3

Umetanje korisnika

Korisničko ime

Lozinka

Potvrda lozinke

☐ Admin

Stvori korisnika

Slika 8: Prikaz mockup-a za stranicu Korisnici

5. Izrada aplikacije

5.1. Postavljanje radnog okruženja

Izrada aplikacije u Oracle APEX-u započinje kreiranjem radnog okruženja. Potrebno je popuniti podatke poput: imena, prezimena, elektroničke adrese, naziva radnog okruženja i države. Nakon što je potvrđen upit za stvaranjem radnog okruženja, bilo je potrebno na prethodno unesenoj elektroničkoj adresi potvrditi zahtjev. Nakon potvrđenog zahtjeva, radno okruženje je uspješno kreirano.

Request a Workspace

●

●

●

●

●

Identification

First Name

Last Name

Email

A link to activate your workspace will be sent to this email address.

Workspace

This workspace name uniquely identifies your development environment and is used when signing in.

Country

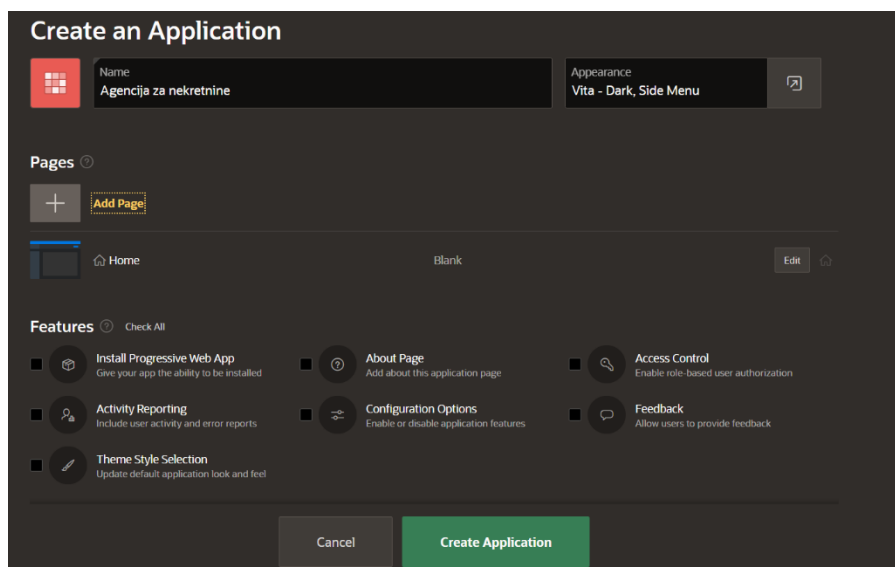
Usage ☒ Personal ☐ Organization

[Oracle Privacy Policy](#)

Slika 9: Kreiranje radnog okruženja

5.2. Kreiranje aplikacije

Nakon što se prijavimo u radno okruženje pomoću korisničkog imena i lozinke, potrebno je kreirati aplikaciju. Odabirom opcije *Create*, zatim *New Application* te je za dizajn odabrana tema *Vita-dark* i *Side-menu* za navigaciju. Unutar polja za naziv aplikacije, uneseno je „Agencija za nekretnine“.



Slika 10: Kreiranje aplikacije

5.3. Izrada baze podataka

Idući korak bio je transformiranje modela podataka u odgovarajuće SQL izraze za stvaranje tablica unutar baze podataka. Radi složenosti i količine samih upita, unutar SQL Workshop-a kreirane su SQL skripte koje sadrže upite za kreiranje tablica sa pripadajućim stupcima, unošenje podataka u svaku tablicu te postavljanje odnosa između tablica uz pomoć ograničenja (primarnih i vanjskih ključeva).

```
CREATE TABLE "MJESTO"
(
  "POŠT_BROJ" NUMBER(5,0) NOT NULL ENABLE,
  "NAZIV_MJESTA" VARCHAR2(50) NOT NULL ENABLE,
  "ID_ŽUPANIJE" NUMBER(5,0) NOT NULL ENABLE,
  CONSTRAINT "MJESTO1_PK" PRIMARY KEY ("POŠT_BROJ")
  USING INDEX ENABLE
)
/
ALTER TABLE "MJESTO" ADD CONSTRAINT "MJESTO_FK" FOREIGN KEY ("ID_ŽUPANIJE")
REFERENCES "ŽUPANIJA" ("ID_ŽUPANIJE") ON DELETE CASCADE ENABLE
/
```

Slika 11: Prikaz SQL upita za kreiranje tablice MJESTO

Također, tablice i njezini pripadajući stupci mogu se kreirati pomoću grafičkog sučelja na način da se odabere opcija *Create table* te se unese naziv tablice, nazivi stupaca, odgovarajući tip stupca te smije li vrijednost unutar stupca biti null ili ne smije. *Precision* se odnosi na duljinu, odnosno broj znamenki unutar varijable tipa NUMBER. *Scale* se kod tipa NUMBER odnosi na broji decimalnih mjesta, dok se kod tipa VARCHAR2 odnosi na duljinu znakovnog niza. Slijedeća fotografija prikazuje kreiranje tablice MJESTO koja sadrži dva stupca – *Pošt_broj* i *Naziv_mjesta*. *Pošt_broj* je tipa NUMBER te se njegova vrijednost sastoji do pet znamenki i ne smije poprimiti null vrijednost. *Naziv_mjesta* je tipa VARCHAR2 te smije primiti vrijednost koja se sastoji do pedeset znakova i vrijednost ne smije biti null. Nadalje, tablica JE_ZAPOSLEN predstavlja agregaciju između tipa entiteta AGENCIJA_ZA_NEKRETNINE i AGENTI. Iz tog razloga mora poprimiti njihove primarne ključeve te sadržavati tri vlastita atributa *Datum_zaposlenja*, *Osnovna_placa*, *Provizija*.

Create Table

Columns

Table Name: MJESTO

☐ Preserve Case

Column Name	Type	Precision	Scale	Not Null	Identity	Move
Pošt_broj	NUMBER	5	0	<input checked="" type="checkbox"/>	- None -	^ v
Naziv_mjesta	VARCHAR2		50	<input checked="" type="checkbox"/>		^ v
	- Select Datatype -					^ v
	- Select Datatype -					^ v
	- Select Datatype -					^ v
	- Select Datatype -					^ v
	- Select Datatype -					^ v
	- Select Datatype -					^ v

Add Column

Cancel

Next >

Slika 12: Kreiranje tablice MJESTO

Create Table

Columns

Table Name: JE_ZAPOSLEN

☐ Preserve Case

Column Name	Type	Precision	Scale	Not Null	Identity	Move
ID_agenta	NUMBER	5	0	<input checked="" type="checkbox"/>	- None -	^ v
ID_agencije	NUMBER	5	0	<input checked="" type="checkbox"/>	- None -	^ v
Datum_zaposlenja	DATE			<input type="checkbox"/>		^ v
Osnovna_placa	NUMBER	10	2	<input type="checkbox"/>	- None -	^ v
Provizija	NUMBER	10	2	<input type="checkbox"/>	- None -	^ v
	- Select Datatype -					^ v
	- Select Datatype -					^ v
	- Select Datatype -					^ v

Add Column

Cancel

Next >

Slika 13: Kreiranje tablice JE_ZAPOSLEN

Slijedeći korak pri kreiranju tablice odnosi se na primarne ključeve (engl. *Primary key*). Primarni ključ ima jedinstvenu vrijednost. Ukoliko tablica nije složena, odnosno ne predstavlja agregaciju ili slabi tip, onda se odabere opcije *Not populated* te se odabire onaj stupac koji ima jedinstvene vrijednosti. U suprotnom, se odabere opcije *Populated by Identity column*, kao što je napravljeno kod tablice JE_ZAPOSLEN te se pod opcijom

Primary Key odabere šifra jakog tipa koji sudjeluje u agregaciji, a kod *Composite Primary Key* primarni ključ odgovarajućeg drugog jakog tipa entiteta koji sudjeluje u agregaciji.

Create Table

Primary Key

Table name: JE_ZAPOSLEN

Primary Key:

- ☐ No Primary Key
- ☐ Populated from a new sequence
- ☐ Populated from an existing sequence
- ☐ Not populated
- ☒ Populated by Identity column

Primary Key Constraint Name: JE_ZAPOSLEN_PK

Primary Key: ID_AGENTA(NUMBER)

Composite Primary Key: ID_AGENCIJE(NUMBER)

Primary Key
A primary key allows each row in a table to be uniquely identified.
If you select to populate your primary key from a new sequence, you will be prompted to enter the new sequence's name. If you select to populate your primary key from an existing sequence, you will be prompted to select the sequence. Both these methods result in the generation of a trigger against your table. You can also select to not populate your primary. This is the only method that allows you to define a composite primary key made up of more than two columns.

< Cancel Next >

Slika 14: Kreiranje tablice - primarni ključ agregacije

Idući korak navodi na stvaranje vanjskih ključeva (engl. *Foreign key*). Kreiramo ih s obzirom na brojnosti veza između dva entiteta. Također, vanjski ključevi su iznimno dodani kod tablice JE_ZAPOSLEN koja predstavlja agregaciju. Naime, prema pravilima prevođenja agregacija JE_ZAPOSLEN trebala bi imati složeni primarni ključ koji se sastoji od *ID_agenta*, iz tablice AGENTI i *ID_agencije* iz tablice AGENCIJA_ZA_NEKRETNINE. Stoga, je prilikom postavljanja primarnog ključa u APEX-u za navedeni primjer bilo potrebno prvo postaviti vanjski ključ koji se odnosi na stupce iz drugih tablica (*ID_agenta* i *ID_agencije*), a potom postaviti složene primarne ključeve kako bi se podaci povlačili iz jedne tablice u drugu. Tijekom postavljanja vanjskih ključeva odabrana je opcija CASCADE.

Create Table

Foreign Key

Foreign Key	Columns	Referenced Table	Referenced Columns	Action
JE_ZAPOSLLEN_AGENTI_FK	ID_AGENTA	AGENTI	ID_AGENTA	cascade

Add Foreign Key

Name: **JE_ZAPOSLLEN_AGENCIJA_FK**

Select Key Column(s): ID_AGENTA, DATUM_ZAPOSLLENJA, OSNOVNA_PLACA, PROVIZIJA

References Table: **AGENCIJA_ZA_NEKRETNINE**

Referenced Column(s): NAZIV_AGENCIJE, ADRESA_AGENCIJE, BROJ_TEL_AGENCIJE, POŠT_BROJ

Options: ☐ Disallow Delete, ☒ Cascade Delete, ☐ Set Null on Delete

Add

Slika 15: Kreiranje tablice - vanjski ključ agregacije

Na kartici *Constraint* moguće je vidjeti ograničenja koja su postavljena za pojedine attribute, primjerice primarni ključ koji pruža jedinstvenu vrijednost atributa unutar jednog stupca te prilikom pokušaja unosa iste vrijednosti pojaviti će se greška. Također, korišteno je ograničenje Check koje onemogućuje null vrijednosti te su prikazana dva vanjska ključa.

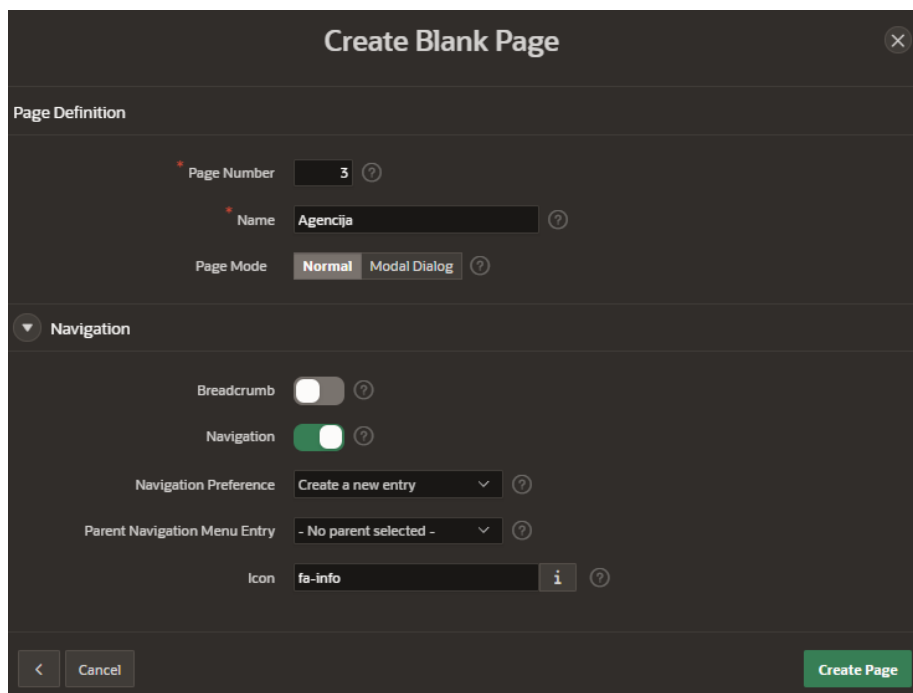
JE_ZAPOSLLEN										
Table	Data	Indexes	Model	Constraints	Grants	Statistics	UI Defaults	Triggers	Dependencies	SQL
Create	Drop	Enable	Disable							
Constraint	Type	Search Condition	Related Constraint	Columns	Delete Rule	Status	Last Change	Index	Invalid	
SYS_C002210382	Check	"ID_AGENTA" IS NOT NULL	-	-	-	ENABLED	06/25/2022 10:21:02 AM	-	-	-
SYS_C002210383	Check	"ID_AGENCIJE" IS NOT NULL	-	-	-	ENABLED	06/25/2022 10:21:02 AM	-	-	-
JE_ZAPOSLLEN_PK	Primary	-	-	ID_AGENTA, ID_AGENCIJE	-	ENABLED	06/25/2022 10:21:02 AM	JE_ZAPOSLLEN_PK	-	-
JE_ZAPOSLLEN_AGENTI_FK	Foreign	-	JE_ZAPOSLLEN_AGENTI_FK (WKSP_AGENCIJAZANEKRETNINE.AGENTI)	ID_AGENTA	CASCADE	ENABLED	06/25/2022 10:21:02 AM	-	-	-
JE_ZAPOSLLEN_AGENCIJA_FK	Foreign	-	JE_ZAPOSLLEN_AGENCIJA_FK (WKSP_AGENCIJAZANEKRETNINE.AGENCIJA_ZA_NEKRETNINE)	ID_AGENCIJE	CASCADE	ENABLED	06/25/2022 10:21:02 AM	-	-	-

Slika 16: Ograničenja nad tablicom JE_ZAPOSLLEN

5.4. Izrada stranica aplikacije i regija unutar stranica za prikaz podataka

Nakon uspješno stvorene aplikacije, bilo je potrebno kreirati stranice koristeći opciju *Create Page*. Prilikom kreiranja stranica, u većini slučajeva, odabrana je opcija *Blank Page* koja se kasnijim dodavanjem regija i elemenata pretvorila u druge vrste stranica, primjerice Report. Nakon što smo odabrali *Blank Page*, potrebno je unijeti broj stranice (automatski se postavlja, no moguće ga je mijenjati), naziv stranice (npr. Agencija). Regije se mogu stvoriti sada ili kasnije (objašnjeno u nastavku). Ostale opcije poput grupe stranica, načina prikazivanja stranice (novi prozor, nova kartica, ista kartica) te breadcrumb-a mogu ostati kako je već postavljeno.

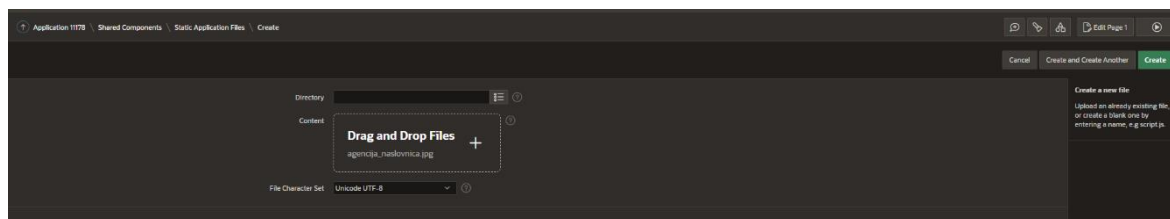
Odabrana je opcija *Navigation* kako bi se stranica stvorila u izborniku te je odabrana opcija *No parent selected* kako stranica ne bi imala roditelja. U slijedećem poglavlju 4.5. (Navigacija) objašnjena je opcija korištena roditelja u izborniku.



Slika 17: Kreiranje Blank Page-a

Regije su korištene za prikaz fotografije, teksta ili tablica. Kreiraju se na način da se u page designer-u svake stranice zasebno u lijevom stupcu na željenom čvoru klikne desni klik i odabere opcija *Create Region*. Sve regije su dodane unutar Body-a.

Umetanje fotografije vrši se na način da je prvo potrebno dodati željenu fotografiju unutar aplikacije. U samoj aplikaciji odabire se opcije *Shared Components*, zatim *Static Application Files* te *Create*. *Directory* je postavljen na *Root* te smo u odjeljku *Content* umetnuli željenu fotografiju.



Slika 18: Ubacivanje fotografije unutar aplikacije

Unutar prethodno kreirane regije kod kartice *Identification* za *Type* je odabrana opcija *Static Content* te je pod *Source* dodan odgovarajući HTML kod za prikaz fotografije.

The screenshot shows the 'Identification' section with 'Title' set to 'New' and 'Type' set to 'Static Content'. The 'Source' section shows 'HTML Code' with the following code:

```
<div align=center>

</div>
```

Slika 19: Postavke regije za prikaz fotografije

Karta je postavljena na isti način kao prethodno opisano prikazivanje slike, jedino je drugačiji HTML kod koji je preuzet s Google Karta.

The screenshot shows the 'Identification' section with 'Title' set to 'Gdje se nalazimo' and 'Type' set to 'Static Content'. The 'Source' section shows 'HTML Code' with the following code:

```
<iframe src="https://www.google.com/maps/embed?pb=!1m18!1m12!1m3!1d2804.9956806886303!2d14.46476171542537!3d45.32870967909962!2m3!1f0!2f0!3f0!3m2!1i1024!2i768!4f13.1!3m3!1m2!1s0x4764a1abeebffe433A0x14e8959b6e168b68!2sU1.%20Radmile%20Matej%C4%8Di%C4%87%202%C2%851000%C2%8Rijeka!5e0!3m2!1shr!2shr!4v1656284281648!5m2!1shr!2shr" width="800" height="600" style="border:0;" allowfullscreen="" loading="lazy" referrerpolicy="no-referrer-when-downgrade"></iframe>
```

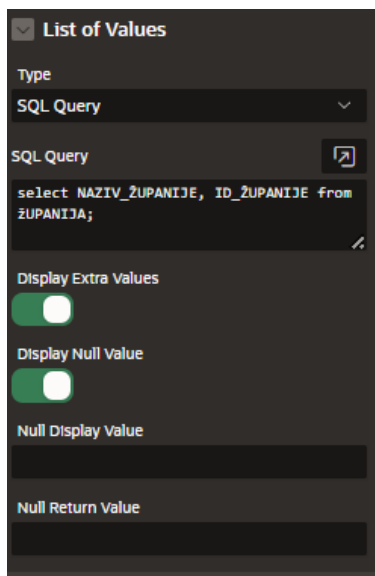
Slika 20: Postavke regije za prikaz karte

Prikaz tablice unutar stranice izrađuje se na način da se unutar novo kreirane regije te unutar kartice *Identification* za *Type* je potrebno odabrati *Classic Report*. Zatim, na kartici *Source* za *Location* odabrati *Local Database*, za *Type* odabrati *Table / View* te u *Table Name* odabrati onu tablicu koju želimo prikazati.

The screenshot shows the 'Identification' section with 'Title' set to 'AGENTI' and 'Type' set to 'Classic Report'. The 'Source' section shows 'Location' set to 'Local Database' and 'Type' set to 'Table / View'.

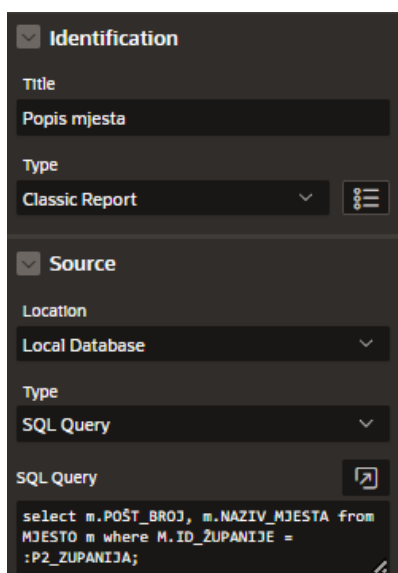
Slika 21: Prikaz postavki za prikaz tablice AGENTI

Prikaz mjesta s obzirom na županiju unutar koje se nalazi ostvareno je na način da su kreirane dvije regije unutar Body-a. Prva regije naziva *Popis županija* sadrži *Hot Text button* i *Select List*. *Select List* ima naziv *P2_ZUPANIJA* te unutar kartice *List of Values* napisati odgovarajući SQL upit.



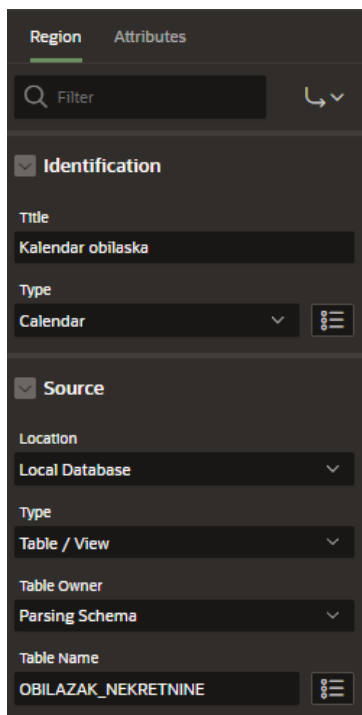
Slika 22: Prikaz SQL upita za tablicu ŽUPANIJA

Kod druge regije (Popis mjesta) na kartici *Identification* pod opcijom *Type* potrebno odabrati *Classic Report*. Zatim, na kartici *Source* za *Location* odabrati *Local Database*, za *Type* odabrati *SQL Query* te napisati SQL upit u odjeljku *SQL Query*.

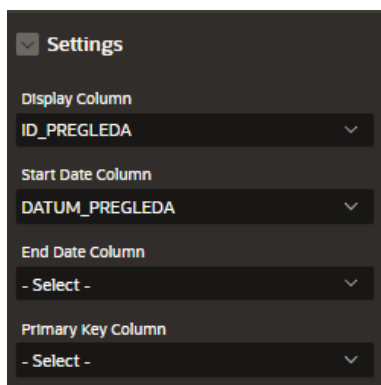


Slika 23: Prikaz SQL upita za prikaz mjesta s obzirom na odabranu županiju

Što se tiče kalendara za pregled obilaska, na novoj regiji na kartici *Region*, pod odjeljkom *Identification* te kod svojstva *Type* potrebno je odabrati *Calendar* te kod odjeljka *Source* odabrati željenu tablicu. Na kartici *Attributes* kod odjeljka *Settings* potrebno je odabrati koji stupac želimo da se prikaže te onaj stupac koji sadrži vrijednost datuma.



Slika 24: Postavke kalendara na kartici Region



Slika 25: Postavke kalendara na kartici Attributes

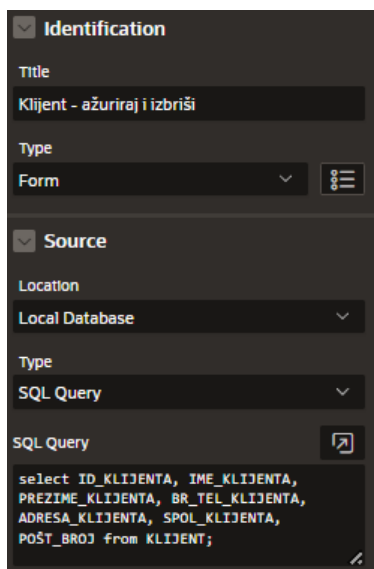
5.5. Izrada stranica aplikacije i regija unutar stranica za prikaz, uređivanje i brisanje podataka

Većina stranica koje omogućavaju unos, ažuriranje i brisanje podataka je izrađena uz pomoć čarobnjaka za kreiranje *Interactive Grid-a* i *Master Detail-a*. U sklopu *Interactive Grid Page-a* i *Master Detail Page-a* ubačen je i *Form Page*. Izvješća su samo za čitanje, ali obrasci mogu mijenjati podatke unutar baze podataka.

5.5.1. Form view

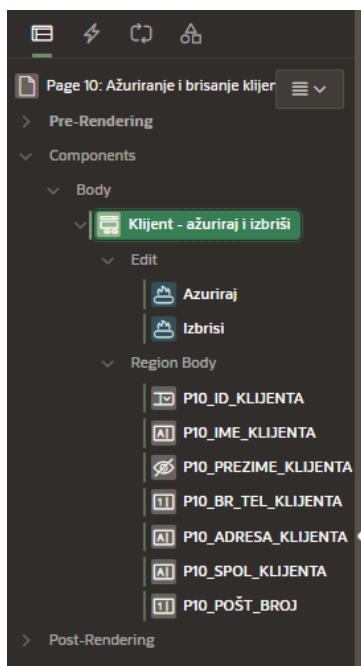
Form view omogućava ažuriranje i brisanje određene n-torke unutar tablice na način da se unutar padajućeg izbornika odabere prezime klijenta te se prikazu svi njegovi podaci koje je moguće ažurirati. Korisnik može tada mijenjati vrijednosti stavki po želji. Klikom na gumb Ažuriraj izmijenit ćete taj zapis da ima te vrijednosti. Klikom na gumb Izbriši izbrisat ćete zapis.

Nakon stvaranja nove regije unutar Body-a potrebno je za *Type* odabrati *Form* te ponovno za *Source* postaviti *Local Database*, *SQL Query* za *Type* te unijeti odgovarajući SQL upit.



Slika 26: Postavke nove regije

Unutar novo stvorene regije dodana su dva *Hot Text* gumba za uređivanje i brisanje. Također, polje koje sadrži *Id_klijenta* je postavljeno na *Select list* te pomoću SQL upita je napravljeno da se odabere prezime klijenta. Iz navedenog razloga je polje koje sadrži *Prezime_klijenta* postavljeno na sakriveno, točnije njegov *Type* je *Hidden*. Unutar odjeljka *Settings* kod *Page Action on Selection* odabire se opcija *Redirect and Set Value*. Također, kod odjeljka *Source* potrebno je uključiti opciju *Primary Key*.



Slika 27: Hidden za prezime i Select list za id

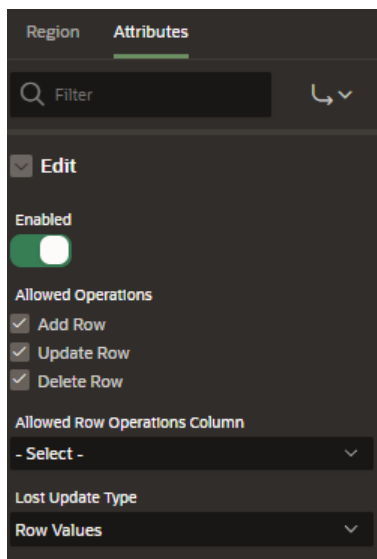
The screenshot shows the 'Identification' settings for an Id field in Oracle APEX. The settings are organized into sections: Identification, Label, Settings, and Layout. Under 'Identification', the 'Name' is set to 'P10_ID_KLIJENTA' and the 'Type' is 'Select List'. Under 'Label', the 'Label' is 'Izaberi klijenta'. Under 'Settings', the 'Page Action on Selection' is 'Redirect and Set Value'. Under 'Layout', the 'Sequence' is '10', the 'Region' is 'Klijent - ažuriraj i izbriši', the 'Position' is 'Region Body', and the 'Start New Row' toggle is turned on.

Slika 28: Postavke za Id polje

The screenshot shows the 'Source' settings for an Id field in Oracle APEX. The settings are organized into sections: Source, Form Region, Column, Data Type, Query Only, Primary Key, and Maintain Session State. Under 'Source', the 'Form Region' is 'Klijent - ažuriraj i izbriši'. Under 'Column', the 'Column' is 'ID_KLIJENTA'. Under 'Data Type', the 'Data Type' is 'NUMBER'. Under 'Query Only', the 'Query Only' toggle is turned off. Under 'Primary Key', the 'Primary Key' toggle is turned on. Under 'Maintain Session State', the 'Maintain Session State' is 'Per Request (Memory Only)'.

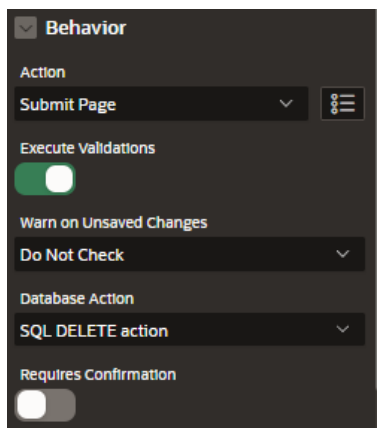
Slika 29: Source za Id polje

Ova forma zahtjeva postavljanje behavior-a na dodanim gumbima (*Hot Text button*) za brisanje i uređivanje. Na kartici *Attributes* od regije potrebno je omogućiti uređivanje te dozvoliti operacije poput ažuriranja i brisanja redaka unutar tablice.

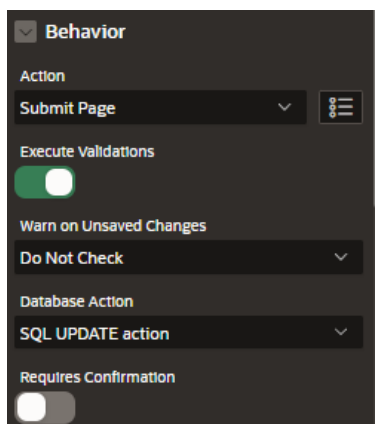


Slika 30: Omogućava dozvola za uređivanje i brisanje unutar regije

Nadalje, postavlja se *Database Action* na *SQL DELETE action* (za brisanje podataka unutar baze podataka) ili *SQL UPDATE action* (za ažuriranje podataka unutar baze podataka).

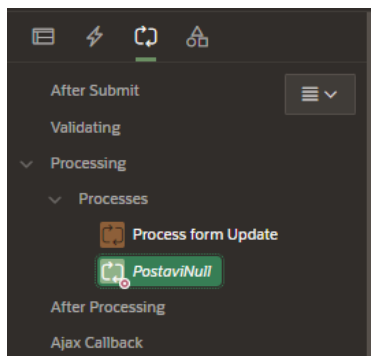


Slika 31: Behavior na gumbu za brisanje

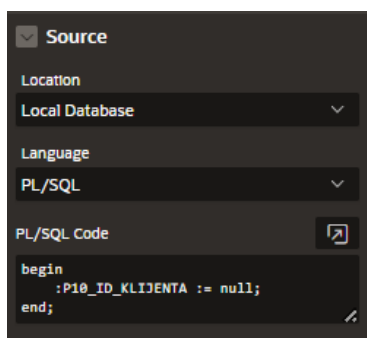


Slika 32: Behavior na gumbu za ažuriranje

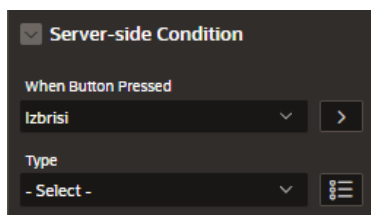
Stvoren je novi proces *PostaviNull* unutar kojeg je u odjeljku *Source* napisan odgovarajući PL/SQL kod, odnosno postavili vrijednost polja *id_klijent* na null. Navedeni proces osigurava da *P10_ID_KLIJENTA* više ne sadržava vrijednost člana kojeg izbrišemo pritiskom na gumb. Zadnji korak je postaviti da se proces poziva kada korisnik klikne na gumb. To je ostvareno na način da je kod određenog procesa (*PostaviNull*) u odjeljku *Server-side Condition* potrebno odabrati kod opcije *When Button Pressed* naziv gumba.



Slika 33: Ubacivanje procesa *PostaviNull*



Slika 34: Postavke *Source*-a od procesa *PostaviNull*

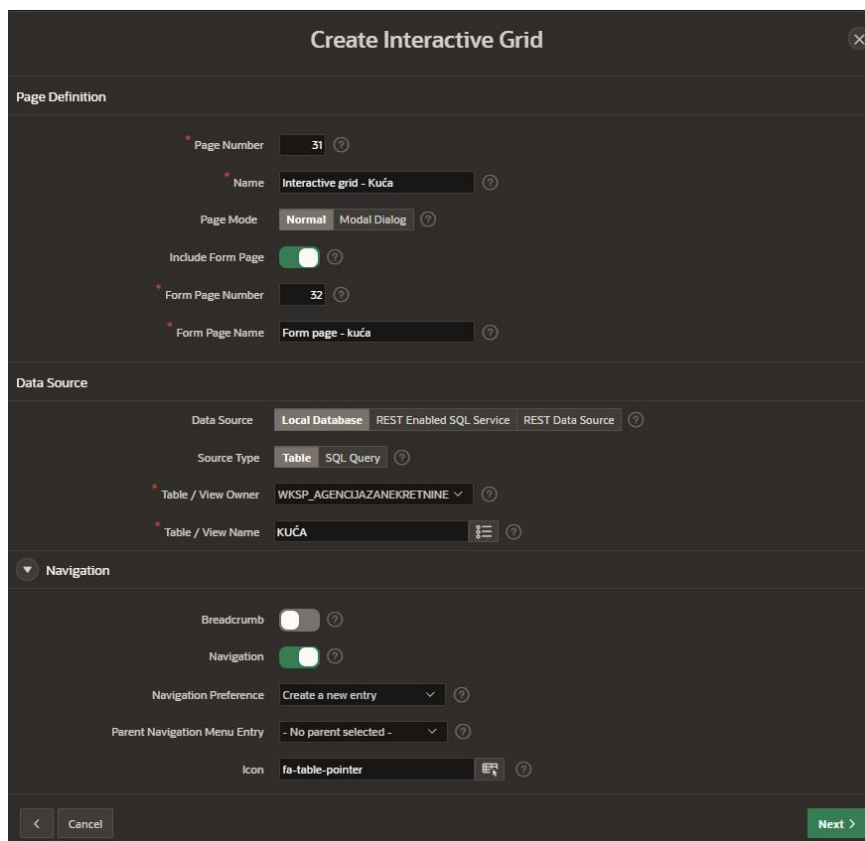


Slika 35: Postavljanje procesa na gumb

5.5.2. Report with form

Kako bi korisnik mogao pregledavati, ažurirati, umetati i brisati podatke koristeći aplikaciju potrebno je izraditi izvješće i forme unutar aplikacije. U interaktivnom izvješću korisnik smije pregledavati podatke koji se trenutno nalaze u bazi podataka. Ukoliko korisnik želi urediti postojeće podatke ili stvoriti nove, otvara se forma. Crvenom bojom su označena obavezna polja koja korisnik treba popuniti. Za uređivanje postojećih podatak potrebno je odabrati simbol olovke, a stvaranje novih podataka vrši pomoću gumba CREATE.

Tijekom kreiranja stranice odabrana je opcija *Interactive grid* te je uključena opcija za stvaranje *Form page-a*. Također, navedena je tablice iz koje će se povlačiti podaci. Za primarni ključ postavljen je *ID_KUĆE*. Unutar forme koja je stvorena za uređivanje potrebno je postaviti na vidljivo polje za unos koje predstavlja primarni ključ te namjestiti da je ono primarni ključ. Postavljanje na vidljivo se vrši na način da se *Type* od polja u kojem se nalazi primarni ključ umjesto *Hidden* staviti u *Number field* te na kartici *Appearance* u polju *Template* postaviti *Required – Floating*.



Slika 36: Kreiranje *Interactive Grid-a*

5.5.3. Master Detail

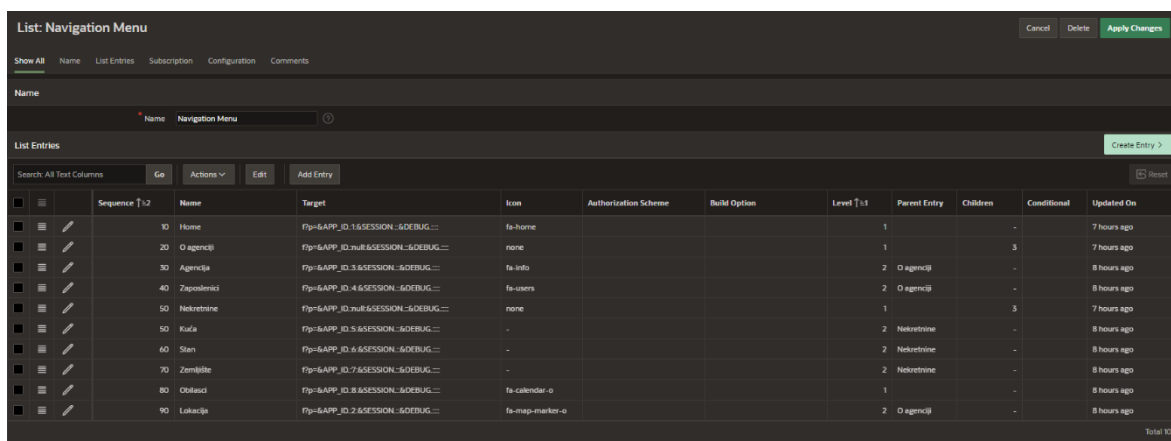
Master Detail je skup od dva izvješća. Koristi se kod složenih tipova entiteta kao što su agregacija ili slabi tip. Ova izvješća su povezana u smislu da svaki redak jednog izvješća ima pridruženi redak u drugom izvješću. Korisnik stupa u interakciju s izvješćima odabirom jednog od glavnih redaka. Ukoliko se odabere ime i prezime agenta, onda se prikazuju podaci o njemu, ali i o agregaciji u kojoj sudjeluje te se na taj način dobivaju informacije o osnovnoj plaći, proviziji, datumu zaposlenje i slično.

Izrađen je *Master detail* izvještaj za složenu tablicu *JE_ZAPOSLLEN*. Za kreiranje navedenog upotrijebljen je čarobnjak za stvaranje stranica te je odabrana *Master Detail forma*. Za primarnu tablicu odabrana je tablica *AGENTI*, a za sekundarnu tablicu odabrana je tablica *JE_ZAPOSLLEN*. Jedine izmjene bile su potrebne da stranici gdje se kreira agent zato što je trebalo *Type* od polja u kojem se nalazi id korisnika umjesto *Hidden* staviti u *Number field* te na kartici *Appearance* u polju *Template* postaviti *Required – Floating*.

5.6. Izrada navigacije

[illegible]

Posljednji korak pri ostvarivanju hijerarhijskog prikaza bio je određene stranice, poput *Agencija*, *Zaposlenici*, *Lokacija* staviti pod roditelja *O agenciji*.

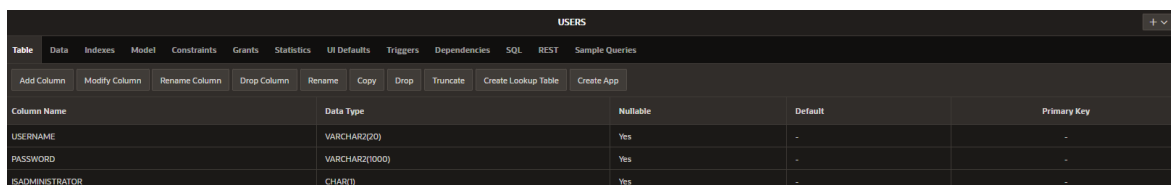


Sequence	Name	Target	Icon	Authorization Scheme	Build Option	Level	Parent Entry	Children	Conditional	Updated On
10	Home	f?p=6APP_ID:16SESSION:6DEBUG::	fa-home			1		-		7 hours ago
20	O agenciji	f?p=6APP_ID:16SESSION:6DEBUG::	none			1		3		7 hours ago
30	Agencija	f?p=6APP_ID:36SESSION:6DEBUG::	fa-info			2	O agenciji	-		8 hours ago
40	Zaposlenici	f?p=6APP_ID:46SESSION:6DEBUG::	fa-users			2	O agenciji	-		8 hours ago
50	Nekretnine	f?p=6APP_ID:56SESSION:6DEBUG::	none			1		3		7 hours ago
60	Kuća	f?p=6APP_ID:56SESSION:6DEBUG::	-			2	Nekretnine	-		8 hours ago
60	Stan	f?p=6APP_ID:66SESSION:6DEBUG::	-			2	Nekretnine	-		8 hours ago
70	Zemljište	f?p=6APP_ID:76SESSION:6DEBUG::	-			2	Nekretnine	-		8 hours ago
80	Odlasci	f?p=6APP_ID:86SESSION:6DEBUG::	fa-calendar-o			1		-		8 hours ago
90	Lokacija	f?p=6APP_ID:26SESSION:6DEBUG::	fa-map-marker-o			2	O agenciji	-		8 hours ago

Slika 39: Prikaz navigacije

5.7. Autentikacija i autorizacija

Po pitanju autentikacije, na samom početku izrađena je zasebna tablica za spremanje podataka o korisniku te sadrži atribute poput korisničkog imena, lozinke i podatka je li korisnik administrator. Tablica USERS ne sadrži nikakva ograničenja, primjerice primarni ključ.



Column Name	Data Type	Nullable	Default	Primary Key
USERNAME	VARCHAR2(20)	Yes	-	-
PASSWORD	VARCHAR2(1000)	Yes	-	-
ISADMINISTRATOR	CHAR(1)	Yes	-	-

Slika 40: Prikaz tablice USERS

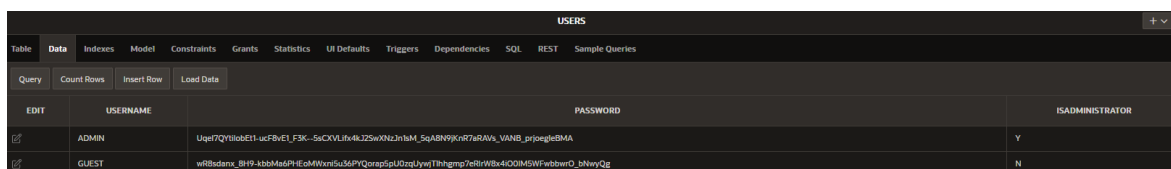
Koristeći SQL upit, prikazan na slici ispod, unesena su dva standardna korisnika u bazu, točnije administrator i gost. Jedan od korisnika je *admin* koji se prijavljuje s zaporkom *12345*, a drugi korisnik je *guest* kojem nije potrebna lozinka.



```
1 insert into USERS (Username, Password, IsAdministrator) values ('ADMIN', apex_util.get_hash(apex_util.get_hash('ADMIN', '12345'), null), 'Y');
2 insert into USERS (Username, Password, IsAdministrator) values ('GUEST', apex_util.get_hash(apex_util.get_hash('GUEST', null), null), 'N');
```

Slika 41: Umetanje korisnika u bazu podataka

Zbog povećanja sigurnosti korišteno je haširanje lozinki pomoću već implementirane metode *apex_util.get_hash*. Na slijedećoj slici se možemo uvjeriti da metoda zbilja ispravno radi.

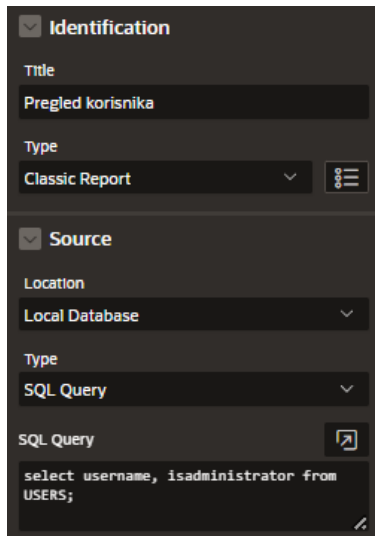


USERNAME	PASSWORD	ISADMINISTRATOR
ADMIN	Uqet7QYtllobEt-ucFbVtE1_F3K--SiCKXVltv4tJ2SwXNzJnIsM_SqABN9jKnr7aRVA_VANB_pjoeleBMA	Y
GUEST	wR8sdanv_8H9-4bbMe6PH-EokMwn5u36PYQorapSpU0zqJywtThhgmp7eRiW8w4QDIMSfWfbwwO_bNvyQg	N

Slika 42: Prikaz podataka korisničkog imena i lozinke korisnika

Stvorene su dvije nove stranice od kojih je jedna za prijavu korisnika, a druga za stvaranje novih i pregled postojećih korisnika. Prilikom pregleda postojećih korisnika

prikazuju se podaci poput korisničkom imena i je li korisnik administrator (Y označava da, N označava ne). Iako je lozinka pohranjena u haširanom obliku, ona se ne prikazuje iz sigurnosnih razloga.



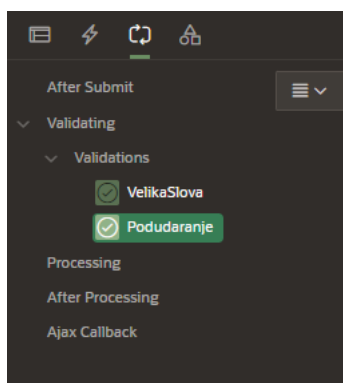
Slika 43: Prikaz postavki regije za pregled korisnika

Za unos korisnika korišten je prethodno objašnjen regija *Static Content* unutar koje se nalaze item-i poput *Text Field-a*, *Password-a*, te *Hot Text* gumba. Kreiran je proces koji osigurava haširanje zaporka prilikom kreiranja novog korisnika.

```
1 declare
2   v_username varchar2(20) := upper(:P15_USERNAME);
3   v_valuesToHash apex_t_varchar2 := apex_t_varchar2(v_username, :P15_PASSWORD);
4 begin
5   insert into USERS (UserName, Password, IsAdministrator) values (v_username, apex_util.get_hash(v_valuesToHash, null), :P15_ADMIN);
6 end;
```

Slika 44: Unos novog korisnika u bazu podataka i haširanje lozinke

Stvorene su dvije validacije od kojih jedna zahtjeva spremanje svih korisničkih imena u obliku velikih slova, dok druga zahtjeva podudaranje između dva korisnikova unosa lozinke. Validacije se stvaraju na kartici *Processing* pod odjeljkom *Validating* u rubrici *Validations*.



Slika 45: Stvaranje validacije

The screenshot shows the Oracle APEX Validation Wizard for a validation named 'Greska_validation1'. The 'Identification' section shows the name 'VelikaSlova'. The 'Execution Options' section shows a sequence of 10. The 'Validation' section shows the 'Editable Region' as '- Select -', the 'Type' as 'No Rows returned', and the 'SQL Query' as 'select * from USERS where UserName = upper(:P19_USERNAME)'. The 'Always Execute' toggle is turned off. The 'Error' section shows the 'Error Message' as 'Greska_validation1'.

Identification

Name
VelikaSlova

Execution Options

Sequence
10

Validation

Editable Region
- Select -

Type
No Rows returned

SQL Query
select * from USERS where UserName = upper(:P19_USERNAME)

Always Execute
☐

Error

Error Message
Greska_validation1

Slika 46: Validacija spremanja korisnikovog imena u obliku velikih slova

The screenshot shows the Oracle APEX Validation Wizard for a validation named 'Lozinke se ne podudaraju!'. The 'Identification' section shows the name 'Podudaranje'. The 'Execution Options' section shows a sequence of 20. The 'Validation' section shows the 'Editable Region' as '- Select -', the 'Type' as 'Expression', the 'Language' as 'PL/SQL', and the 'PL/SQL Expression' as '(:P19_PASSWORD = :P19_CONFIRM) or (:P19_PASSWORD is null and :P19_CONFIRM is null)'. The 'Always Execute' toggle is turned off. The 'Error' section shows the 'Error Message' as 'Lozinke se ne podudaraju!'.

Identification

Name
Podudaranje

Execution Options

Sequence
20

Validation

Editable Region
- Select -

Type
Expression

Language
PL/SQL

PL/SQL Expression
(:P19_PASSWORD = :P19_CONFIRM) or (:P19_PASSWORD is null and :P19_CONFIRM is null)

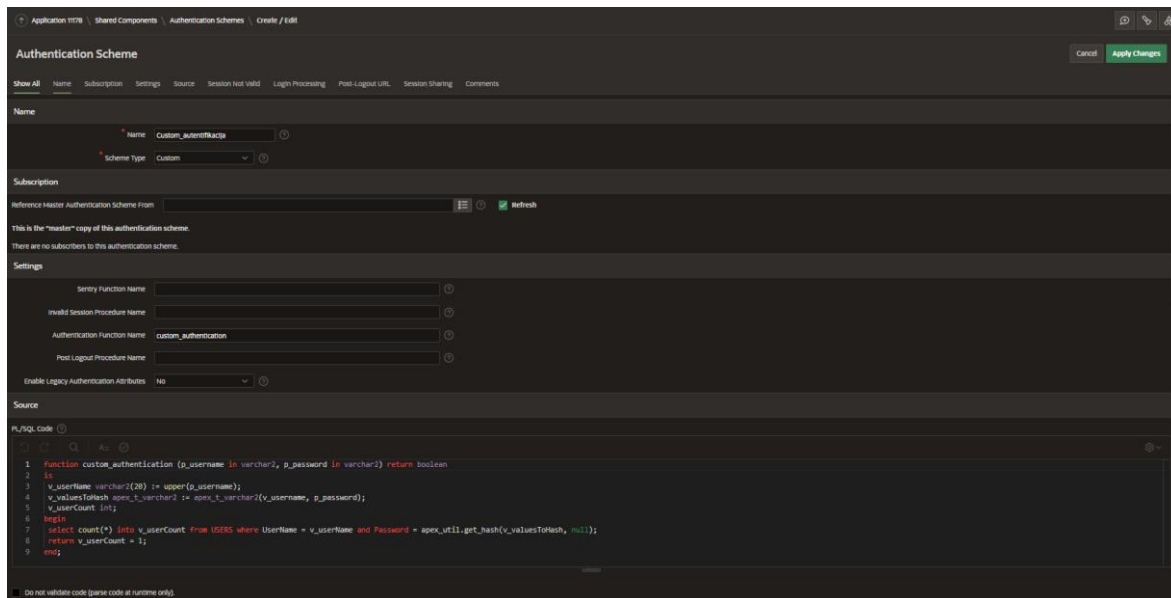
Always Execute
☐

Error

Error Message
Lozinke se ne podudaraju!

Slika 47: Validacija podudaranja lozinka

Idući korak je stvaranje autentikacijske scheme. Unutar *Shared Components* nalazi se rubrika *Authentication Schemes* gdje se odabere opcija *Create (Based on a pre-configured scheme from the gallery)*. Popune se polja s nazivom, vrstom scheme (*Custom*), nazivom funkcije te se unese odgovarajući PL/SQL kod.

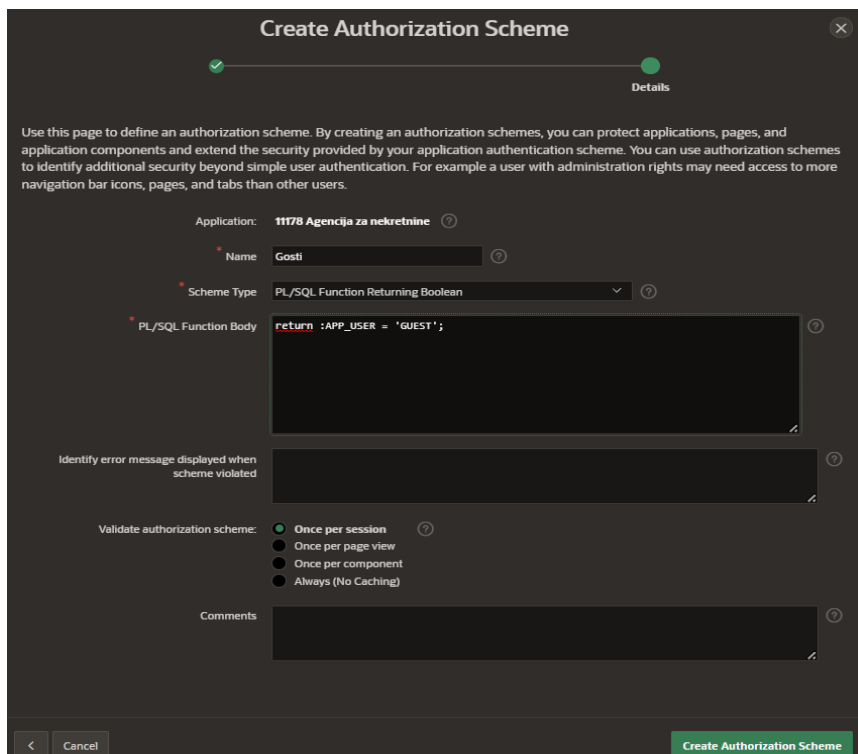


The screenshot shows the 'Create Authentication Scheme' dialog in Oracle APEX. The 'Name' field is 'Custom_Autentifikacija', 'Scheme Type' is 'Custom', and 'Authentication Function Name' is 'custom_authentication'. The 'Source' tab shows the following PL/SQL code:

```
1 function custom_authentication (p_username in varchar2, p_password in varchar2) return boolean
2 is
3   v_username varchar2(30) := upper(p_username);
4   v_valuesToHash apex_t_varchar2 := apex_t_varchar2(v_username, p_password);
5   v_userCount int;
6 begin
7   select count(*) into v_userCount from USERS where Username = v_username and Password = apex_util.get_hash(v_valuesToHash, null);
8   return v_userCount = 1;
9 end;
```

Slika 48: Stvaranje autentikacijske scheme

Nakon napravljene autentikacije slijedi autorizacija gdje je potrebno unutar *Shared Components* u rubrici *Authorization Schemes* odabrati opciju *Create (From Scratch)*. Popunjavaju se polja koja sadrže naziv autorizacijske scheme, vrstu autorizacijske scheme (*PL/SQL Function Returning Boolean*) te odgovarajući *PL/SQL Function Body*.



The screenshot shows the 'Create Authorization Scheme' dialog in Oracle APEX. The 'Application' is '11178 Agencija za nekretnine', 'Name' is 'Gosti', 'Scheme Type' is 'PL/SQL Function Returning Boolean', and the 'PL/SQL Function Body' is 'return :APP_USER = 'GUEST';'. The 'Validate authorization scheme' section has 'Once per session' selected. The 'Comments' field is empty.

Slika 49: Prikaz autorizacijske scheme za goste

Vanessa Siljan: Agencija za nekretnine - poslovna aplikacija nad relacijskom bazom podataka (Oracle APEX)

The screenshot shows the 'Authorization Scheme' configuration page in Oracle APEX. The page is titled 'Authorization Scheme' and has tabs for 'Show All', 'Name', 'Subscription', 'Authorization Scheme', 'Evaluation Point', and 'Comments'. The 'Name' tab is selected. The configuration includes: Application: 11178 Agencija za nekretnine; Name: Administratori; Subscription: Reference Master Authorization Scheme From (with a Refresh button); Authorization Scheme: Scheme Type: Exists SQL Query; SQL Query: select * from USERS where Username = :APP_USER and IsAdministrator = 'Y'; Identify error message displayed when scheme violated: Pristup dozvoljen samo administratorima!; Evaluation Point: Validate authorization scheme: Once per session, Once per page view (selected), Once per component, Always (No Caching); Comments: (empty text area).

Slika 50: Prikaz autorizacijske scheme za administratore

Kao što prikazuje sljedeća slika, autorizacija je primijenjena na razini cijele aplikacije osim na javnim stranicama poput Početna, Lokacija, Agencija, Kuća, Stan, Zemljište, O agenciji, Login page gdje je unutar kartice *Security* pod odjeljkom *Authentication* potrebno odabrati opciju *Page Is Public*.

The screenshot shows the 'Application 11178' configuration page in Oracle APEX, specifically the 'Authorization' tab. The page has tabs for 'Show All', 'Authentication', 'Authorization', 'Session Management', 'Session State Protection', 'Browser Security', and 'Database Session'. The 'Authorization' tab is selected. The configuration includes: Authorization Scheme: Administratori; Run on Public Pages: (toggle off); Run on Background Job: (toggle on); Source for Role or Group Schemes: Access Control User Role Assignments.

Slika 51: Primjena autorizacijske scheme na razinu cijele aplikacije

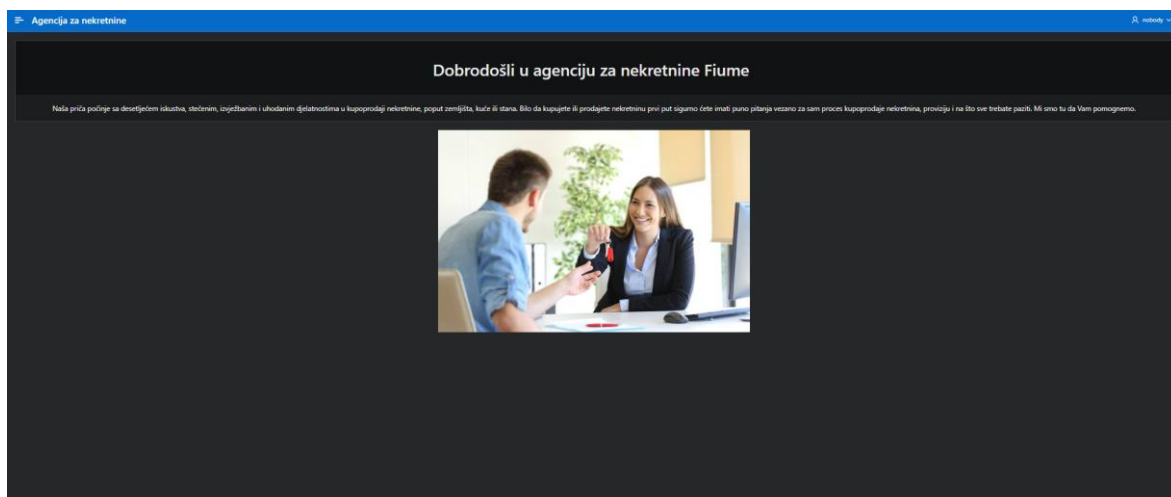
The screenshot shows the 'Security' settings page in Oracle APEX. The page has a 'Security' section with the following settings: Authorization Scheme: - Select -; Authentication: Page Is Public; Deep Linking: Application Default; Page Access Protection: Arguments Must Have Checksum; Form Auto Complete: Off; Browser Cache: Application Default.

Slika 52: Postavljanje stranice na javno

6. Prikaz kreirane aplikacije

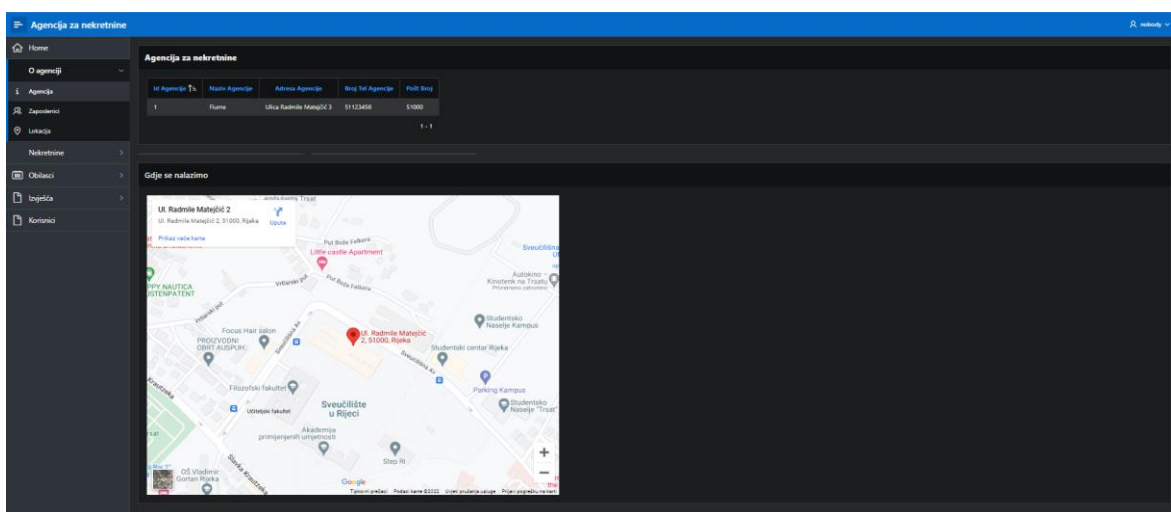
Pregled aplikacije moguć je na slijedećem linku: https://apex.oracle.com/pls/apex/r/agencija_za_nekretnine/agencija-za-nekretnine/home?session=0. Moguća je prijava u aplikaciju s korisničkim imenom *admin* te korisničkom zaporkom *12345*. U nastavku su prikazane neke od fotografija završne verzije aplikacije za nekretnine.

Na slici 53. prikazana je početna stranica s kratkom porukom koja se prikazuje na samom početku.



Slika 53: Prikaz početne stranice

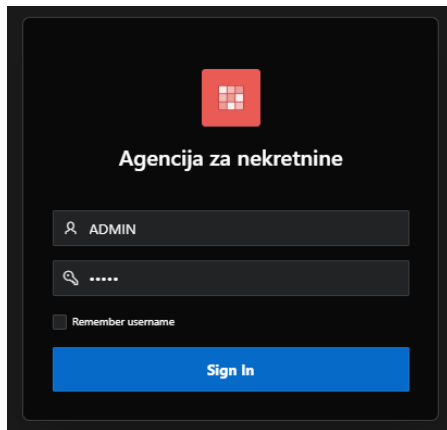
Na slici 54. prikazan je ekran koji se učitava ukoliko korisnik unutar navigacije odabere karticu *Agencija*. Prikazana je tablica koja sadrži osnovne podatke o agenciji poput njenog id-a, naziva agencije, adrese, broja telefona te poštanskog broja. Također, prikazuje se karta te je označena lokacija agencije.



Slika 54: Prikaz informacija o agenciji

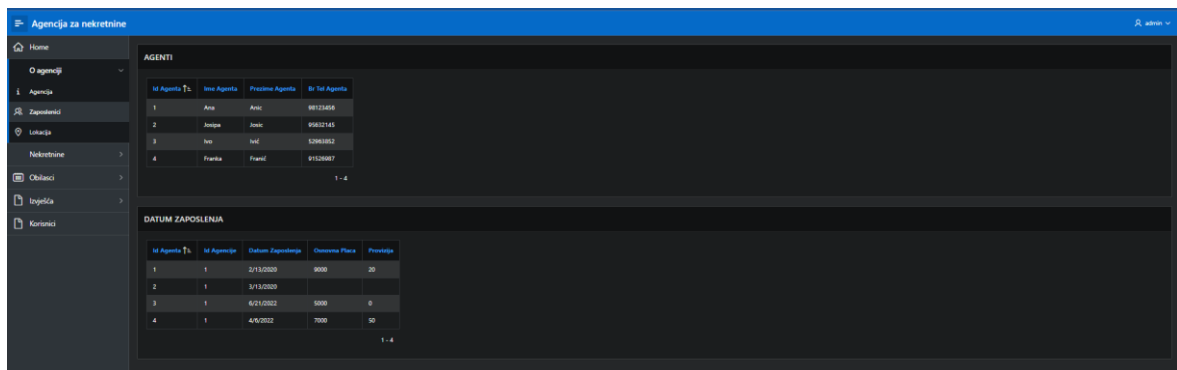
Na slijedećoj fotografiji prikazan je prozor za prijavu korisnika (administratora) u aplikaciju kako bi mogao vršiti izmjene unutar baze podataka poput brisanja, ažuriranja ili

umetanja novih podataka. Tijekom prijave, korisnik je dužan unijeti podatke poput korisničkog imena i lozinke. Također, pruža mu se mogućnost spremanja prethodno popunjenih podataka.



Slika 55: Prikaz login stranice

Slijedeća fotografija prikazuje stranicu Zaposlenici gdje je moguć pregled osnovnih podataka o agentu poput njegovog id-a, imena, prezimena te telefonskog broja. Unutar druge tablice prikazane niže dolje, moguć je pregled datuma zaposlenja pojedinog agenta te iznosa njegove plaće i provizije.

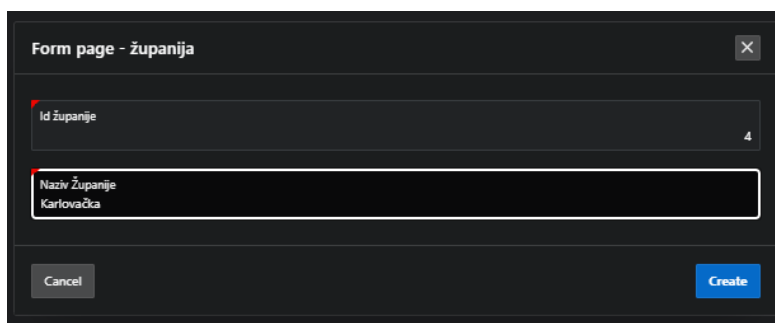


Id Agent	Ime Agent	Prezime Agent	Br Tel Agent
1	Ava	Avic	98731458
2	Josipa	Jovic	90832145
3	Iva	ivic	52943892
4	Franka	Frank	91238987

Id Agent	Id Agencije	Datum Zaposlenja	Mesečna Plaća	Provizija
1	1	2/13/2020	8000	20
2	1	3/13/2020		
3	1	6/21/2022	5000	0
4	1	4/8/2022	7000	30

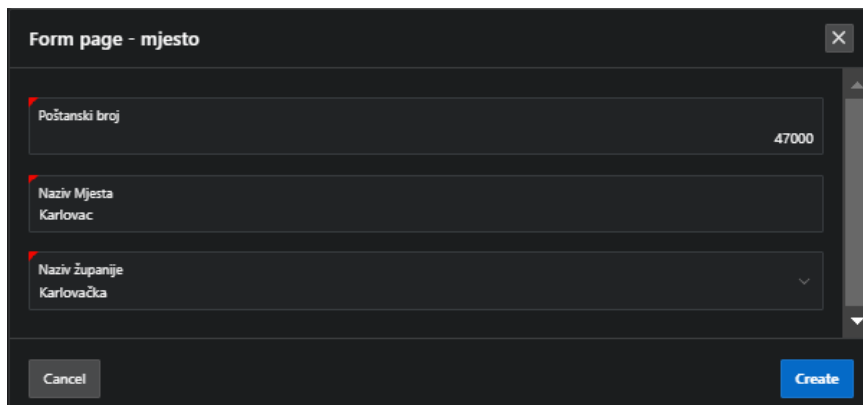
Slika 56: Prikaz informacija o zaposlenicima

Na slici 57. prikazano je kreiranje nove županije. Kako bi se kreirala nova županija potrebno je unijeti podatke poput njenog id-a i naziva županije.



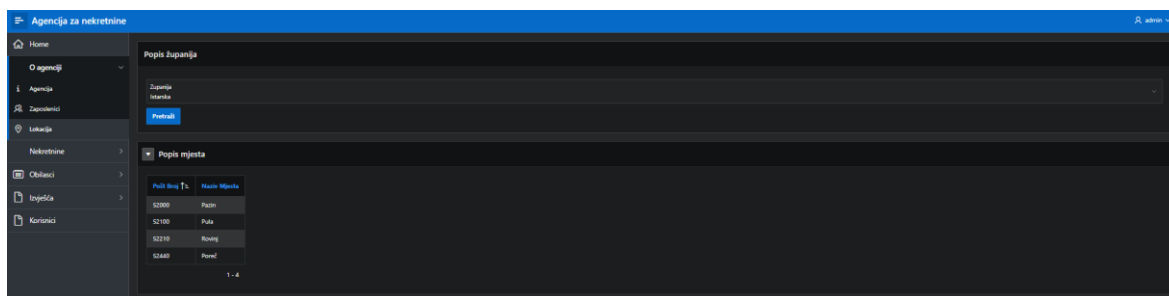
Slika 57: Prikaz umetanja županije

Slijedeća fotografija prikazuje kreiranje mjesta gdje je korisnik dužan unijeti poštanski broj, naziv mjesta te odabrati naziv županije u kojem se mjesto nalazi.



Slika 58: Prikaz umetanja mjesta

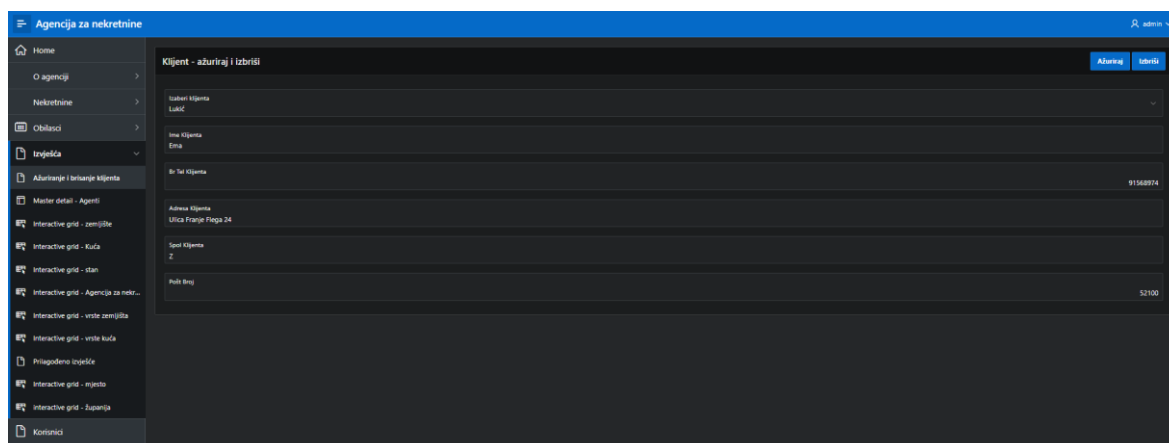
Na slijedećoj fotografiji moguće je filtrirati mjesta s obzirom na naziv županije u kojoj se mjesto nalazi. Korisnik treba odabrati naziv županije te prilikom pritiska na gumb *Pretraži*, prikazati će se sva mjesta koja se nalaze u željenoj županiji.



Pošt. broj	Naziv Mjesta
52000	Paštin
52100	Paštin
52210	Rešanj
52400	Perut

Slika 59: Prikaz filtriranja mjesta s obzirom na županiju

Iduća fotografija omogućava administratoru ažuriranje i brisanje podataka o određenom klijentu kojeg administrator odabire iz padajućeg izbornika. Brisanje i ažuriranje podataka se vrši uz pomoć gumba koji se nalaze gore desno.



Slika 60: Prikaz obrasca za ažuriranje i brisanje određenog klijenta

Na slici 61. moguće je primijetiti obrazac za dodavanje podataka o novoj kući unutar baze podataka. Kako bi se uspješno izvršio unos, potrebno je popuniti sva obavezna polja koja su označena crvenom bojom te potvrditi započetu radnju pritiskom na gumb *Create*.

Form page - kuća

Id kuće 4

Adresa Kuće
Valvidalska ulica 56

Broj Soba Kuće 4

Površina Kuće 85

Okućnica
DA

Cijena Kuće 75000

Bazen
DA

Garaža
NE

Agencija
Fiume

Vista kuće
Dvojna

Grad
Pula

Cancel Create

Slika 61: Prikaz obrasca za umetanje kuće

Slijedeća fotografija prikazuje isti obrazac kao što je bio onaj prethodno objašnjen, no ovaj obrazac omogućava ažuriranje i brisanje podataka o odabranoj kući. Moguće je i odustati od promjena pritiskom na gumb *Cancel*.

Form page - kuća

Id kuće 3

Adresa Kuće
Valturska ulica 5

Broj Soba Kuće 5

Površina Kuće 105

Okućnica
DA

Cijena Kuće 89000

Bazen
NE

Garaža
DA

Agencija
Fiume

Vrsta kuće
Samostojeća

Grad
Pula

Cancel Delete Apply Changes

Slika 62: Prikaz obrasca za uređivanje i brisanje kuća

Slijedeće prikazana fotografija korisniku omogućuje pregled svih kuća i njezinih osnovnih podataka poput adrese, broja soba, površine, cijene, mjesta u kojem se nalazi, vrsti kuće kojoj pripada i tomu slično. Također, postoji polje za pretraživanje po adresi na kojoj se kuća nalazi. Gore desno nalazi se gumb *Create* koji korisniku pruža mogućnost kreiranja nove kuće. U svakom retku nalazi se mogućnost uređivanja određene kuće. Ukoliko se odabere opcija uređivanja, korisniku se prikazuje prethodno prikazana fotografija.

Agencija za nekretnine

Home O agenciji Nekretnine Kuća Stan Zastigle Obitelji Imenja Kontakti

Interactive grid - Kuća

Search All Text Columns

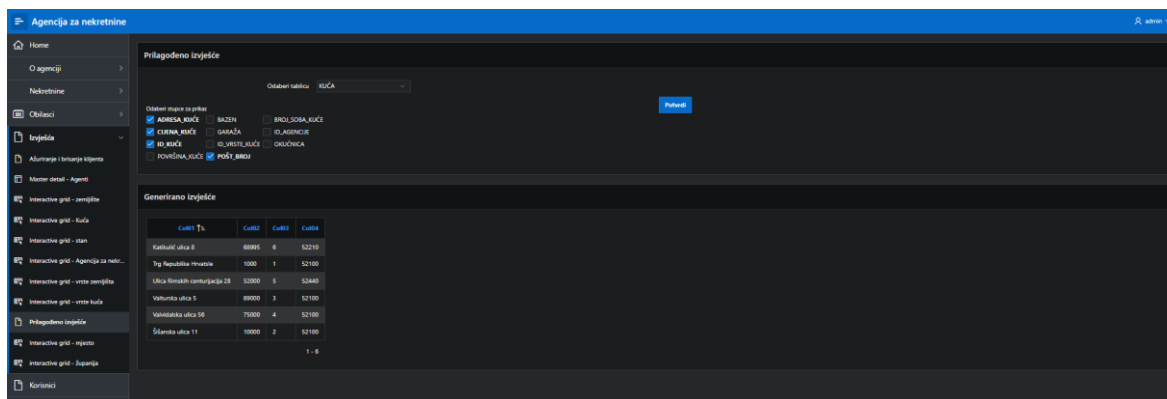
	Adresa kuće	Broj soba	Površina	Okućnica	Cijena	Bazen	Garaža	Naziv agencije	Vrsta kuće	Mjesto
	Valturska ulica 5	5	105	DA	89000	NE	DA	Fiume	Samostojeća	Pula
	Ulica Rimskih vojnika 28	2	54	NE	12000	NE	NE	Fiume	Samostojeća	Horat
	Tig Republike Hrvatske	3	75	NE	10000	NE	NE	Fiume	Obitelj	Pula
	Čičakova ulica 11	3	98	DA	10000	NE	DA	Fiume	Obitelj	Pula
	Vakarska ulica 16	4	85	DA	75000	DA	NE	Fiume	Dvoje	Pula
	Katolič ulica 8	4	78	DA	68000	NE	NE	Fiume	Samostojeća	Horat

1 rows selected

Total: 6

Slika 63: Prikaz popisa svih kuća

Iduća fotografija prikazuje prilagođeno izvješće koje se dinamički stvara na način da korisnik odabere tablicu iz padajuće liste te označi sva polja koja želi da se prikažu u izvješću. Kako bi se radnja uspješno izvršila, potrebno je kliknuti na gumb *Potvrdi*.



Agencija za nekretnine

Prilagođeno izvješće

Odaberi mapu za prikaz:

Odaberi mapu za prikaz:

- ☒ ADRESA_KUĆE
- ☐ BAZEN
- ☐ BROJ_SOBAMA_KUĆE
- ☒ CIJENA_KUĆE
- ☐ GARAJZA
- ☐ KLASIFIKACIJE
- ☒ KUĆA_KUĆE
- ☐ KLASIFIKACIJE_KUĆA
- ☐ OBLASTI_KUĆE
- ☒ PLOŠĆINA_KUĆE
- ☐ POŠTA_BROJ

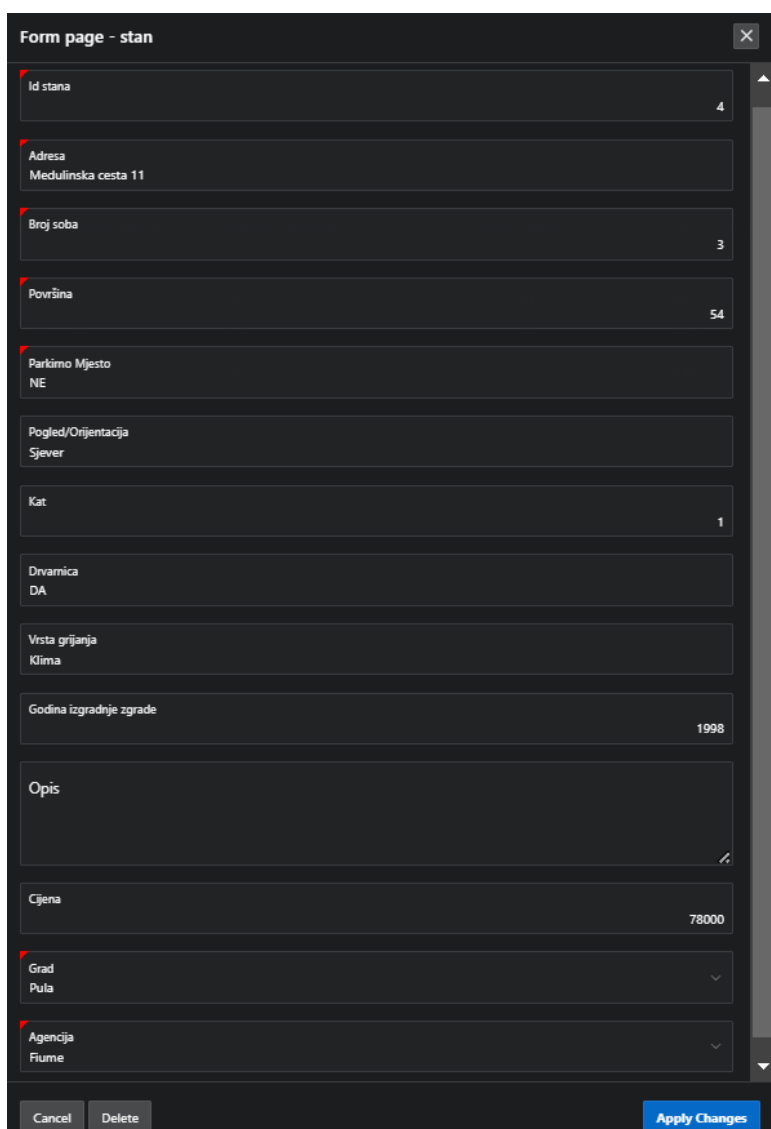
Generirano izvješće

Col1	T1	Col2	Col3	Col4
Katolički ulica 8	88995	6	52210	
Trg Republike Hrvatske	1000	1	52100	
Ulica Rimskih carinijarja 28	52000	5	52440	
Vahunka ulica 5	88000	2	52100	
Vahunka ulica 56	75000	4	52100	
Škarska ulica 11	10000	2	52100	

1 - 6

Slika 64: Prikaz prilagođenog izvješća za tablicu KUĆA

Na slici 65. moguće je ažurirati ili obrisati podatke o određenom stanu. Ažuriranje se vrši na način da korisnik izmjeni željene podatke te odabere gumb *Apply Changes*. Ukoliko se korisnik odluči za brisanje pojedinog stana, potrebno je kliknuti na gumb *Delete*.



Form page - stan

Id stana: 4

Adresa: Medulinska cesta 11

Broj soba: 3

Površina: 54

Parkirno Mjesto: NE

Pogled/Orijentacija: Sjever

Kat: 1

Divarnica: DA

Vrsta grijanja: Klima

Godina izgradnje zgrade: 1998

Opis:

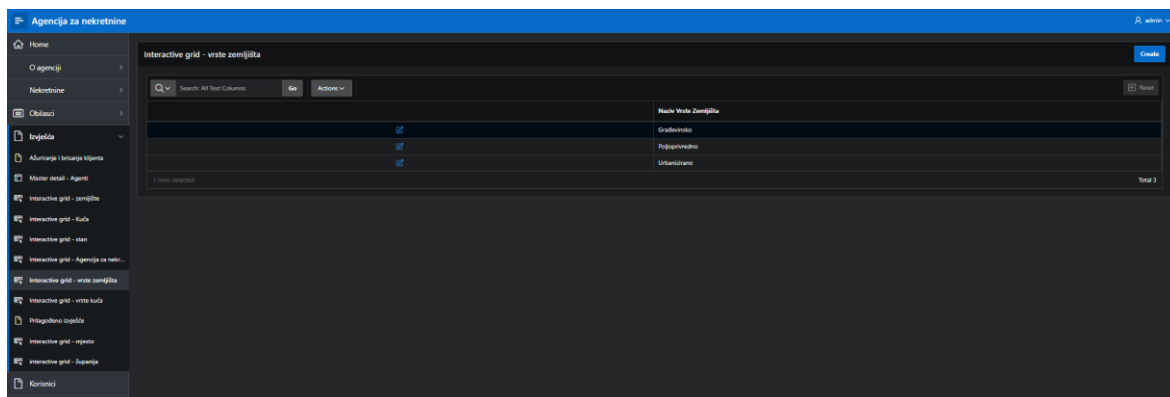
Cijena: 78000

Grad: Pula

Agencija: Fiume

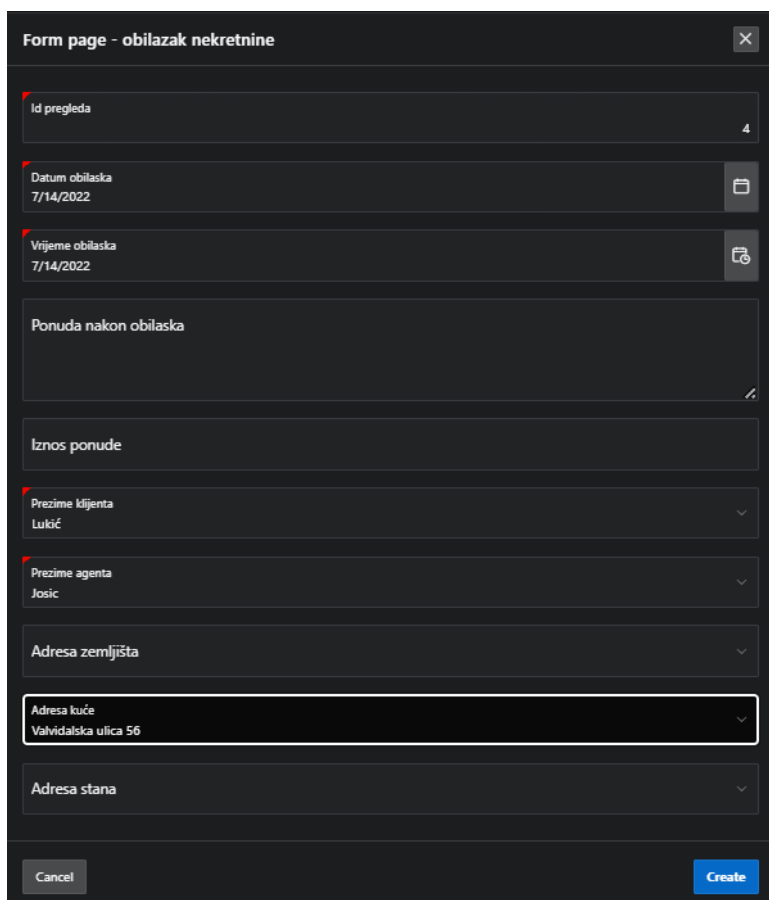
Slika 65: Prikaz obrasca za ažuriranje i brisanje podataka o stanu

Slijedeća fotografija prikazuje sve vrste zemljišta. Moguće je vršiti pretraživanje po nazivu zemljišta. Također, postoji gumb *Create* (gore desno) koji pruža mogućnost stvaranja novog zemljišta. Svako pojedinačno zemljište se može urediti. Ukoliko se odabere opcija uređivanja, onda se korisniku pružaju radnje poput ažuriranja ili brisanja podataka o pojedinoj vrsti zemljišta.



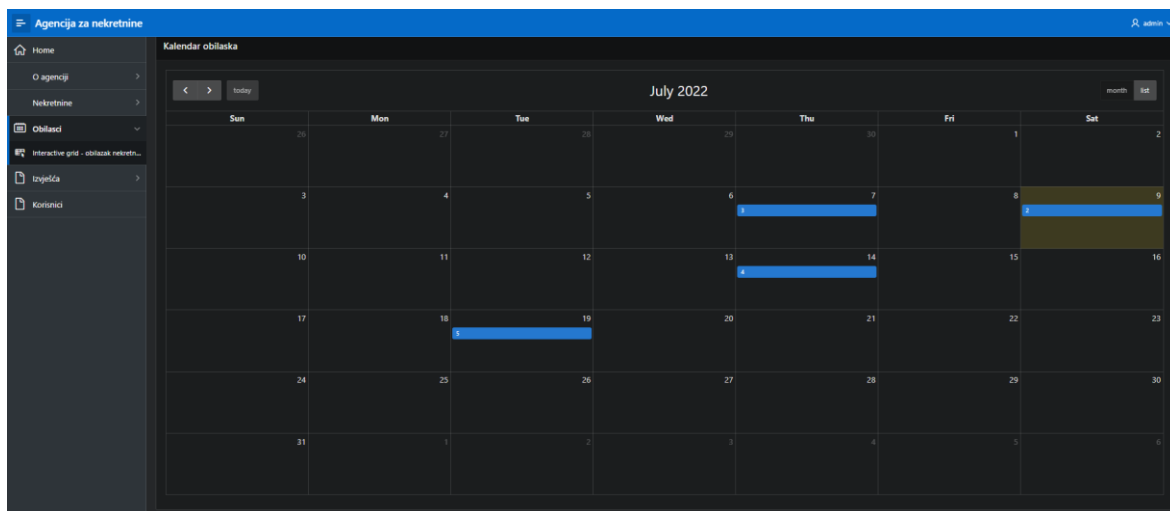
Slika 66: Prikaz mogućnosti dodavanja ili uređivanja odabrane vrste zemljišta

Na slici 67. prikazan je postupak kreiranja novog događaja za obilazak nekretnine. Naime, potrebno je ispuniti obavezna polja poput id-a pregleda, datuma i vremena obilaska te iz padajućih listi odabrati prezime klijenta koji želi obići nekretninu i prezime agenta koji vodi obilazak nekretnine.



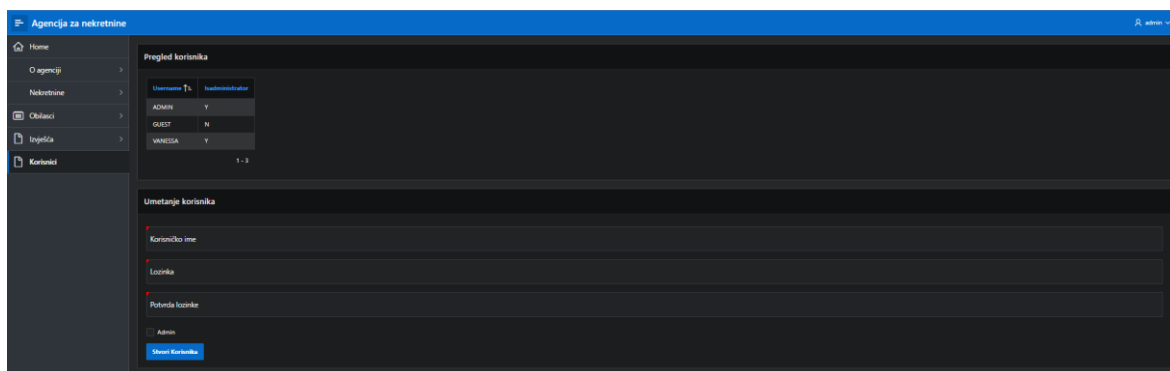
Slika 67: Prikaz umetanja obilaska nekretnine

Slijedeća fotografija prikazuje kalendar. Plava polja sadrže informaciju o id-u pregleda. Kalendar nudi dvije mogućnosti prikaza. Prva mogućnost prikaza je po mjesecima (prikazana na slici ispod), a druga mogućnost je lista koja sadrži uzlazno sortirane datume obilaska nekretnine.



Slika 68: Prikaz kalendara obilaska

Na slici 69. prikazana je tablica svih korisnika koji se trenutno nalaze u bazi podataka. Ukoliko korisnik unutar stupca *IsAdministrator* ima oznaku Y, onda korisnik ima ulogu administratora. Isto tako, oznaka N označava da korisnik nije administrator. Zbog sigurnosnih razloga nije moguć prikaz korisnikove lozinke (iako se ona ne može jednostavno pročitati zato što je spremljena u haširanom obliku). Unutar iste stranice moguće je unijeti novog korisnika na način da se popune obavezna polja (označena crvenom bojom) kao što su korisničko ime i lozinka. Zbog sigurnosti, potrebno je ponoviti unos lozinke. Prilikom kreiranja novog korisnika moguće je odabrati opciju *Admin* kako bi novostvoreni korisnik imao administratorske ovlasti unutar aplikacije. Na samom kraju, kako bi se potvrdilo stvaranje novog korisnika, potrebno je kliknuti na gumb *Stvori Korisnika*.



Slika 69: Prikaz pregleda i mogućnosti stvaranja korisnika

7. Zaključak

Krajnji rezultat ovog završnog rada je poslovna aplikacija nad relacijskom bazom podataka za agenciju za nekretnine izrađena koristeći alat Oracle APEX. Svaki alat ima svoje prednosti i nedostatke pa tako i ovaj. Neke od prednosti Oracle APEX-a jesu slijedeće: jednostavno postavljanje, dozvoljava timski rad, jednostavno povezivanje s bazom podataka, jednostavan za učenje, mala količina koda, brz razvoj aplikacije, skalabilnost i mogućnost pristupa velikog broja korisnika, fleksibilnost u prilagođavanju izgleda i funkcionalnosti aplikacije, nije potrebna licenca za rad... Neki od nedostataka korištenja Oracle APEX-a jesu: nemogućnost povezivanja s različitim bazama podataka (npr. My SQL), offline način korištenja. Alat Oracle APEX korišten je u mnogim industrijama kao što su: zdravstvena, automobilska, obrazovna, znanstvena, prijevozna... Mnogim korisnicima je bitno to što je alat besplatan, lako dostupan te brz i stabilan za korištenje. Stoga, nije čudno što je u posljednje vrijeme stekao veliku popularnost. Na samom kraju možemo zaključiti kako je Oracle APEX alat na visokoj razini koji nudi izradu kvalitetnih aplikacija koristeći relacijsku bazu podataka.

S obzirom da potražanja za softverom nikada nije bila veća, a zastarjele razvojne metodologije jednostavno ne mogu pratiti današnje trendove, došlo je do pojave brzog razvoja aplikacija koje omogućavaju RAD alati. Brzi razvoj aplikacija odnosi se na pristup razvoju softvera koji stavlja manji naglasak na planiranje, pa čak i specifikacije dizajna, a više na brzi razvoj i brzu povratnu informaciju. Prototipovi se brzo isporučuju korisnicima kako bi se osiguralo da su izravno uključeni u povratne informacije i razvoj. Koristeći metode koje su bile usmjerene na brzinu umjesto promišljanja, alati za brzi razvoj aplikacija (RAD) omogućili su razvojnim programerima način za brzo, učinkovito stvaranje softvera i bez potrebe za ulaganjem u ogromne proračune za razvoj.

U usporedbi s alatom Clarion s kojim sam se susrela na nekom drugom kolegiju, APEX nudi responzivan i moderniji dizajn. Također, u APEX-u je moguće ostvariti timsku suradnju, dok je u Clarion rad samostalan. Još jedna od prednosti APEX-a s obzirom na Clarion je mogućnost rada u online okruženju. S obzirom na probleme koje sam imala tijekom korištenja Clariona, točnije često rušenje programa, kod Oracle APEX-a takvih problema nije bilo. Jedino što sam primijetila u Oracle APEX-u je kratkotrajna stanka pri izradi Master Detail-a, no to nije stvorilo nikakve probleme tijekom izrade.

Tijekom izrade aplikacije u alatu Oracle APEX najviše mi se svidjela jednostavnost, intuitivnost i lakoća učenja. Uz to što alat nudi mnogobrojne funkcionalnosti, na internetu je moguće pronaći besplatne tutoriale kao i knjige. S obzirom da se skoro sve implementira korištenjem grafičkog sučelja, nije potrebno preveliko programsko znanje. Za mene osobno ogromna je prednost što se sve radi u online okruženju bez ikakvog preuzimanja na osobno računalo te mogu raditi bilo kada i bilo gdje. Moguće je uređivati neke dijelove dizajna, poput izgleda navigacije, te dodavati HTML kod zbog dodatnog uređivanja tekstualnog sadržaja.

Na samom kraju, zadovoljna sam cijelim tijekom izrade aplikacije u Oracle APEX alatu i postignutim rezultatom. Smatram kako postoji prilika za nadogradnju aplikacije kako bi uistinu mogla postati dio jednog poslovnog sustava agencije za nekretnine.

8. Popis slika

Slika 1: Prikaz grafičkih oblika konceptata strukture EV metode, preuzeto iz (Pavlič, 2011.)	8
Slika 2: Prikaz modela entiteti-veze	12
Slika 3: Prikaz mockup-a za početnu stranicu	15
Slika 4: Prikaz mockup-a za stranicu O agenciji	15
Slika 5: Prikaz mockup-a za stranicu Lokacija	16
Slika 6: Prikaz mockup-a za stranicu Obilasci	16
Slika 7: Prikaz mockup-a za stranicu Agenti	17
Slika 8: Prikaz mockup-a za stranicu Korisnici	17
Slika 9: Kreiranje radnog okruženja	18
Slika 10: Kreiranje aplikacije	19
Slika 11: Prikaz SQL upita za kreiranje tablice MJESTO	19
Slika 12: Kreiranje tablice MJESTO	20
Slika 13: Kreiranje tablice JE_ZAPOSLEN	20
Slika 14: Kreiranje tablice - primarni ključ agregacije	21
Slika 15: Kreiranje tablice - vanjski ključ agregacije	22
Slika 16: Ograničenja nad tablicom JE_ZAPOSLEN	22
Slika 17: Kreiranje Blank Page-a	23
Slika 18: Ubacivanje fotografije unutar aplikacije	23
Slika 19: Postavke regije za prikaz fotografije	24
Slika 20: Postavke regije za prikaz karte	24
Slika 21: Prikaz postavki za prikaz tablice AGENTI	24
Slika 22: Prikaz SQL upita za tablicu ŽUPANIJA	25
Slika 23: Prikaz SQL upita za prikaz mjesta s obzirom na odabranu županiju	25
Slika 24: Postavke kalendara na kartici Region	26
Slika 25: Postavke kalendara na kartici Attributes	26
Slika 26: Postavke nove regije	27
Slika 27: Hidden za prezime i Select list za id	27
Slika 28: Postavke za Id polje	28
Slika 29: Source za Id polje	28
Slika 30: Omogućava dozvola za uređivanje i brisanje unutar regije	29
Slika 31: Behavior na gumbu za brisanje	29
Slika 32: Behavior na gumbu za ažuriranje	29
Slika 33: Ubacivanje procesa PostaviNull	30
Slika 34: Postavke Source-a od procesa PostaviNull	30
Slika 35: Postavljanje procesa na gumb	30
Slika 36: Kreiranje Interactive Grid-a	31
Slika 37: Prikaz Master Detail izvješća za agregaciju JE_ZAPOSLEN	32
Slika 38: Kreiranje List Entry-a	32
Slika 39: Prikaz navigacije	33
Slika 40: Prikaz tablice USERS	33
Slika 41: Umetanje korisnika u bazu podataka	33
Slika 42: Prikaz podataka korisničkog imena i lozinke korisnika	33
Slika 43: Prikaz postavki regije za pregled korisnika	34

Slika 44: Unos novog korisnika u bazu podataka i haširanje lozinke.....	34
Slika 45: Stvaranje validacije	34
Slika 46: Validacija spremanja korisnikovog imena u obliku velikih slova	35
Slika 47: Validacija podudaranja lozinka.....	35
Slika 48: Stvaranje autentikacijske scheme.....	36
Slika 49: Prikaz autorizacijske scheme za goste	36
Slika 50: Prikaz autorizacijske scheme za administratore	37
Slika 51: Primjena autorizacijske sheme na razinu cijele aplikacije	37
Slika 52: Postavljanje stranice na javno.....	37
Slika 53: Prikaz početne stranice	38
Slika 54: Prikaz informacija o agenciji	38
Slika 55: Prikaz login stranice	39
Slika 56: Prikaz informacija o zaposlenicima	39
Slika 57: Prikaz umetanja županije.....	39
Slika 58: Prikaz umetanja mjesta.....	40
Slika 59: Prikaz filtriranja mjesta s obzirom na županiju	40
Slika 60: Prikaz obrasca za ažuriranje i brisanje određenog klijenta.....	40
Slika 61: Prikaz obrasca za umetanje kuće	41
Slika 62: Prikaz obrasca za uređivanje i brisanje kuća	42
Slika 63: Prikaz popisa svih kuća	42
Slika 64: Prikaz prilagođenog izvješća za tablicu KUĆA	43
Slika 65; Prikaz obrasca za ažuriranje i brisanje podataka o stanu	43
Slika 66: Prikaz mogućnosti dodavanja ili uređivanja odabrane vrste zemljišta	44
Slika 67: Prikaz umetanja obilaska nekretnine.....	44
Slika 68: Prikaz kalendara obilaska	45
Slika 69: Prikaz pregleda i mogućnosti stvaranja korisnika	45

9. Literatura

Čandrlić, S. (n.d.). Dizajn korisničkog sučelja i iskustva - wireframe. Rijeka, Hrvatska.

Kissflow. (25. Studeni 2020.). (Kissflow, Producent, & Kissflow) Preuzeto 5. Srpanj 2022. iz Kissflow: <https://kissflow.com/application-development/rad/rapid-application-development-tools/>

mendix. (n.d.). Preuzeto 5. Srpanj 2022. iz mendix: <https://www.mendix.com/low-code-guide/>

Oracle APEX. (n.d.). Preuzeto 10. Lipanj 2022. iz Oracle APEX: <https://apex.oracle.com/en/learn/documentation/>

Pavlić, M. (2011.). *Oblikovanje baza podataka*. Rijeka: Odjel za informatiku Sveučilišta u Rijeci.

Sciore, E. (2020.). *Understanding Oracle APEX 20 Application Development*. apress.

Wikipedia. (n.d.). Preuzeto 5. Srpanj 2022. iz Wikipedia: https://en.wikipedia.org/wiki/Low-code_development_platform

10. Popis priloga

https://apex.oracle.com/pls/apex/r/agencija_za_nekretnine/agencija-za-nekretnine/home?session=0