

Restorani - izrada poslovne aplikacije nad relacijskom bazom podataka u Oracle APEX 22.1.1

Kopričanec, Ivana

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://um.nsk.hr/um:nbn:hr:195:508691>

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

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



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

Ivana Kopričanec

Restorani – izrada poslovne aplikacije
nad relacijskom bazom podataka u
Oracle APEX 22.1.1

Završni rad

Mentor: prof. dr. sc. Patrizia Pošćić

Rijeka, srpanj 2022.

Rijeka, 30. lipnja 2022.

Zadatak za završni rad

Pristupnik/ca: Ivana Kopričanec

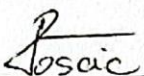
Naziv završnog rada: Restorani - poslovna aplikacija nad relacijskom bazom podataka (Oracle APEX)

Naziv završnog rada na eng. jeziku: Restaurants - 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 poslovanja nekog restorana. Aplikacija će biti izgrađena nad relacijskom BP u alatu Oracle APEX. Od studentice se očekuje da kreira model podataka i odgovarajuću poslovnu aplikaciju.

Mentor

Prof. dr. sc. Patrizia Pošćić



Voditelj za završne radove

Doc. dr. sc. Miran Pobar



Zadatak preuzet: 30. lipnja 2022.



(potpis pristupnika)

Sažetak

Unutar ovog rada prikazan je i opisan proces nastajanja poslovne aplikacije „Restorani“ za čiju je izradu korišten Oracle Application Express (Oracle APEX) alat. Prije same izrade aplikacije predstavljen je dijagram entiteti-veze te pripadajući relacijski model koji odražavaju poslovanje restorana. Za izradu dijagrama entiteta i veza korišten je web alat Figma dok je za izradu aplikacije korištena verzija Oracle APEX-a 22.1.1. Cilj rada je predstaviti korištenje Oracle APEX alata na temelju vlastitog primjera, gdje je svako objašnjenje popraćeno slikovnim prikazima ekrana. U uvodu je detaljnije opisan alat Oracle APEX te metodologija prema kojoj je nastao model entiteti-veze dok zaključak sadržava cjelokupni sažetak odrađenih zadataka u procesu razvoja aplikacije.

Ključne riječi

Aplikacija, Oracle, APEX, Baza podataka, DEV, Entitet, Veza, Atribut, Agregacija, Primarni ključ, Vanjski ključ, Restoran

Sadržaj

Sadržaj.....	4
1. Uvod.....	5
2. Model entiteti-veze.....	6
3. Relacijski model.....	8
4. Izrada aplikacije u Oracle APEX 22.1.1	9
4.1 Kreiranje radnog okruženja.....	9
4.2 Kreiranje tablica	10
4.3 Kreiranje aplikacije	14
4.3.1 Blank Page stranice	15
4.3.2 Interactive Grid izvješće.....	21
4.3.3 Master Detail izvješće	24
4.3.4 Izračuni (Trigger i Computation)	26
4.4 Izrada navigacije	27
4.5 Autentifikacija i autorizacija	28
5. Prikaz izrađene aplikacije.....	36
6. Zaključak.....	43
7. Literatura	44
8. Popis slika	45
9. Popis priloga.....	47

1. Uvod

Uz napredovanje tehnologije i digitalizaciju gotovo svih aspekata ljudskih života, važnu ulogu igra razvoj informacijskih sustava i aplikacija čiji je cilj olakšati poslovanje, smanjiti trošak održavanja i najbitnije, djelatnicima i korisnicima pružiti okolinu koja je ugodna i jednostavna za rad. U prošlosti je vođenje poslova i upravljanje poslovnim sustavima bilo mnogo kompliciranije i zamršenije iz razloga što nije postojao luksuz računala i sustava koji bi preuzeli teret provođenja izračuna, pohrane podataka, komunikacije sa raznim servisima, automatizacije svega prethodno navedenog i naposljetku, integracije u jedan cjeloviti sustav. Kako bi se riješio taj problem, najveći doprinos dala su računala odnosno razvoj informacijskih sustava. Što se tiče poslovanja kompanija i tvrtki, uz sam proces proizvodnje i prodaje dobara, jednako veliku ulogu igra i način na koji se posao obavlja. Kako bi se smanjilo „trenje“ po pitanju pohrane podataka u poslovanju te upravljanju ulaznim i izlaznim podacima, osmišljene su računalne aplikacije čiji je cilj upravo taj, uklanjanje „trenja“. Takve aplikacije sastoje se primarno od produkcijske baze podataka, u kojoj su pohranjeni svi podaci o svakom detalju koji je vezan uz poslovanje kompanije, i aplikacije koja vrši razne manipulacije tim podacima te prati nastale izmjene.

Što se tiče baza podataka i razvoja softvera, jedna od većih korporacija koja prva pada na pamet jest Oracle Corporation. Osnovana 16. lipnja 1977. godine u gradu Santa Clara, Kalifornija nekoć je bila najjača softverska kompanija na svijetu po prihodima i dosegu tržišta. (Oracle Corporation 2022.)

Jedan od poznatijih proizvoda Oracle korporacije jest Oracle Application Express, poznatiji kao APEX. APEX je razvojna platforma bazirana na low-code pristupu čiji je cilj smanjiti potrebnu razinu znanja u području programiranja pri razvoju aplikacija. Korisniku je omogućen razvoj sigurnih, skalabilnih i vrlo fleksibilnih programskih rješenja koja su spremna za korištenje u stvarnom poslovanju. (APEX n.d.)

Kao predznanje koje bi bilo poželjno kod razvijaa, važno je istaknuti poznavanje načina organizacije podatkovnog sustava kao i njegovo međudjelovanje. Prije samog razvijanja aplikacije u Apexu potrebno je osmisliti model poslovanja za što je korišten dijagram entiteta i veza prema metodici MIRIS. „*Specijalizirana metodologija MIRIS (skraćeno od hrvatskog Metodologija za Razvoj Informacijskog Sustava) je skup metoda i uputa čiji je ukupni cilj projektirati i izgraditi informacijski sustav. Poznata je i kao metodika MIRIS. Ta specijalizirana metodologija propisuje faze razvoja i aktivnosti pojedine faze do potrebne razine detalja informacijskih sustava.*“ (Pavlič 2011.)

Ova metodika zasniva se na postojanju metode EV (entitet-veza) čije su sastavnice upravo razne vrste entiteta od kojih su korišteni jaki tip, slabi tip, agregacija, veze, brojnosti i atributi. Entitet možemo definirati kao pojam u poslovanju o kome se zahtijeva čuvanje nekih podataka. (Pavlič 2011.) Prvi i najšire korišten entitet, odnosno tip entiteta je jaki tip entiteta, a njemu je svojstveno da ima vlastiti primarni ključ i nije ovisan o drugim tipovima entiteta. Za razliku od jakog tipa, slabi tip entiteta je ovisan o jakom tipu entiteta na kojeg je povezan. Kada se govori o složenijim odnosima između entiteta moguće je da se dogodi pojava apstrakcije u kojoj se tip veze između dvaju ili više entiteta tretira kao novi tip entiteta. Takav tip se naziva agregacija. Uz tipove entiteta vežu se pojmovi atributa koje možemo definirati kao svojstva tipa entiteta kojemu pripada. Veze koje se nalaze između tipova entiteta su u većini slučajeva obilježene brojnostima. *Brojnost tipa veze je broj (upisuje se eksplicitno na DEV) koji kaže koliko entiteta pojedinoga tipa entiteta E1 sudjeluje (pojavljuje se) u tipu veze V s entitetom iz tipa entiteta E.* (Pavlič 2011.)

2. Model entiteti-veze

Realizacija aplikacije započinje osmišljavanjem i stvaranjem modela entiteti-veze, odnosno DEV-a (dijagrama entiteta i veza) što predstavlja grafički prikaz modela podataka sustava. U nastavku slijedi opis izrađenog modela entiteta i veza za aplikaciju Restorani, prikazanog na slici 1.

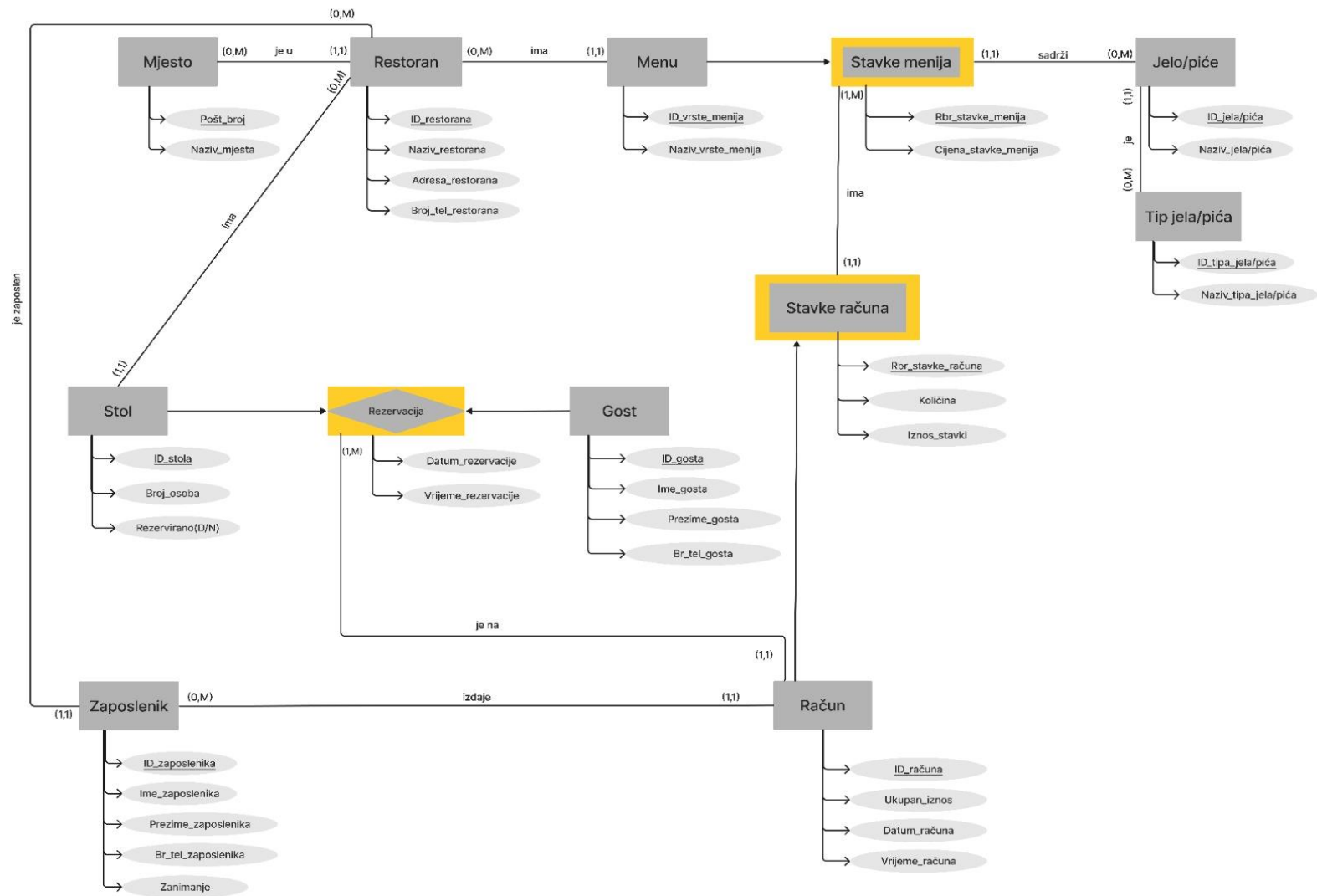
Model sadrži ukupno dvanaest tipova entiteta od čega je devet jakih tipova entiteta, dva slaba tipa entiteta te jedna agregacija. Jaki tipovi entiteta koji se pojavljuju jesu: MJESTO, RESTORAN, MENU, JELO/PIĆE, TIP JELA/PIĆA, STOL, GOST, ZAPOSLENIK i RAČUN. Nadalje, slabi tipovi entiteta koje model sadrži jesu STAVKE MENIJA i STAVKE RAČUNA, dok agregaciju predstavlja tip entiteta REZERVACIJA.

Svaki od tipova entiteta posjeduje primarni ključ koji predstavlja jedinstveni identifikator tog tipa entiteta te pripadajuće atribute. Jaki tipovi entiteta imaju zaseban, vlastiti primarni ključ, primarni ključ slabog tipa entiteta čini kombinacija primarnih ključeva jakog i slabog tipa entiteta te kako agregacija nastaje ako veza između dva tipa entiteta ima brojnosti (0,M):(0,M), njezin primarni ključ je kombinacija primarnih ključeva tipova entiteta koji je tvore. Na dijagramu entiteta i veza primarni ključevi podcrtani su punom linijom. U nastavku, kao primjer, ukratko je opisan jedan jaki tip entiteta, jedan slabi tip entiteta te agregacija.

Jaki tip entiteta MJESTO sastoji se od atributa *Pošt_broj* te *Naziv_mjesta*. Atribut *Pošt_broj* ima funkciju primarnog ključa ovog tipa entiteta zbog toga što svako mjesto ima različit poštanski broj, odnosno jedan, konkretan poštanski broj može imati maksimalno jedno mjesto.

Slabi tip entiteta STAVKE MENIJA sadrži atribute *ID_vrste_menija*, *Rbr_stavke_menija* i *Cijena_stavke_menija*, od čega *ID_vrste_menija* te *Rbr_stavke_menija* čine primarni ključ. Razlog zbog kojeg je *ID_vrste_menija* dio primarnog ključa ovog tipa entiteta je taj što bez jakog tipa entiteta MENU, slabi tip entiteta STAVKE MENIJA ne bi postojao. Tip entiteta STAVKE MENIJA je ovisan o jakom tipu entiteta MENU te ukoliko neki konkretan menu više ne postoji, ne bi smjele postojati niti stavke tog menija.

Agregacija REZERVACIJA ima atribute *ID_stola*, *ID_gosta*, *Datum_rezervacije* i *Vrijeme_rezervacije*. Ovu agregaciju čine dva jaka tipa entiteta STOL i GOST, a nastala je jer je veza između ova dva jaka tipa entiteta brojnosti (0,M):(0,M). Sukladno tome, agregacija poprima kao primarni ključ kombinaciju primarnih ključeva dva jaka tipa entiteta koja je čine, *ID_stola* te *ID_gosta*.



Slika 1: Dijagram entiteti-veze poslovanja restorana

3. Relacijski model

Na temelju izrađenog modela entiteta i veza te pravila prevođenja, nastao je relacijski model prikaza u nastavku. Prethodno spomenuti primarni ključevi podcrtani su punom linijom dok su vanjski ključevi, ukoliko postoje, navedeni na kraju, ukošenim slovima. Općenito, vanjski ključ predstavlja poveznicu između dvije tablice. Nastaje ukoliko brojnost veze na promatranom tipu entiteta iznosi (1,1) dok na tipu entiteta s kojim je povezan brojnost iznosi (0,M). Tada vanjski ključ na strani promatranog entiteta poprima vrijednost primarnog ključa entiteta sa čije je strane brojnost (0,M).

MJESTO (Pošt_broj, Naziv_mjesta)

RESTORAN (ID_restorana, Naziv_restorana, Adresa_restorana, Broj_tel_restorana, *Pošt_broj*)

MENU (ID_vrste_menija, Naziv_vrste_menija, *ID_restorana*)

TIP JELA/PIĆA (ID_tipa_jela/pića, Naziv_tipa_jela/pića)

JELO/PIĆE (ID_jela/pića, Naziv_jela/pića, *ID_tipa_jela/pića*)

STOL (ID_stola, Broj_osoba, Rezervirano(D/N), *ID_restorana*)

GOST (ID_gosta, Ime_gosta, Prezime_gosta, Br_tel_gosta)

ZAPOSLENIK (ID_zaposlenika, Ime_zaposlenika, Prezime_zaposlenika, Br_tel_zaposlenika, Zanimanje, *ID_restorana*)

RAČUN (ID_računa, Ukupan_iznos, Datum_računa, Vrijeme_računa, *ID_stola*, *ID_gosta*, *ID_zaposlenika*)

STAVKE MENIJA (ID_vrste_menija, *Rbr_stavke_menija*, Cijena_stavke_menija, *ID_jela/pića*)

REZERVACIJA (ID_stola, ID_gosta, Datum_rezervacije, Vrijeme_rezervacije)

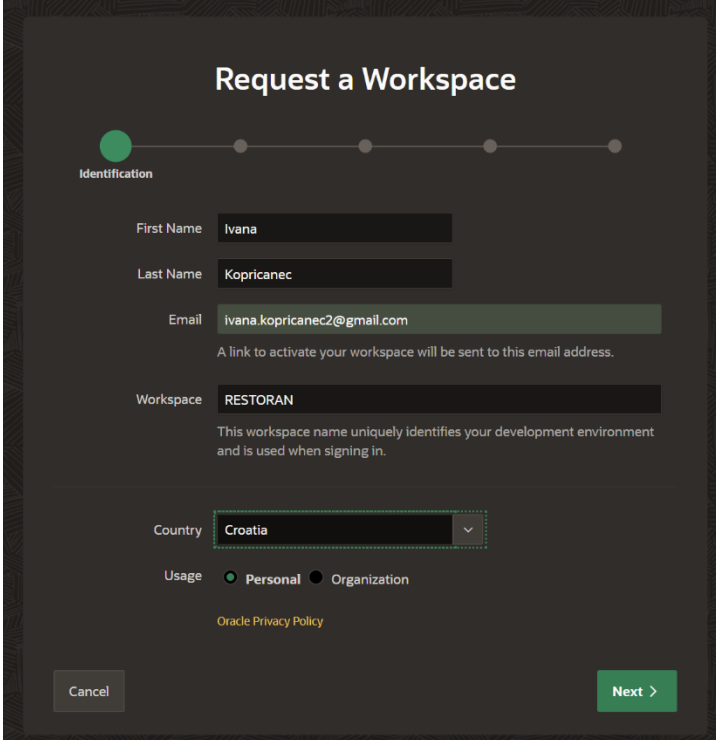
STAVKE RAČUNA (ID_računa, *Rbr_stavke_računa*, Količina, Iznos_stavki, *ID_vrste_menija*, *Rbr_stavke_menija*)

4. Izrada aplikacije u Oracle APEX 22.1.1

U idućem dijelu rada slijedi opis izrade same aplikacije koristeći Oracle APEX.

4.1 Kreiranje radnog okruženja

Prilikom izrade aplikacije u Oracle APEX-u prvo je potrebno na službenoj internet stranici Oracle APEX zatražiti kreiranje besplatnog radnog okruženja klikom na opciju *Request a Workspace*. Na slici 2 prikazan je ispunjen obrazac, zahtjev za kreiranje radnog okruženja te nakon popunjavanja svih traženih podataka, potrebno je podnijeti zahtjev odabirom opcije *Submit Request*.



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.

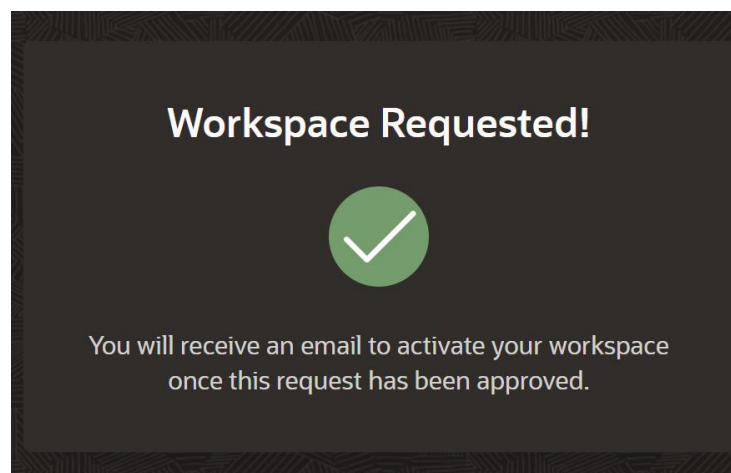
Country

Usage Personal Organization

[Oracle Privacy Policy](#)

Slika 2: Zahtjev za kreiranje radnog okruženja

Ukoliko je zahtjev uspješno podnesen, korisnik na e-mail dobije obavijest o odobrenju te klikom na poveznicu unutar e-maila dovršeno je kreiranje radnog okruženja što prikazuje slika 3.



Slika 3: Uspješno kreiranje radnog okruženja

4.2 Kreiranje tablica

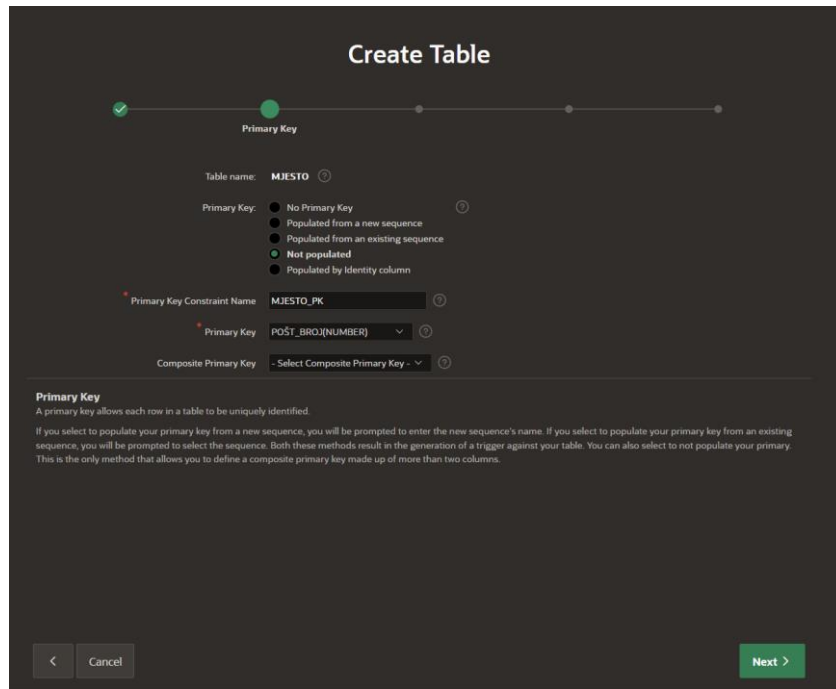
Idući korak u izradi aplikacije bio je kreiranje baze podataka, odnosno kreiranje tablica prema izrađenom relacijskom modelu. Tablice su kreirane uz pomoć opcije *Create Object – Table* unutar Oracle APEX sučelja na način da su prvo uključene tablice koje nemaju vanjskih ključeva i koje su na modelu entiteta i veza prikazani kao jaki tipovi entiteta. Potom su izrađene one tablice koje sadrže vanjske ključeve. Slika 4 prikazuje primjer kreiranja tablice MJESTO koja ne sadrži vanjski ključ.

Column Name	Type	Precision	Scale	Not Null	Identity	Move
POŠT_BROJ	NUMBER	5	0	<input checked="" type="checkbox"/>	None	▲ ▼
NAZIV_MJESTA	VARCHAR2		30	<input checked="" type="checkbox"/>		▲ ▼
	- Select Datatype -					▲ ▼
	- Select Datatype -					▲ ▼
	- Select Datatype -					▲ ▼

Slika 4: Kreiranje tablice MJESTO

Na slici je vidljivo kako je potrebno upisati sve attribute koje MJESTO sadrži, odrediti tip podataka, maksimalnu duljinu podataka, a ukoliko je riječ o tipu *NUMBER* i duljinu decimalnih mjesta te odrediti smije li podatak poprimiti *Null* vrijednost. U ovom slučaju, atribut *Pošt_broj* je tipa *NUMBER*, duljina mu je 5, a duljina decimalnih mjesta jednaka je 0. Također, podatak ne smije poprimiti *Null* vrijednost jer će imati ulogu primarnog ključa. Atribut *Naziv_mjesta* je tipa *VARCHAR2*, maksimalna duljina je 30 i podatak ne smije poprimiti *Null* vrijednost.

Ono što preostaje kod kreiranja tablice MJESTO, zbog toga što nema vanjskih ključeva, jest postavljanje primarnog ključa, što je prikazano slikom 5.



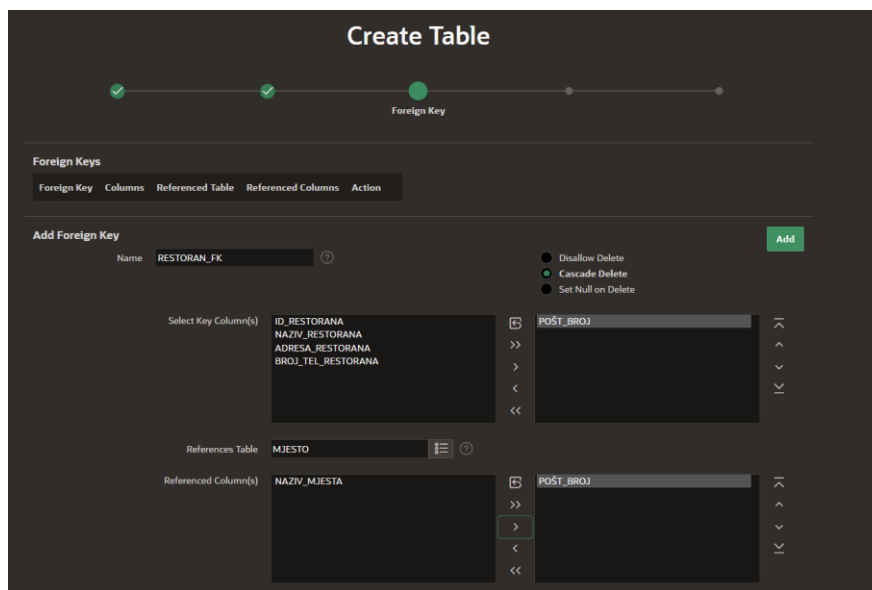
Slika 5: Kreiranje primarnog ključa tablice MJESTO

Nakon postavljanja primarnog ključa moguće je odmah otići na opciju *Create Table* nakon čega tablica postaje vidljiva u Oracle APEX sučelju (slika 6).

Column Name	Data Type	Nullable	Default	Primary Key
POŠT_BROJ	NUMBER(5,0)	No	-	1
NAZIV_MJESTA	VARCHAR2(30)	No	-	-

Slika 6: Tablica MJESTO

Ukoliko tablica ima vanjski ključ, potrebno je prilikom kreiranja tablice izvršiti jedan dodatan korak. Za primjer je uzeta tablica RESTORAN koja kao vanjski ključ sadrži atribut *Pošć_broj* iz tablice MJESTO. U procesu uvrštavanja podataka treba pripaziti da atribut *Pošć_broj* ima iste parametre kao i u roditeljskoj tablici. Nadalje, nakon postavljanja primarnog ključa kako je već opisano ranije, potrebno je postaviti atribut *Pošć_broj* kao vanjski ključ. Slika 7 prikazuje stvaranje vanjskog ključa tablice RESTORAN.



Slika 7: Vanjski ključ tablice RESTORAN

Iz atributa tablice RESTORAN potrebno je izdvojiti atribut *Pošt_broj*, u polje *References Table* odabrati roditeljsku tablicu MESTO te, također, izdvojiti isti atribut *Pošt_broj*. Opcija *Cascade Delete* služi brisanju podataka u podređenoj tablici ukoliko je zapis izbrisan iz nadređene tablice, što bi u ovom slučaju značilo da ako se obriše određeno mjesto, automatski će se obrisati i restoran iz tog mjesta. Nakon opisanog koraka potrebno je kliknuti ociju *Add* kako bi se uvedene promjene pohranile i potom kreirati tablicu RESTORAN.

Izrada tablice za agregaciju ili slabi tip entiteta je vrlo slična, razlika je u načinu stvaranja primarnog ključa što je prikazano kroz primjer tablice REZERVACIJA. Naime, agregacija REZERVACIJA poprima kao primarni ključ primarne ključeve entiteta koji čine tu agregaciju, dakle *ID_stola* te *ID_gosta*. Te attribute potrebno je uključiti u tablicu pazeći da se naziv i tip podataka podudaraju sa nazivom i tipom podataka u roditeljskoj tablici (slika 8).

Create Table

Columns

Table Name: REZERVACIJA

Preserve Case

Column Name	Type	Precision	Scale	Not Null	Identity	Move
ID_stola	NUMBER	5	0	<input checked="" type="checkbox"/>	- None -	^ v
ID_gosta	NUMBER	5	0	<input checked="" type="checkbox"/>	- None -	^ v
Datum_rezervacije	DATE			<input checked="" type="checkbox"/>		^ v
Vrijeme_rezervacije	TIMESTAMP		3	<input checked="" type="checkbox"/>		^ v
	- Select Datatype -					^ v
	- Select Datatype -					^ v
	- Select Datatype -					^ v
	- Select Datatype -					^ v

Add Column

Cancel Next >

Slika 8: Atributi tablice REZERVACIJA

Prilikom određivanja primarnog ključa, za razliku od jakih tipova entiteta, potrebno je odabrati opciju *Populated by Identity column*. Nakon toga odabiru se oba atributa koja čine primarni ključ kako je prikazano na slici 9.

Create Table

Primary Key

Table name: REZERVACIJA

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: REZERVACIJA_PK

Primary Key: ID_STOLA(NUMBER)

Composite Primary Key: ID_GOSTA(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.

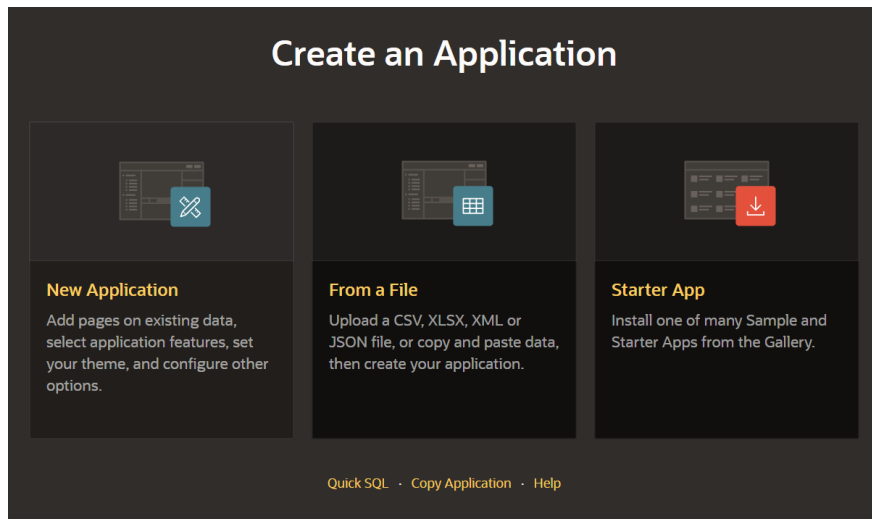
< Cancel Next >

Slika 9: Kreiranje primarnog ključa tablice REZERVACIJA

Osim postavljanja atributa *ID_stola* i *ID_gosta* kao primarni ključ potrebno ih je postaviti i kao vanjski ključ, kako je već opisano ranije u radu, jer se povlače podaci iz postojećih tablica STOL i GOST. Na taj se način kreiraju tablice za slabi tip entiteta i agregaciju.

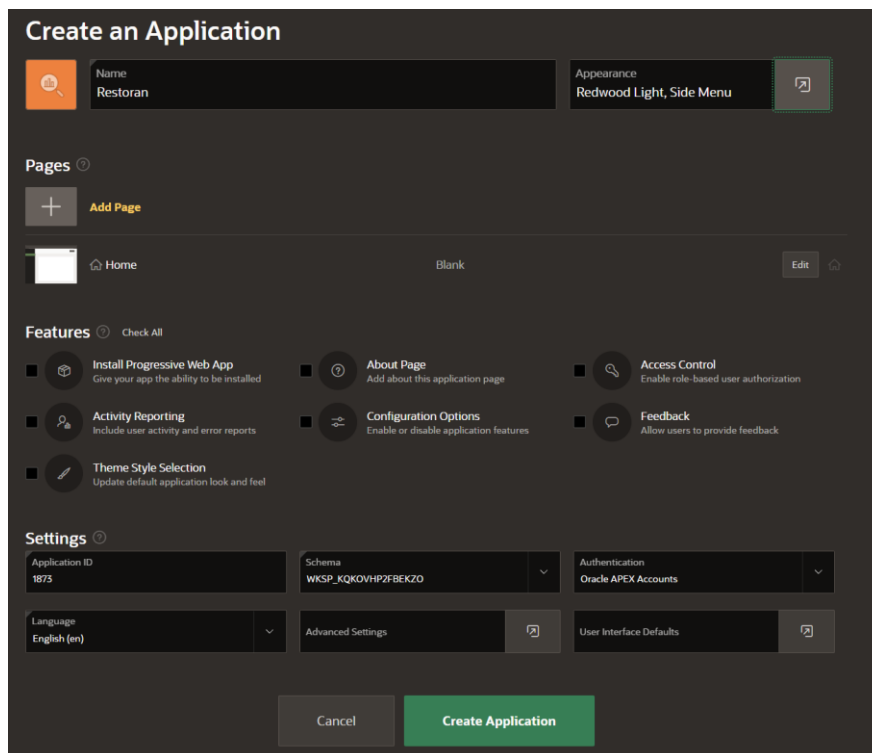
4.3 Kreiranje aplikacije

Nakon kreiranja baze podataka slijedilo je kreiranje same aplikacije Restoran. Za izradu aplikacije korištena je opcija *Create* → *New Application* koja se nalazi unutar opcije *App Builder* u sklopu Oracle APEX sučelja (slika 10).

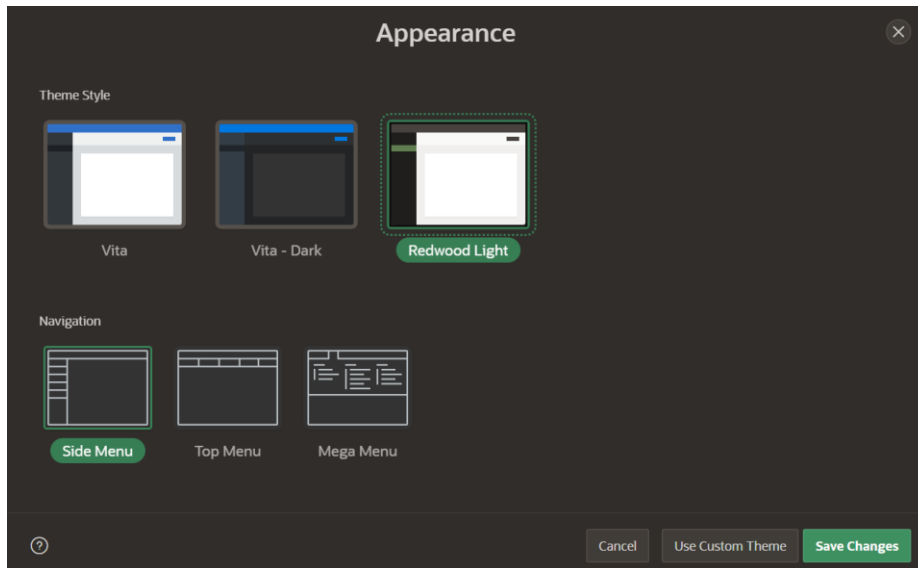


Slika 10: Kreiranje aplikacije

Klikom na *New Application* otvara se novi prozor gdje se definira ime aplikacije, nude opcije za podešavanje izgleda aplikacije (odabir teme i vrste navigacije) te prikazuju dodatne značajke poput informacija o aplikaciji, konfiguracije, kontrole pristupa i slično (slika 11). Za izgled, odnosno vizualni dojam odabrana je tema *Redwood Light* dok je za navigaciju odabrana opcija *Side Menu* kako bi se glavni izbornik nalazio sa lijeve strane (slika 12).



Slika 11: Kreiranje aplikacije Restoran



Slika 12: Odabir izgleda aplikacije

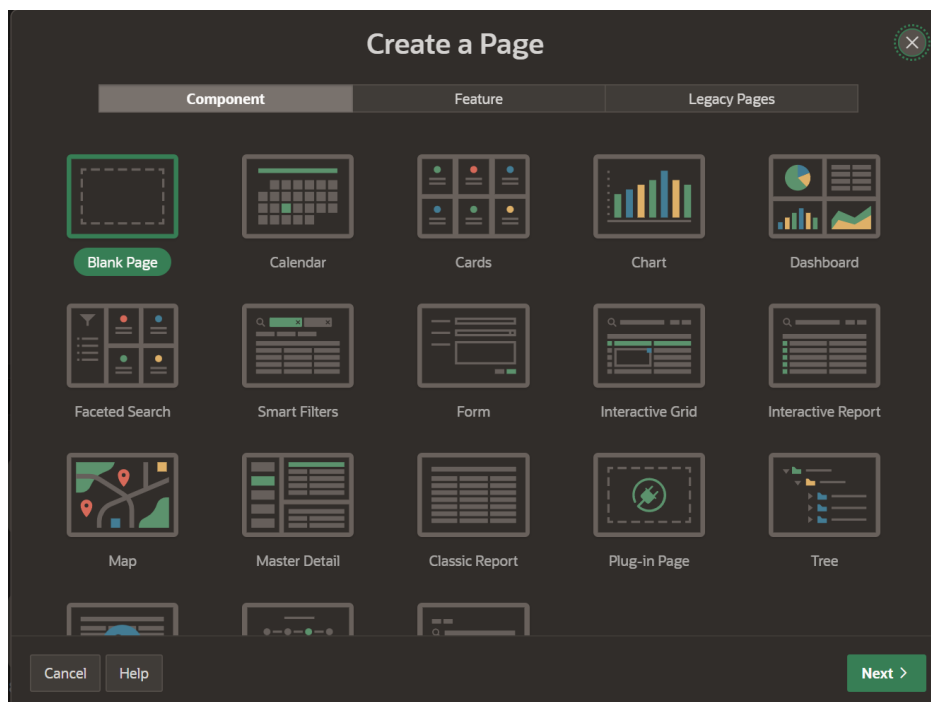
Klikom na gumb *Create Application* stvorena je aplikacija Restoran te se prikazuje sučelje za upravljanje samom aplikacijom.

4.3.1 Blank Page stranice

Kako bi se u aplikaciji mogao pregledavati sadržaj, potrebno je uključiti odgovarajuće tablice. Za prikaz tablica korišten je tip *Blank Page* unutar kojeg se pomoću regija smješta sadržaj za pregled.

4.3.1.1 Pregled sadržaja

Na primjeru stranice *Jelovnik* opisan je postupak kreiranja i podešavanja postavki *Blank Page*-a. Prvi korak pri stvaranju stranice za pregled je kreiranje nove stranice tipa *Blank Page* (slika 13) te njezino imenovanje i postavljanje navigacije (slika 14).

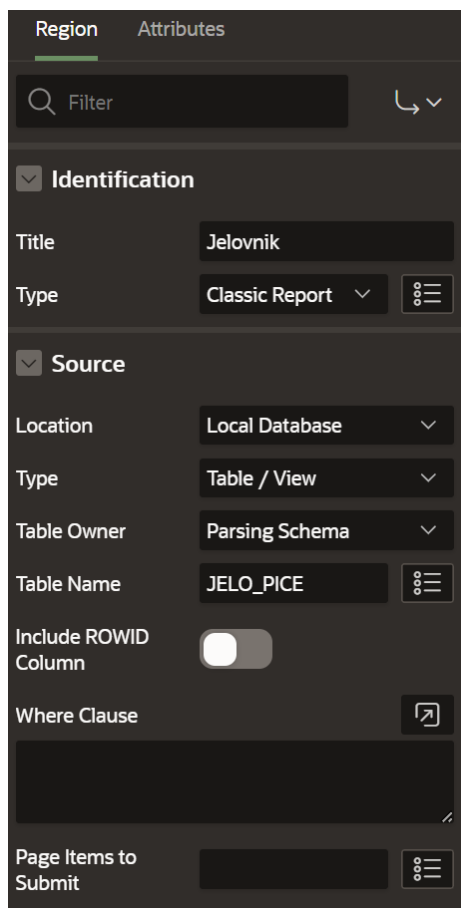


Slika 13: Kreiranje stranice tipa *Blank Page*

Slika 14: Postavke stranice tipa *Blank Page*

U odjeljku za postavljanje navigacije, kao vrijednost polja *Parent Navigation Menu Entry* postavljena je druga stranica tipa *Blank Page*, *Menu*. Razlog tome je što će se na taj način postići da unutar aplikacije, u navigaciji, stranica *Jelovnik* bude podređena stranici *Menu*. Odnosno, stranica *Jelovnik* će se nalaziti unutar padajućeg izbornika stranice *Menu*.

Nakon kreiranja same stranice slijedi izrada regije unutar koje će se prikazivati željena tablica sa sadržajem. To se postiže u sučelju izrađene stranice gdje se u lijevom izborniku unutar kategorije *Components* -> *Body* desnim klikom miša odabire opcija *Create Region*. Odabirom novonastale regije pojavljuje se izbornik na desnoj strani sučelja sa mogućim postavkama regije. Najvažniji korak je odabir tipa regije unutar kategorije *Identification* koji je postavljen na *Classic Report*, format prikaza tabličnih podataka, te odabir tablice unutar kategorije *Source* čiji podaci će se prikazivati (slika 15). U ovom slučaju to je tablica JELO_PÍĆE.

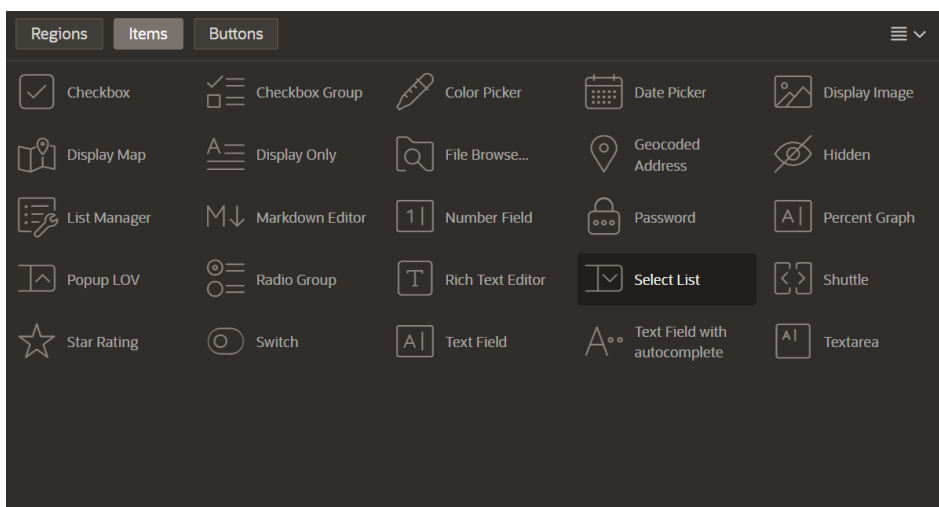


Slika 15: Postavke regije Jelovnik

Klikom na gumb Save u gornjem desnom kutu sučelja, završena je izrada prikaza podataka određene tablice.

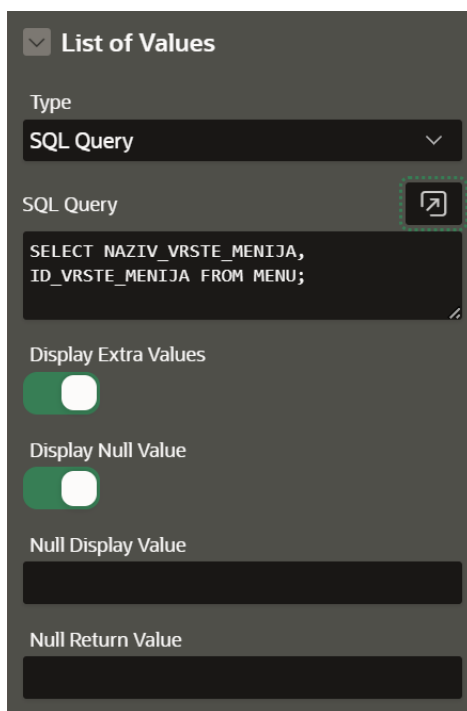
4.3.1.2 Filtriranje i pregled sadržaja

Kako bi se uz pregled podaci mogli i filtrirati, na primjeru tablica MENU te STAVKE_MENIJA, korištena je padajuća lista (*engl. Select List*) koja je postupkom drag and drop ubačena na stranicu iz prozora za odabir *Item-a* (*engl. Item Gallery*).



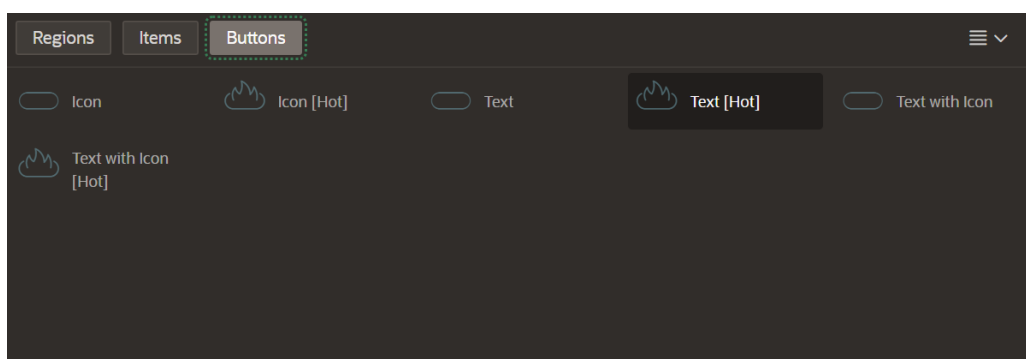
Slika 16: Padajuća lista iz prozora Item Gallery

Padajuća lista smještena je unutar prethodno kreirane regije (*Components -> Body -> Create Region*) tipa *Static Content*. Kako bi se unutar padajuće liste nudili željeni rezultati, u ovom slučaju nazivi vrste menija, potrebno je sastaviti odgovarajući SQL upit i upisati ga unutar kategorije *List of Values -> SQL Query* u izborniku sa desne strane (slika 17).

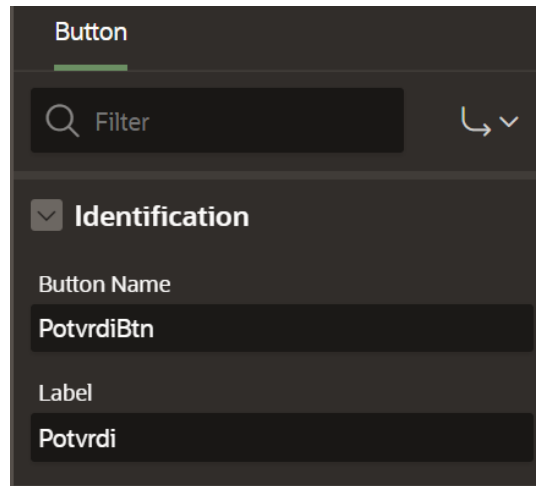


Slika 17: SQL upit za padajuću listu *Vrste menija*

Potrebno je, također, ubaciti i gumb na čiji će se klik pretražiti sadržaj. To se ostvaruje na isti način kao i dodavanje padajuće liste jedino što se postupak drag and drop vrši iz prozora za odabir gumba (*engl. Button Gallery*), prikazano na slici 18. Gumb se smješta unutar iste regije kao i spomenuta padajuća lista te mu je izmijenjeno ime u *Potvrdi* unutar kategorije *Identification* u izborniku sa desne strane (slika 19).



Slika 18: Gumb za filtriranje vrsta menija



Slika 19: Postavke gumba za filtriranje vrsta menija

Posljednji korak je izrada nove regije unutar kojeg će se nalaziti tablica s filtriranim rezultatima. Kako se prema odabiru željene vrste menija trebaju prikazati rezultati sa stavkama koje taj menu sadrži, potrebno je u regiju s rezultatima uključiti tablicu STAVKE_MENIJA. Nova regija treba biti tipa *Classic Report* te je unutar *Source* dijela u desnom izborniku potrebno sastaviti SQL upit za prikazivanje filtriranih rezultata (slika 20). Unutar upita, :P4_NEW predstavlja opciju odabranu u padajućoj listi te se na taj način ostvaruje filtriranje, odnosno provjerava se koje stavke imaju *id_vrste_menija* isti kao *id_vrste_menija* odabranog menija iz padajuće liste.

```

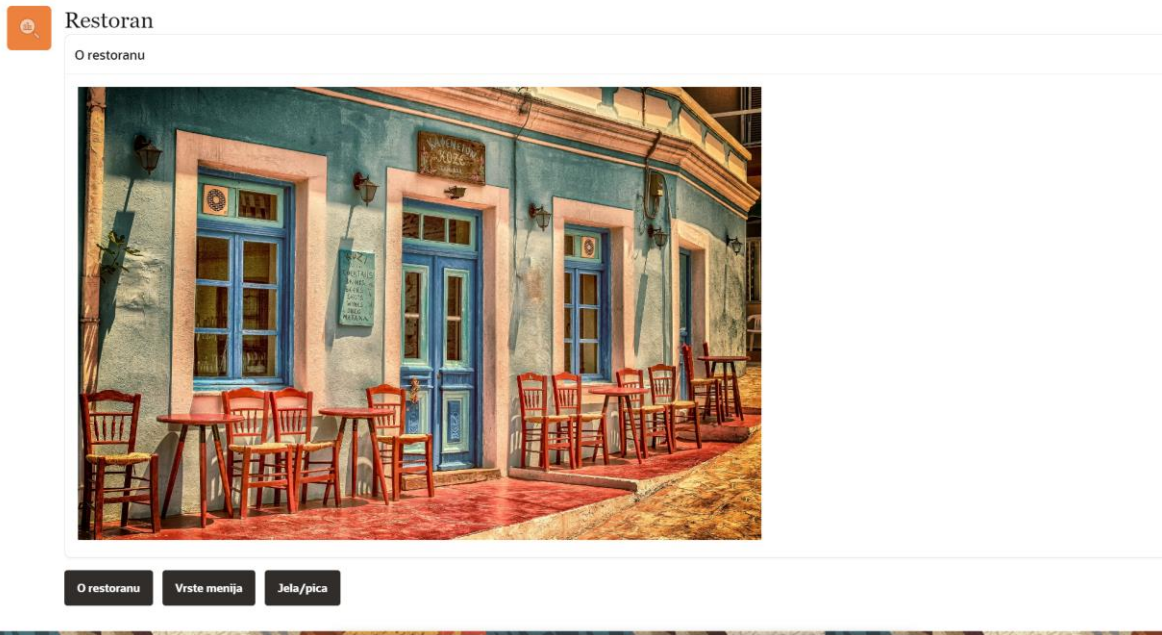
1 select k.ID_VRSTE_MENIJA,
2      k.RBR_STAVKE_MENIJA,
3      k.CIJENA_STAVKE_MENIJA,
4      k.ID_JELA_PICA
5 from STAVKE_MENIJA k where k.ID_VRSTE_MENIJA=:P4_NEW;

```

Slika 20: SQL upit za filtriranje stavka menija

4.3.1.3 Prikaz slikovnih datoteka i gumba

Postupak prikaza slikovne datoteke te gumba koji vode na određenu stranicu u aplikaciji objašnjen je na primjeru *Home* stranice (slika 21).



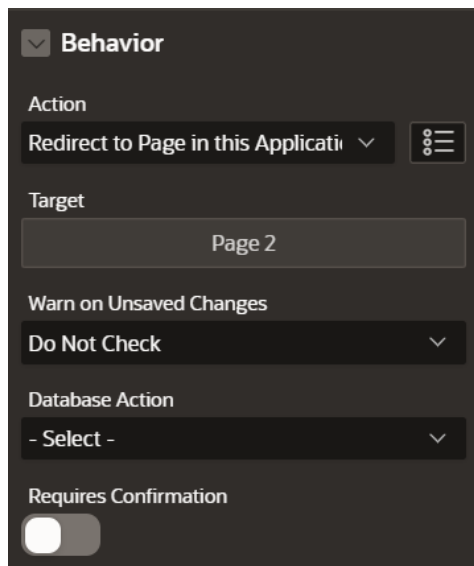
Slika 21: Home stranica – slikovni prikaz i gumbi

Naime, za dodavanje prikaza slikovne datoteke potrebno je stvoriti novu regiju tipa *Static Content*. U kategoriji *Source* u desnom izborniku moguće je uključiti sliku pomoću HTML koda. Prethodno, željenu sliku treba prenijeti unutar *Static Application Files* u dijeljenim komponentama aplikacije (engl. *Shared Components*) postupkom drag and drop. Potom se unutar spomenutog HTML polja dodaje kod kao na slici 22. U *src* dio upisuje se puni naziv slikovne datoteke kojeg je moguće iščitati prilikom prijenosa u *Static Application Files*.

```
Code Editor - HTML Code
1 
```

Slika 22: HTML kod za prikaz slike

Gumb se na stranicu ubacuje iz galerije za odabir gumba (engl. *Button Gallery*) što je već opisano u radu. Kako bi se postigao rezultat da se klikom na gumb korisnik preusmjeri na zadanu stranicu u aplikaciji, potrebno je unutar desnog izbornika u kategoriji *Behavior* pod *Action* postaviti opciju *Redirect to Page in this Application*. Isto tako, treba i definirati na koju će stranicu gumb preusmjeriti korisnika što se može učiniti unutar polja *Target* gdje se iz liste stranica aplikacije odabire ona na koju će se vršiti preusmjeravanje (slika 23).

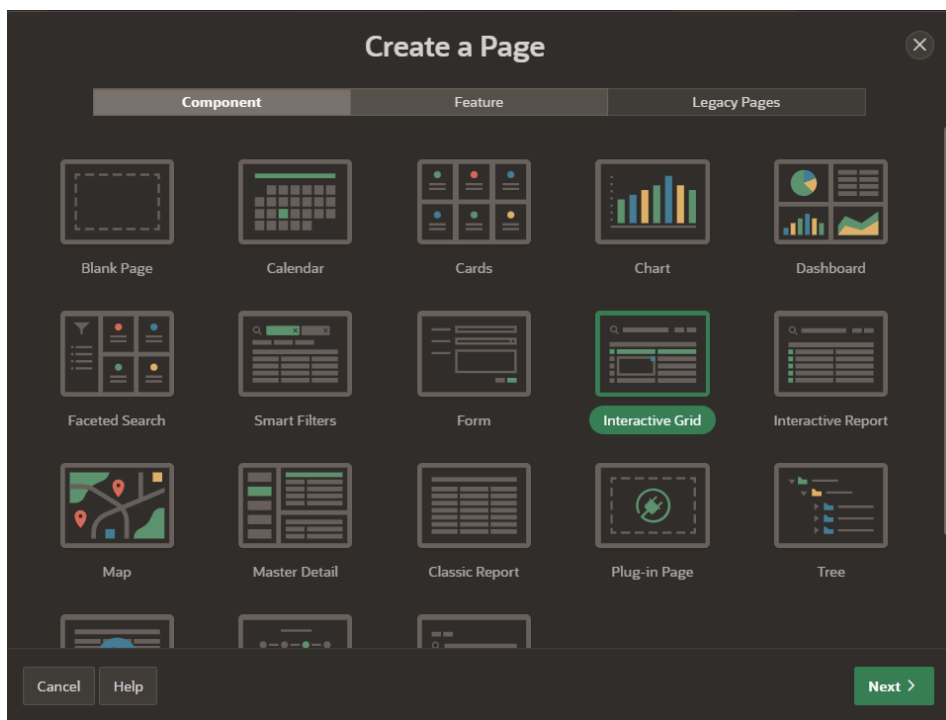


Slika 23: Gumb za preusmjeravanje na stranicu u aplikaciju

4.3.2 Interactive Grid izvješće

Općenito, izvješća služe samo za čitanje (*engl. read-only*), a ukoliko je uz čitanje potrebno i uređivati treba se koristiti forma (*engl. Form Page*). Interactive Grid izvješće popraćeno je formom kako bi se objedinile ove dvije funkcionalnosti, odnosno izravno pregledale i ažurirale vrijednosti izvješća.

Dodavanje Interactive Grid stranice objašnjeno je na primjeru jakog tipa entiteta MJESTO. Proces započinje kreiranjem nove stranice tipa *Interactive Grid* (slika 24). Potom slijedi određivanje nekakvih osnovnih informacija o izvješću poput naziva, automatskog uključivanja forme, odabira tablice koja će se prikazivati u izvješću, postavljanja navigacije i slično (slika 25).



Slika 24: Kreiranje Interactive Grid izvješća za tablicu MJESTO

Create Interactive Grid

Page Definition

* Page Number: ?

* Name: ?

Page Mode: Normal Modal Dialog ?

Include Form Page: ?

* Form Page Number: ?

* Form Page Name: ?

Data Source

Data Source: Local Database REST Enabled SQL Service REST Data Source ?

Source Type: Table SQL Query ?

* Table / View Owner: ?

* Table / View Name: ?

Navigation

Breadcrumb: ?

Navigation: ?

Navigation Preference: ?

Parent Navigation Menu Entry: ?

Icon: ?

< Cancel
Next >

Slika 25: Informacije Interactive Grid izvješća za tablicu MJESTO

Posljednji korak kojeg treba izvršiti prije kreiranja stranice jest postavljanje primarnog ključa. Za tablicu mjesto primarni ključ je *Pošt_broj* pa je sukladno tome, stupac primarnog ključa u izvješću upravo *Pošt_broj*.

Primary Key

Primary Key Column 1 POŠT_BROJ (Number) ?

Primary Key Column 2 - Select - ?

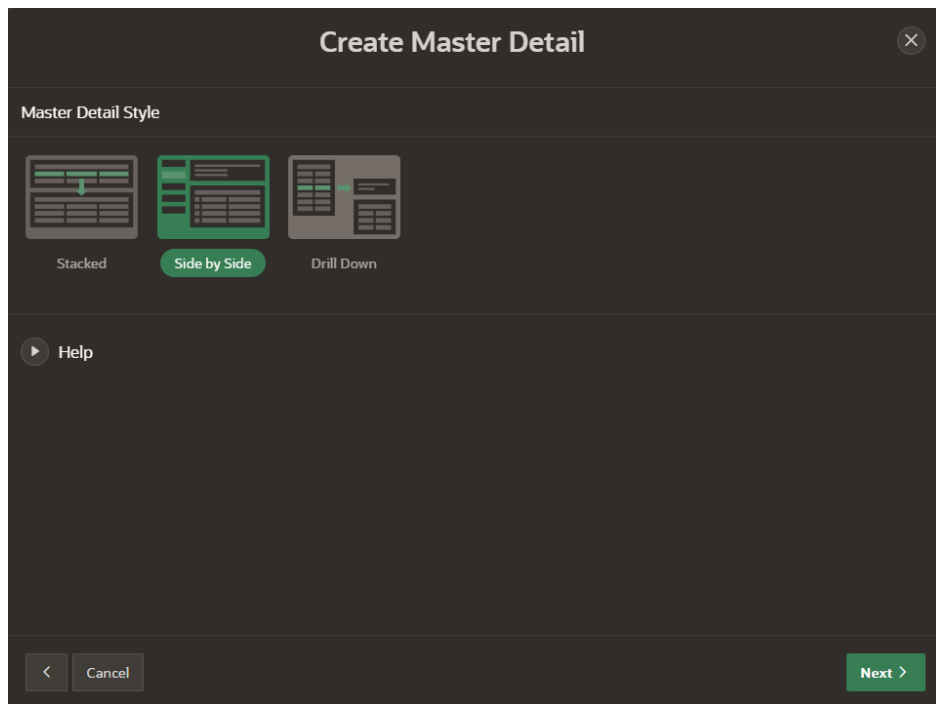
< Cancel Create Page

Slika 26: Postavljanje primarnog ključa u Interactive Grid izvješću za tablicu MJESTO

4.3.3 Master Detail izvješće

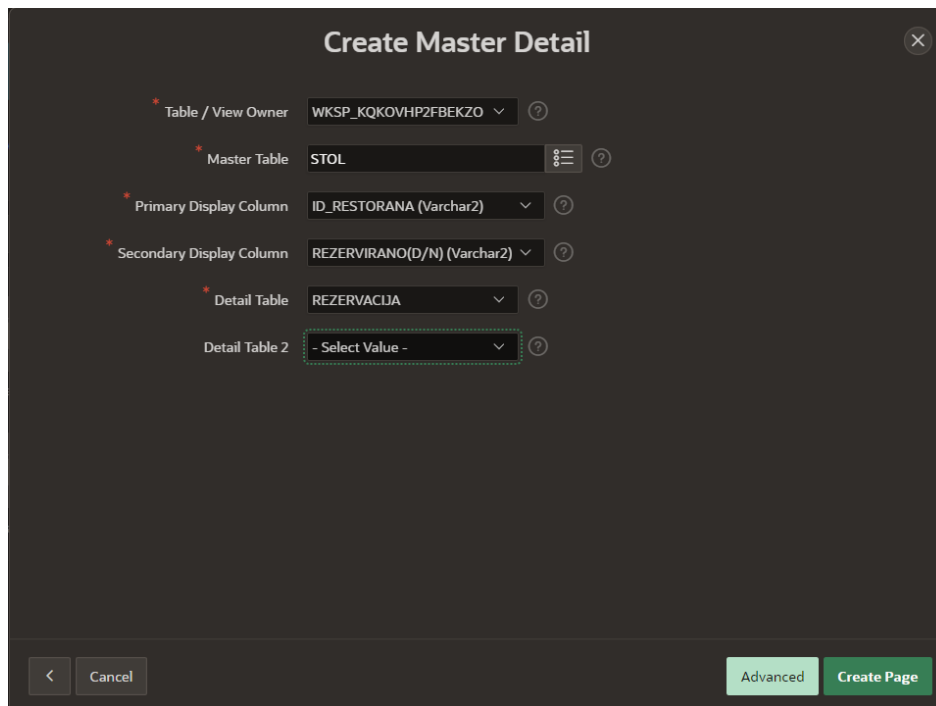
Kako bi se podaci pregledali, izmijenili, dodali ili obrisali, za slabi tip entiteta i agregaciju koristi se Master Detail izvješće (*engl. Master Detail Report*). Master Detail je zapravo izvješće koje se sastoji od dva objedinjena, povezana izvješća, glavnog i detaljnog. Odabirom jednog retka glavnog izvješća, prikazuje se odgovarajuće detaljno izvješće.

Na primjeru agregacije REZERVACIJA opisan je postupak stvaranja Master Detail izvješća. Prvi korak u izradi bio je kreiranje nove stranice tipa Master Detail (*Create Page -> Master Detail*). Potom je za stil stranice postavljena opcija *Side by Side* kako bi se u lijevom stupcu prikazali osnovni podaci glavnog izvješća dok se detalji prikazuju na sredini ekrana (slika 27).



Slika 27: Odabir stila Master Detail izvješća

Klikom na gumb *Next* otvara se novi prozor gdje se upisuje željeno ime izvješća i postavljaju određene značajke navigacije (breadcrumb, ikona, itd.). Nadalje, prije konačnog kreiranja izvješća, potrebno je odrediti tablicu glavnog i tablicu detaljnog izvješća. Za tablicu glavnog izvješća postavlja se jedan od dva jaka tipa entiteta koji čine agregaciju, a za detaljno izvješće postavlja se sama agregacija. U ovom slučaju, glavno izvješće čini tablica STOL dok detaljno izvješće čini REZERVACIJA (slika 28). Isto tako, polja *Primary Display Column* i *Secondary Display Column* služe određivanju koji će atributi glavnog izvješća biti sadržani u spomenutom lijevom stupcu prikaza.



Slika 28: Kreiranje Master Detail izvješća za tablicu REZERVACIJA

4.3.4 Izračuni (Trigger i Computation)

Unutar aplikacije postoje polja poput iznosa stavki i ukupnog iznosa za koje bi se trebao stvoriti automatski izračun koristeći vrijednosti drugih atributa. Automatsko izračunavanje vrijednosti može imati velik utjecaj na kompleksnost i učinkovitost aplikacije. Za izračunavanje vrijednosti *Iznos_stavki* iz tablice STAVKE_RAČUNA koja nastaje množenjem vrijednosti *kolicina* sa vrijednošću *Cijena_stavke_menija* iz STAVKE_MENIJA koristit će se okidač (engl. *Trigger*). Okidač je u suštini PL/SQL izraz povezan s tablicom, prikazom, shemom ili samom bazom podataka. Izvršava se automatski kada se dogodi određeni događaj.

U ovom slučaju, kreiran je okidač kao na slici 29 unutar SQL Workshop-a odabirom opcije SQL Commands. Množenjem vrijednosti atributa *Cijena_stavke_menija* iz tablice STAVKE_MENIJA te vrijednosti atributa *kolicina* iz tablice STAVKE_RACUNA dobiva se nova vrijednost koju je potrebno pohraniti kao vrijednost atributa *Iznos_stavki* iz tablice STAVKE_RACUNA. Važno je naglasiti kako treba postaviti uvjet, pomoću *where* klauzule, u kojem se definira podudaranje između atributa *Id_vrste_menija* i *Rbr_stavke_menija* u obje tablice. Ispravnost napisanog izraza može se provjeriti klikom na opciju *Run* i provjerom rezultata pokretanja unutar opcije *Results*. Ukoliko je izvršavanje uspješno, pojavit će se obavijest o uspješnom kreiranju okidača (engl. *Trigger created*). Okidač se može pohraniti odabirom opcije *Save* i navođenjem naziva novonastalog SQL izraza. Sve pohranjene SQL izraze moguće je pregledati odabirom opcije *Saved SQL*. Isto tako, atribut *Iznos_stavki* je unutar Page Designer-a stranice *Stavke_racuna* postavljen na *Hidden* kako se ne bi prikazivao pri uređivanju ili kreiranju novih stavki jer je to automatski izračun.

```

1 create or replace trigger "trigger_stavke_racuna" before insert or update on STAVKE_RACUNA
2 for each row
3 declare
4   v_cijena_stavke_menija number;
5 begin
6   select cijena_stavke_menija
7   into v_cijena_stavke_menija
8   from STAVKE_MENIJA
9   where id_vrste_menija = :new.id_vrste_menija and rbr_stavke_menija = :new.rbr_stavke_menija;
10  :new.iznos_stavki := :new.kolicina * v_cijena_stavke_menija;
11 end;

```

The screenshot shows a SQL IDE interface with a command window containing the above SQL code. Below the command window, there are tabs for 'Results', 'Explain', 'Describe', 'Saved SQL', and 'History'. A search bar is visible with 'Find' and 'Go' buttons. At the bottom, a table lists the trigger details:

Owner	Name	Description	SQL	Updated By	Updated
IVANA.KOPRICANEC2@GMAIL.COM	StavkeRacuna_Trigger	-	create or replace trigger "trigger_stavke_racuna" before insert or update on STAVKE_RACUNA for each row declare	IVANA.KOPRICANEC2@GMAIL.COM	45 hours ago

Slika 29: Okidač za izračun vrijednosti *Iznos_stavki*

Drugi izračun tiče se ukupnog iznosa računa, odnosno zbroja svih stavki nekog računa. Za navedeni problem korištena je opcija izračuna (engl. *Computation*). Izračuni su događaji koji pripremaju stavke s podacima te se postavljaju na konkretno polje koje je potrebno izračunati.

Naime, za kreiranje izračuna, potrebno je unutar Page Designer-a stranice *Racun* desnim klikom miša na polje *Ukupan_iznos* odabrati opciju *Create Computation*. Nakon toga se na desnoj strani pojavljuje izbornik gdje se unutar kategorije *Identification* definira naziv izračuna dok se unutar kategorije *Computation* postavlja tip *SQL Query (return single value)*. Potrebno je i unijeti SQL izraz za izračunavanje potrebne vrijednosti što je učinjeno unutar polja SQL Query kao na slici 30. Korištena je funkcija *sum()* koja zbraja sve vrijednosti navedene unutar zagrada. Polje *Ukupan_iznos* je kao prethodni *Iznos_stavki* postavljen na *Hidden* jer se o radi o automatskom izračunu.

```

1 select sum(iznos_stavki)
2 from stavke_racuna sr
3 join racun r on sr.id_racuna = r.id_racuna
4 where sr.id_racuna = :P14_ID_RACUNA

```

Slika 30: Izračunavanje vrijednosti *Ukupan_iznos* koristeći *Computation*

4.4 Izrada navigacije

Prilikom svakog kreiranja stranice, kreirana je i nova opcija u samoj navigaciji. Unutar dijeljenih komponenti aplikacije (engl. *Shared Components*) moguće je pronaći opciju *Navigation Menu (Shared Components -> Navigation -> Navigation Menu)* koja prikazuje postojeći navigacijski menu aplikacije (slika 31).

List: Navigation Menu Cancel Delete Apply Changes

Show All Name List Entries Subscription Configuration Comments

Name

Name Navigation Menu

List Entries Create Entry >

Search: All Text Columns Go Actions Edit Add Entry Reset

<input type="checkbox"/>		Sequence ↑=2	Name	Target	Icon	Authorization Scheme	Build Option	Level ↑=1	Parent Entry	Children	Conditional	Update
<input checked="" type="checkbox"/>		10	Home	f?p=6APP_ID:1:6APP_SESSION::6DEBU...	fa-home			1		-		2 week
<input type="checkbox"/>		20	O restoranu	f?p=6APP_ID:2:6APP_SESSION::6DEB...	fa-info			1		-		2 week
<input type="checkbox"/>		30	Menu	f?p=6APP_ID:3:6APP_SESSION::6DEB...	fa-cutlery			1		3		2 week
<input type="checkbox"/>		40	Vrste menija	f?p=6APP_ID:4:6APP_SESSION::6DEBU...	-			2	Menu	-		2 week
<input type="checkbox"/>		50	Popis jela/pica	f?p=6APP_ID:5:6APP_SESSION::6DEBU...	-			2	Menu	-		2 week
<input type="checkbox"/>		60	Rezervacija	f?p=6APP_ID:6:6APP_SESSION::6DEB...	fa-calendar			1		-		2 week
<input type="checkbox"/>		70	Jelovnik	f?p=6APP_ID:8:6APP_SESSION::6DEB...	-			2	Menu	-		2 week
<input type="checkbox"/>		80	Izvjescja	f?p=6APP_ID:10:6APP_SESSION::6DEB...	fa-file-o			1		9		2 day
<input type="checkbox"/>		90	MasterDetail_Stavke...	f?p=6APP_ID:11:6APP_SESSION::6DEB...	-			2	Izvjescja	-		2 week
<input type="checkbox"/>		110	MasterDetail_Rezerva...	f?p=6APP_ID:18:6APP_SESSION::6DEB...	-			2	Izvjescja	-		2 week
<input type="checkbox"/>		120	InteractiveGrid_Mjesto	f?p=6APP_ID:21:6APP_SESSION::6DEB...	-			2	Izvjescja	-		2 week
<input type="checkbox"/>		130	InteractiveGrid_Restor...	f?p=6APP_ID:23:6APP_SESSION::6DEB...	-			2	Izvjescja	-		2 week
<input type="checkbox"/>		140	InteractiveGrid_TipJel...	f?p=6APP_ID:25:6APP_SESSION::6DEB...	-			2	Izvjescja	-		2 week
<input type="checkbox"/>		150	InteractiveGrid_JeloPice	f?p=6APP_ID:27:6APP_SESSION::6DEB...	-			2	Izvjescja	-		2 week

Slika 31: Navigation Menu

Vidljivo je kako pojedini elementi sadrže ikonu dok su neki elementi podređeni drugima, odnosno sadrže roditeljski element. Svaki element navigacije moguće je i dodatno urediti klikom na simbol za uređivanje sa lijeve strane (slika 32).

List Entry Cancel Delete Apply Changes

Show All Entry Target Current List Entry Conditions Authorization Configuration Click Counting User Defined Attributes Developer Resources

Entry

List: Navigation Menu

Parent List Entry: - No Parent List Item -

Sequence: 10

Image/Class: fa-home

Attributes: [icon]

Alt Attribute: [input]

List Entry Label: Home

Target

Target type: Page in this Application

Page: 1

reset pagination for this page

Printer Friendly

Request: [input]

Clear Cache: [input]

Set these items: [input]

With these values: [input]

URL Target: [input]

Slika 32: Uređivanje pojedinog elementa navigacije

4.5 Autentifikacija i autorizacija

Za postizanje autentifikacije stvorena je nova tablica *Users* unutar Oracle APEX sučelja kako je opisano ranije u radu. Tablica sadrži atribute *Username*, *Password* te *IsAdministrator* koji služi

raspoznavanju je li neki korisnik administrator sustava (slika 33). Također, ne postoje ograničenja poput primarnog ključa, vanjskog ključa i slično.

Column Name	Data Type	Nullable	Default	Primary Key
USERNAME	VARCHAR2(50)	Yes	-	-
PASSWORD	VARCHAR2(1000)	Yes	-	-
ISADMINISTRATOR	CHAR(1)	Yes	-	-

Slika 33: Kreirana tablica Users

Nakon same izrade tablice, u bazu su unesena dva korisnika koristeći *SQL Scripts* unutar *SQL Workshop-a*, kako je prikazano slikom 34. Prvi korisnik je administrator sustava stoga mu je vrijednost atributa *IsAdministrator* postavljena na „Y“. Korisničko ime i lozinka jesu redom *ADMIN*, *admin*. Iz sigurnosnih razloga korišteno je heširanje lozinki što je omogućeno pomoću implementirane metode *apex_util.get_hash*. Rezultat heširanja može se vidjeti na slici 35. Drugi je korisnik zapravo gost koji će imati mogućnost isključivo pregledavanja sadržaja. Korisničko ime je *GUEST* dok je lozinka postavljena na *null*, odnosno nije je potrebno unositi.

```

1 insert into USERS (UserName, Password, IsAdministrator) values ('ADMIN', apex_util.get_hash(apex_t_varchar2('ADMIN', 'admin'), null), 'Y');
2 insert into USERS (UserName, Password, IsAdministrator) values ('GUEST', apex_util.get_hash(apex_t_varchar2('GUEST', null), null), 'N');

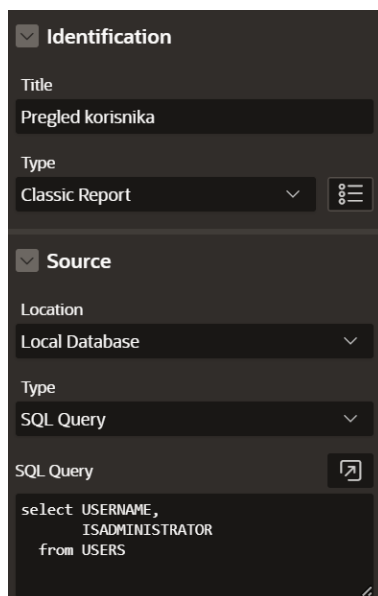
```

Slika 34: Kreiranje korisnika unutar SQL Scripts

EDIT	USERNAME	PASSWORD	ISADMINISTRATOR
	ADMIN	Zs4AulDyYmRpNeete-5TWohu5UyyVaxnifvIIUvselwCIAmpa50YB4ZleGusGsq64rorIM3-yf5Zf0sMIayw	Y
	GUEST	wR8sdanx_8H9-khbMa6PHEoMwXm5u36PYQorap5PU0zqLjywjTlhhgmp7e6RrW8x4i00IM5WfwbwO_bNwyOg	N

Slika 35: Pregled kreiranih korisnika

Idući korak je stvaranje stranice za pregled postojećih korisnika i dodavanje novih korisnika. Stranica je tipa *Blank Page* u kojoj su dodane dvije regije *Pregled korisnika* te *Kreiranje korisnika*. Regija *Pregled korisnika* je *Classic Report*, a parametri su postavljeni kao na slici 36 kako bi se uspješno prikazivali podaci o korisničkom imenu te je li postojeći korisnik admin.



Slika 36: Postavke regije *Pregled korisnika*

Regija *Kreiranje korisnika* je tipa *Static Content* te su u nju iz galerije za odabir *Item-a* i gumba smještena polja tipa *Text Field*, *Password*, *Checkbox* te *Text[Hot]* gumb. Kako bi se korisnik ispravno kreirao, potrebno je stvoriti novi proces desnim klikom na kategoriju *Processing* i odabirom opcije *Create Process*. Proces služi heširanju lozinke novog korisnika što je ostvareno kodom na slici 37.

```
Code Editor - PL/SQL Code
1 declare
2   v_username varchar2(20) := upper(:P7_USERNAME);
3   v_valuesToHash apex_t_varchar2 := apex_t_varchar2(v_username, :P7_PASSWORD);
4 begin
5   insert into USERS (UserName, Password, IsAdministrator) values (v_username, apex_util.get_hash(v_valuesTo
6 end;
```

Slika 37: Proces za unos novog korisnika

Također, na sličan način stvorene su dvije validacije, desnim klikom miša na kategoriju *Validations* i odabirom opcije *Create Validation*. Prva validacija tiče se pohranjivanja svih korisničkih imena u obliku velikih slova (slika 38) dok se pomoću druge validacije provjerava odgovara li ponovljena lozinka unesenoj lozinki (slika 39).

☑ Identification

Name
Username

☑ Execution Options

Sequence
10

☑ Validation

Editable Region
- Select -

Type
No Rows returned

SQL Query

```
select * from USERS where UserName =  
upper(:P7_USERNAME)
```

Always Execute

Slika 38: Username validation

Identification

Name

Password

Execution Options

Sequence

20

Validation

Editable Region

- Select -

Type

Expression

Language

PL/SQL

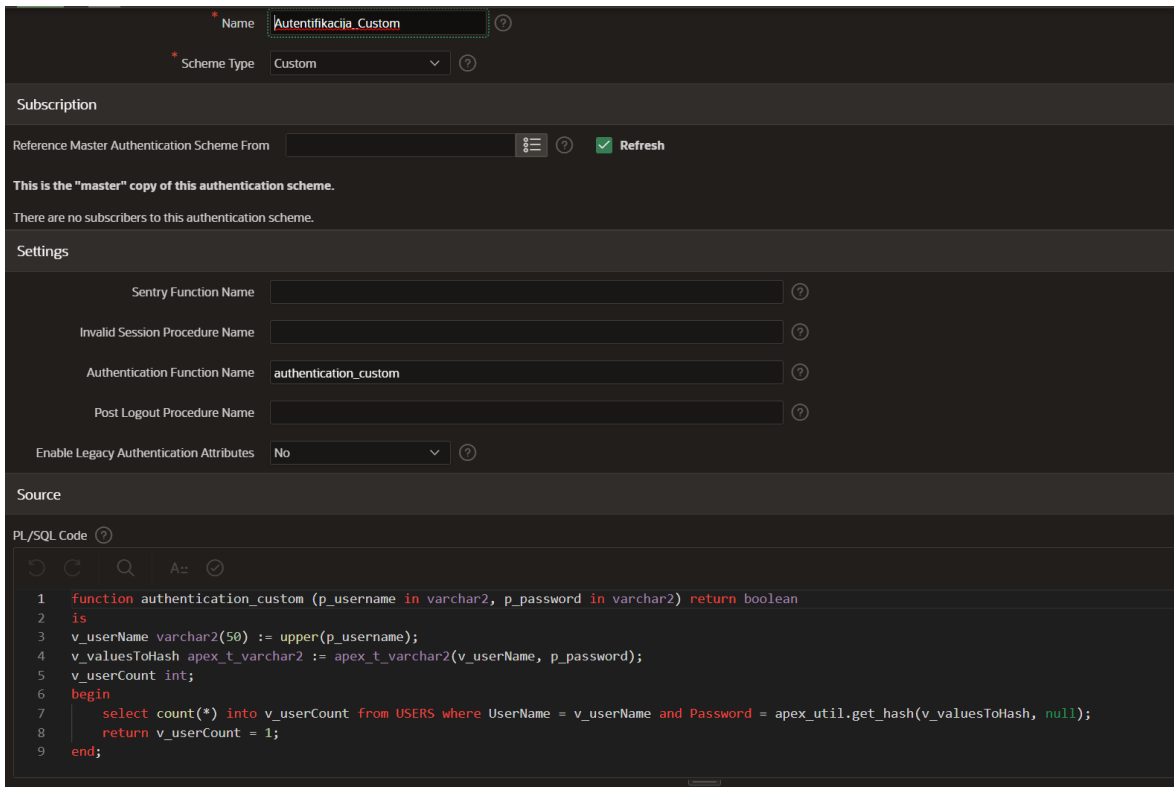
PL/SQL Expression

```
(:P7_PASSWORD = :P7_PASSWORD_CONFRIM)
or (:P7_PASSWORD is null and
:P7_PASSWORD_CONFRIM is null)
```

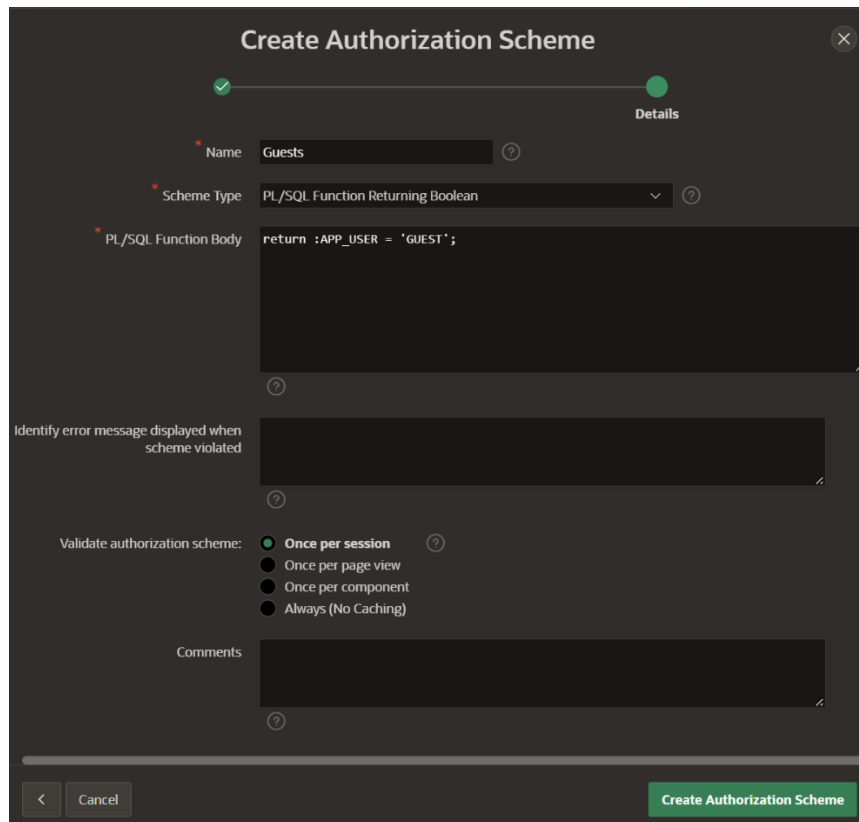
Always Execute

Slika 39: Password validation

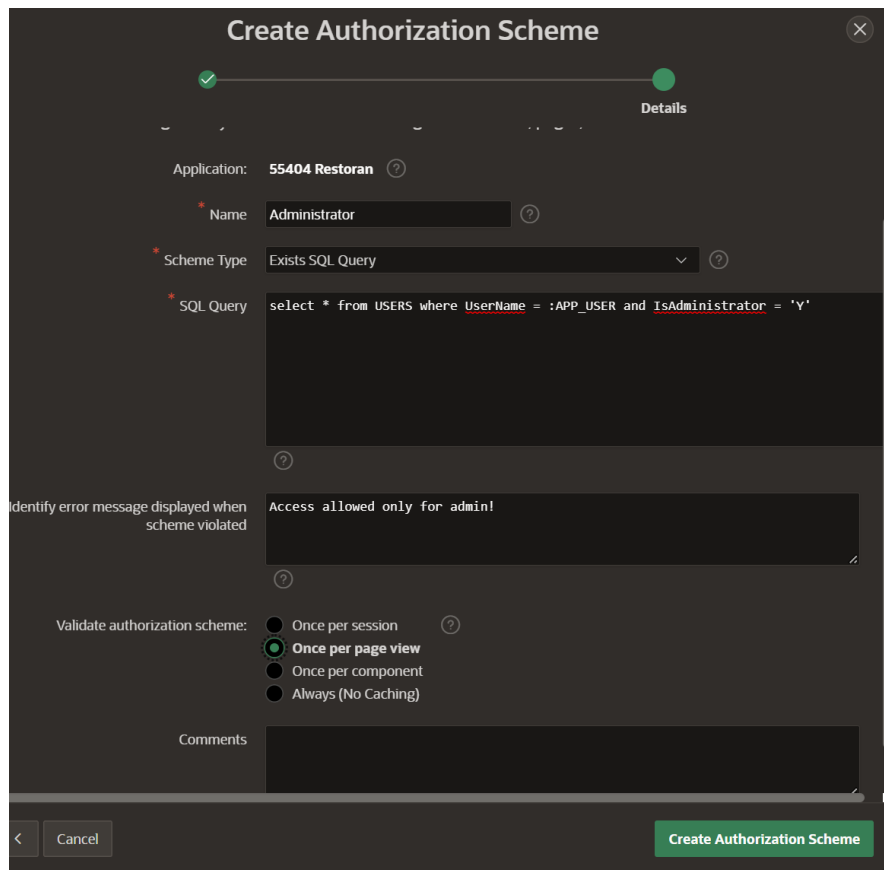
Zadnji korak je izrada triju shema, jedne autentifikacijske i dvije autorizacijske. To je postignuto odabirom *Authentication Scheme* i *Authorization Scheme* unutar dijeljenih komponenti aplikacije (engl. *Shared Components*) pod kategorijom *Security*. Za kreiranje potrebno je odabrati opciju *Create* pri čemu se za autentifikacijsku shemu odabire opcija *Based on a pre-configured scheme from the gallery* dok se za autorizacijsku odabire *From Scratch*. Nadalje, potrebno ih je ispuniti kako je prikazano na slikama 40, 41 i 42. Postoje dvije autorizacijske sheme zbog dvije vrste korisnika, administratora sustava i gosta.



Slika 40: Authentication scheme

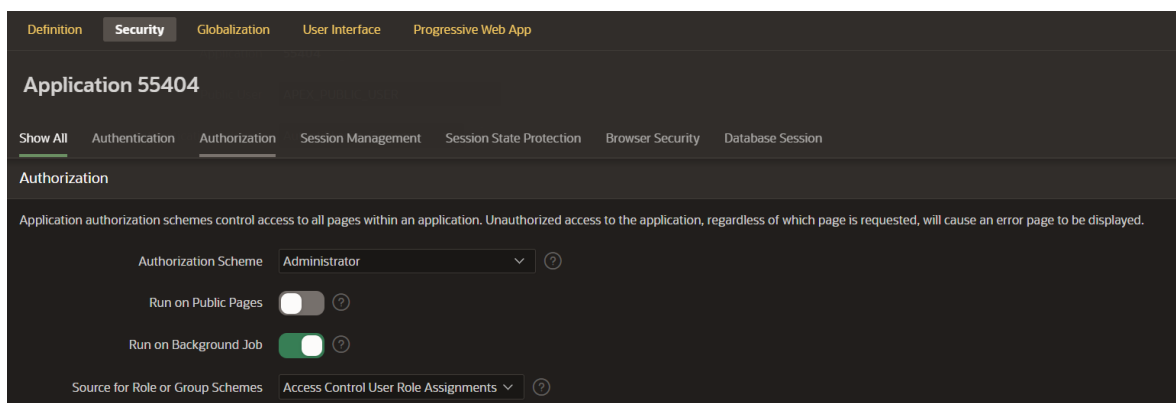


Slika 41: Authorization scheme – Guest

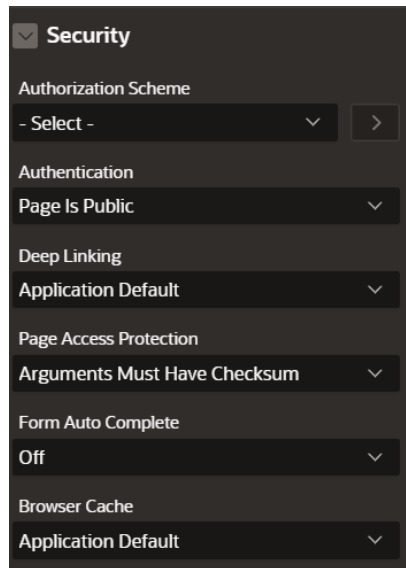


Slika 42: Authorization scheme – Administrator

Autorizacija je primijenjena na razini aplikacije što je postignuto postavljanjem administratorske sheme unutar polja *Authorization scheme* kao na slici 43, a kako se prijava kao administrator ne bi odnosila na sve stranice u aplikaciji potrebno je unutar svake stranice s pregledom sadržaja u kategoriji *Security* postaviti stranicu na javno (slika 44). Na taj način i gosti mogu pregledavati sadržaj aplikacije bez prijave.



Slika 43: Primjena autorizacije na razini aplikacije

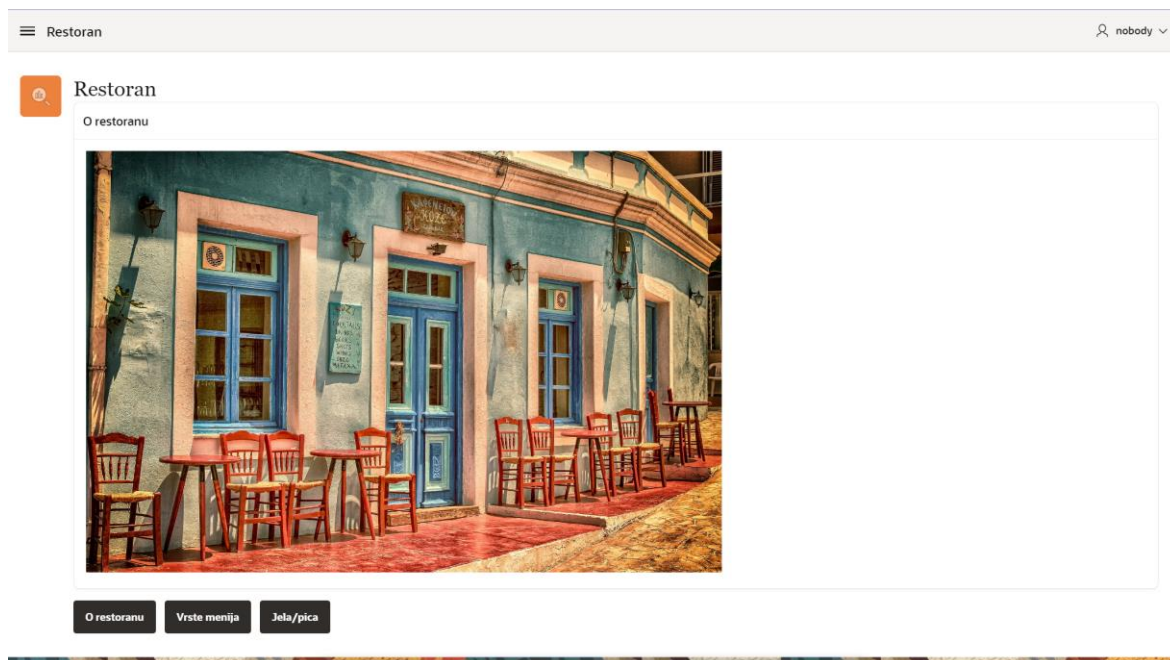


Slika 44: Postavljanje stranice na javno

5. Prikaz izrađene aplikacije

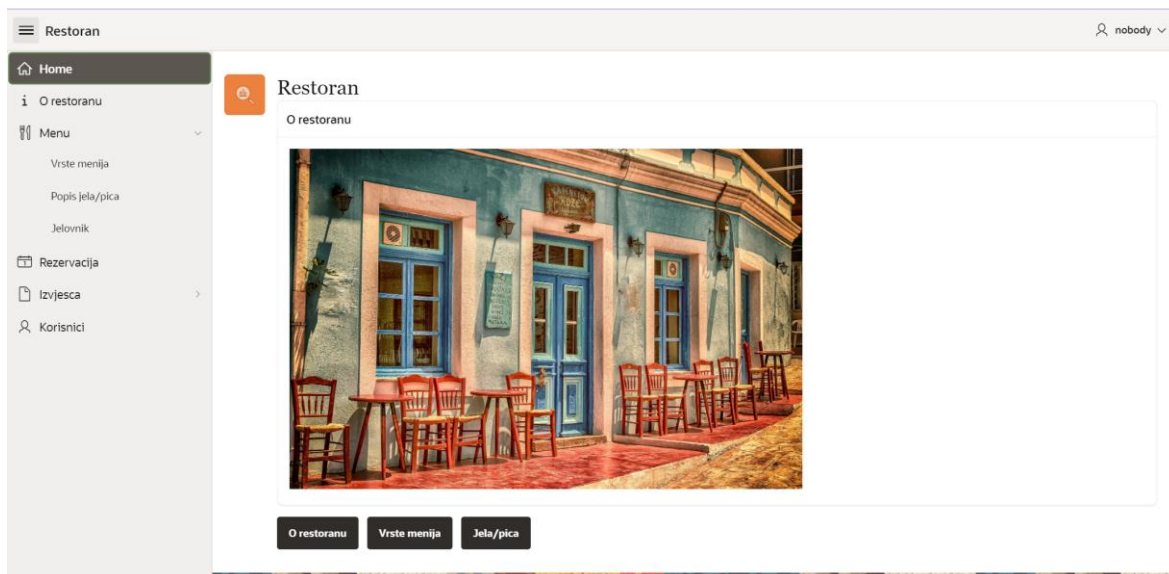
Izrađenu aplikaciju moguće je pregledati na sljedećoj poveznici: <https://apex.oracle.com/pls/apex/r/restoran/restoran/home?session=6978733455181>. Za pristup dodatnim funkcionalnostima poput dodavanja, uređivanja i brisanja moguće se prijaviti s korisničkim imenom *ADMIN* te lozinkom *admin*. U nastavku su prikazane neke od fotografija završnog produkta, aplikacije *Restoran*.

Prilikom pokretanja aplikacije, ono što se prvo pojavi jest početna stranica (engl. *Home Page*) prikazana na slici 45.

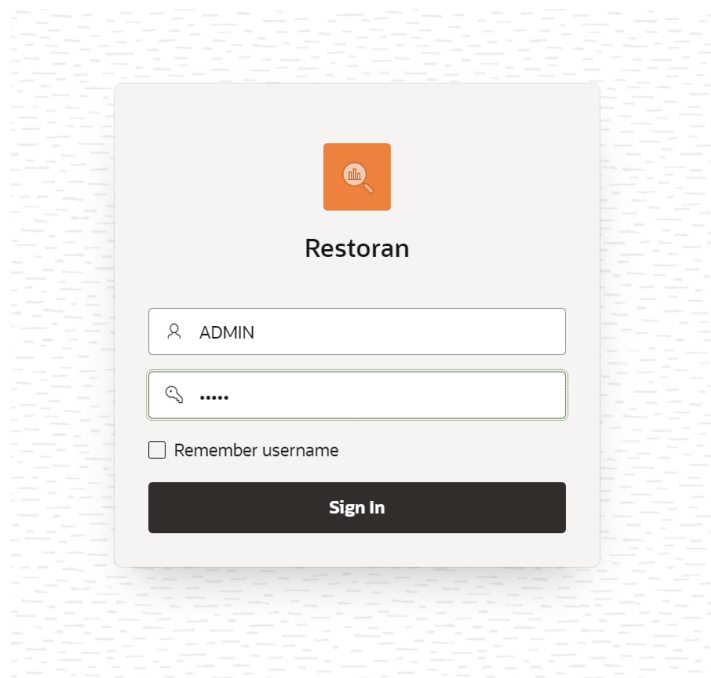


Slika 45: Početna stranica

U gornjem lijevom kutu moguće je prikazati izbornik sa sveukupnih pet kategorija, od čega su dva padajuća izbornika (slika 46). Za prikaz kategorija *Izvjeshća* te *Korisnici*, potrebno je imati administratorske ovlasti, odnosno prijaviti se kao administrator (slika 47).

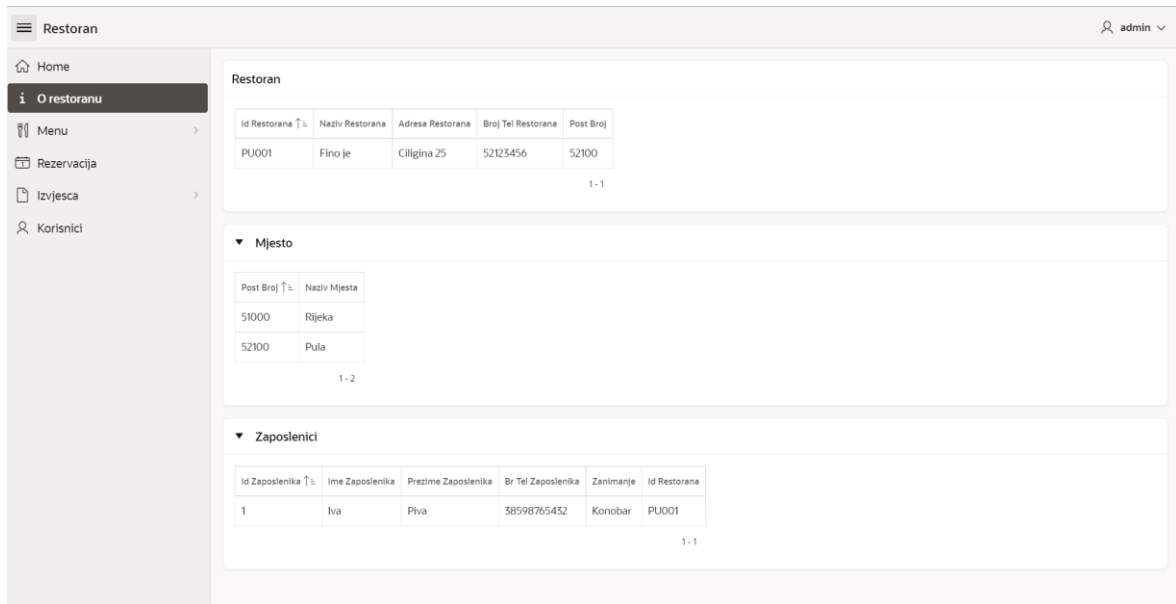


Slika 46: Prikaz navigacije

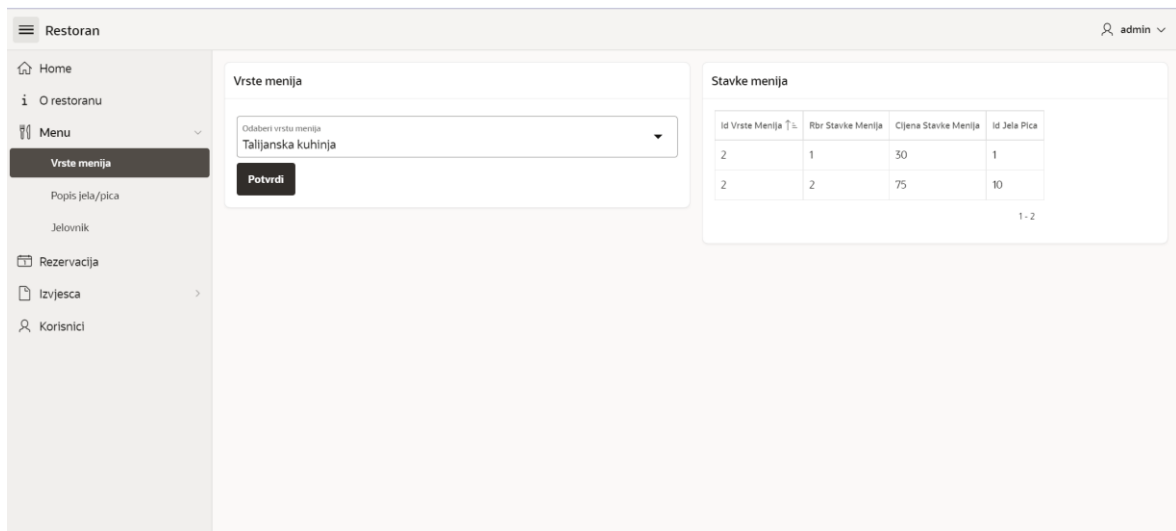


Slika 47: Prijava kao administrator

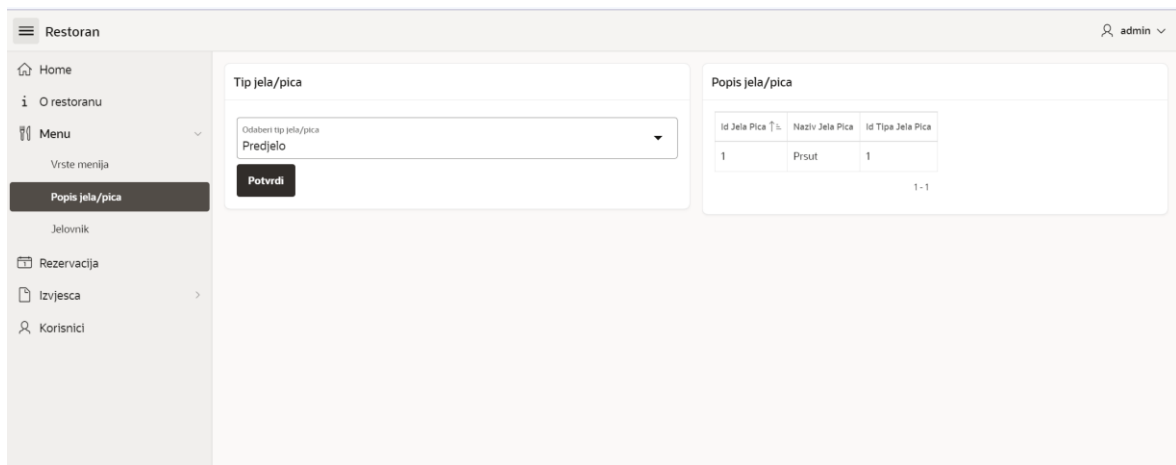
Od pregleda i filtriranja sadržaja, omogućeno je pregledavanje podataka o restoranu (slika 48), filtriranje stavka menija prema vrsti menija (slika 49), filtriranje jela i pića prema tipu jela i pića (slika 50), pregled cijelog jelovnika restorana (slika 51) te pregled rezervacija (slika 52).



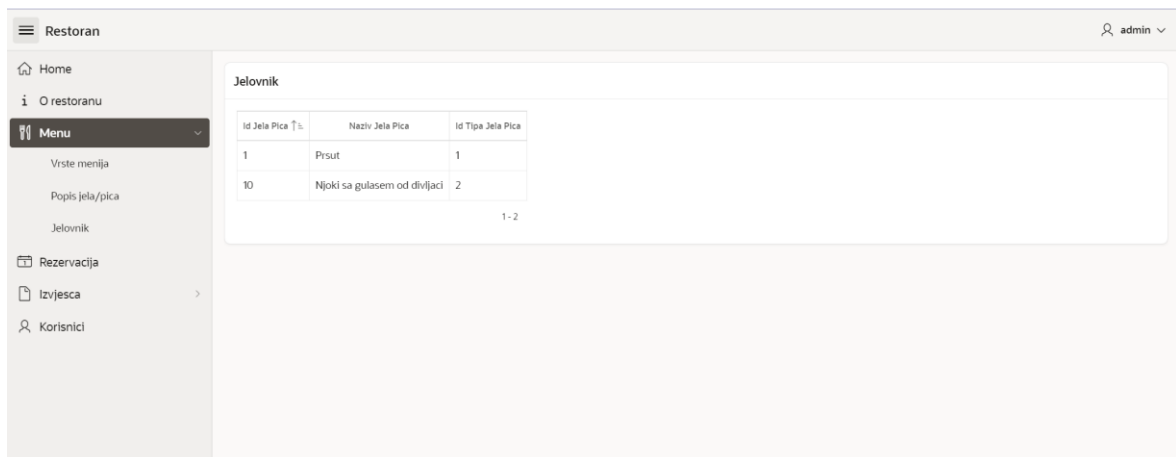
Slika 48: Stranica *O restoranu*



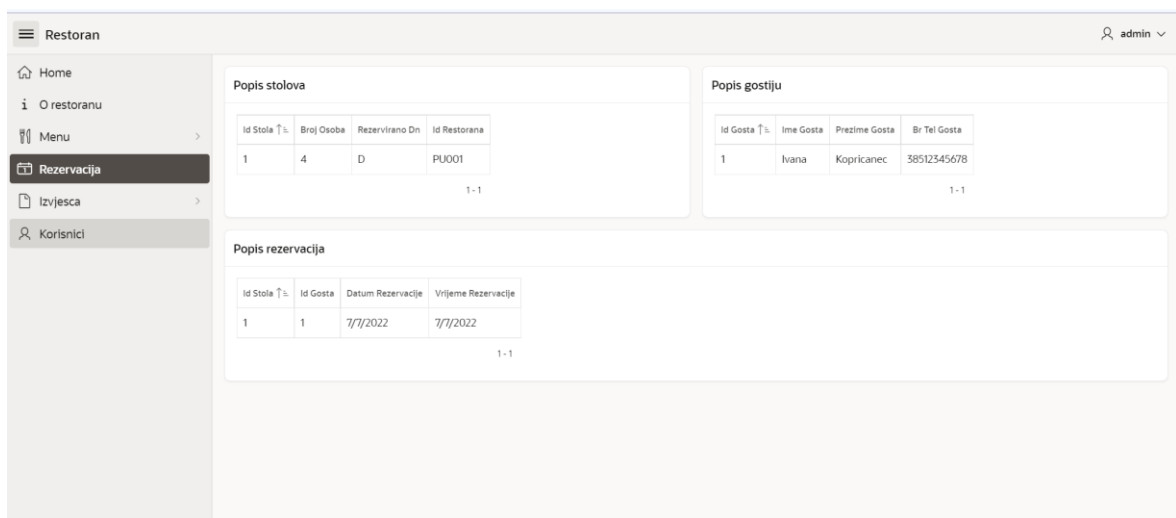
Slika 49: Stranica *Vrste menija*



Slika 50: Stranica *Popis jela/pica*



Slika 51: Stranica *Jelovnik*



Slika 52: Stranica *Rezervacija*

Kao administrator, moguće je kreirati novi sadržaj i urediti ili izbrisati postojeći. U nastavku su prikazani neki od primjera, Master Detail-a te Interactive Grid-a.

Restoran admin

Home
O restoranu
Menu
Rezervacija
Izvjescja

MasterDetail_StavkeMenija
7/7/2022
6/7/2022
MasterDetail_Rezervacija
7/28/2022
7/28/2022
InteractiveGrid_Mjesto
InteractiveGrid_Restoran
InteractiveGrid_TipJelaPica
InteractiveGrid_JeloPice
InteractiveGrid_Gost
InteractiveGrid_Zaposlenik
MasterDetail_StavkeRacuna
Korisnici

MasterDetail_StavkeRacuna

Search...

Racun Edit

Ukupan Iznos: 630
Datum Racuna: 7/7/2022
Vrijeme Racuna: 6/7/2022
Id Stola: 1
Id Gosta: 1
Id Zaposlenika: 1

Stavke Racuna +

	Rbr Stavke Racuna	Kolicina	Iznos Stavki	Id Vrste Menija	Rbr Stavke Menija
<input type="checkbox"/>	1	4	280	1	2
<input type="checkbox"/>	2	3	90	1	1
<input type="checkbox"/>	3	4	120	1	1
<input type="checkbox"/>	4	2	140	1	2

1 - 4

Slika 53: Prikaz Master Detail-a Stavke računa

Racun ✕

Id racuna

Datum Racuna 📅
Required

Vrijeme Racuna 📅
Required

Id Stola
Required

Id Gosta
Required

Id Zaposlenika
Required

Cancel Create

Slika 54: Kreiranje novog računa (Master Detail)

Stavke Racuna
✕

Id Racuna

1

Required

Rbr Stavke Racuna

Required

Kolicina

Required

Id Vrste Menija

Required

Rbr Stavke Menija

Required

Cancel

Create

Slika 55: Kreiranje nove stavke računa (Master Detail)

Restoran
admin

InteractiveGrid_Restoran
Create

Q
Search: All Text Columns
Go
Actions
Reset

	Naziv Restorana	Adresa Restorana	Broj Tel Restorana	Post Broj
✎	Fino je	Ciligina 25	52123456	Pula

1 rows selected
Total 1

Slika 56: Prikaz Interactive Gride-a Restoran

FormPage_Restoran

ID restorana Required

Naziv Restorana Required

Adresa Restorana Required

Broj Tel Restorana

Post Broj

Pula

Rijeka

Cancel Create

Slika 57: Kreiranje novog restorana

U administratorske ovlasti uključen je i pregled postojećih korisnika kao i dodavanje novih kao što je prikazano na slici 58.

Restoran admin

Home

O restoranu

Menu

Rezervacija

Izvjesca

Korisnici

Pregled korisnika

Username	Isadministrator
ADMIN	Y
GUEST	N
IVANA	Y

1-3

Kreiranje korisnika

Username
IVANA

Password

Confirm Password

Is Administrator

Kreiraj

Slika 58: Prikaz stranice *Korisnici*

Izgled i funkcionalnosti cjelokupne aplikacije moguće je ispitati na gore spomenutoj poveznici.

6. Zaključak

Završni produkt ovog rada je poslovna aplikacija za restoran gdje korisnik, kao gost, može pronaći informacije o restoranu kao i pregledavati i filtrirati sadržaj restorana dok administrator, uz to, ima mogućnost dodavanja novog sadržaja te uređivanja ili brisanja postojećeg. Prije same izrade aplikacije pomoću Oracle APEX alata, izrađeni su odgovarajući modeli koji sadržavaju poslovanje restorana. Za izradu aplikacije temelj su predstavljali izrađeni modeli, a korišteno je APEX sučelje te SQL (PL/SQL) izrazi.

Kako bi aplikacija bila učinkovita i kvalitetno izrađena, potrebno je napraviti čvrst temelj, odnosno poznavanje poslovnog sustava pretvoriti u ispravan i detaljan model. Pri izradi modela, ali i općenito, vrlo je važno razmišljati do kojih bi poteškoća moglo doći u kasnijoj izradi kako bi se izbjegla pojava greški i nelogičnosti u radu aplikacije.

Oracle APEX, kao i sve, ima svoje prednosti i nedostatke. Glavna prednost Oracle APEXA-a je low-code koji omogućava implementaciju programa s vrlo malom količinom samostalnog programiranja. Obzirom da se sve odvija uz pomoć grafičkog sučelja, ovaj alat je idealan za osobe koje nemaju opširnog programerskog znanja. Zbog svoje jednostavnosti, pogodan je za učenje te uz to što se iskorištava fleksibilnost i skalabilnost, promiče se i produktivnost. Također, postavljanje radnog okruženja je jednostavno i lako obzirom da se cijeli proces odvija uz pomoć čarobnjaka. Još neke od prednosti ovog alata jesu: brz razvoj aplikacije, jednostavno povezivanje s nekoliko baza podataka (uz već implementiranu Oracle APEX bazu podataka), fleksibilnost u dizajnu i rasporedu stranica, podržavanje velikog broja korisnika bez rušenja aplikacije i drugo. Kao jedna veća prednost mogla bi se istaknuti i značajka, bitna mnogim korisnicima, da se može raditi u dva različita okruženja kao što su: online okruženje i lokalno okruženje. Na taj način, cijeli rad je omogućće obaviti korištenjem internetskog preglednika ili preuzimanjem cjelokupnog sustava na osobno računalo. Najvažnije je kako je ovaj alat je dostupan svima te nije potrebna licenca za rad.

Govoreći o nedostacima, može se istaknuti nekoliko činjenica poput nemogućnosti povezivanja s većim brojem drugih trenutno popularnih baza podataka, automatskog brisanje aplikacije nakon određenog vremenskog perioda neaktivnosti te offline način rada.

Krajnji zaključak je kako je Oracle APEX podosta jednostavan alat za brzu izradu aplikacija. Elementi se mogu namjestiti jednostavnim klikom miša unutar samog sučelja, bez pisanja koda. Upravo to, APEX čini prigodnim za one koje interesira low-code aplikacija. Isto tako, besplatan je i lako dostupan alat što uvelike pridonosi njegovoj rasprostranjenosti i učestalosti korištenja.

7. Literatura

APEX, Oracle. *What is Oracle APEX?* n.d. <https://apex.oracle.com/en/> (pokušaj pristupa Srpanj 2022.).

Materijali s predavanja i vježbi kolegija Administriranje i sigurnost baza podataka. n.d. (pokušaj pristupa Srpanj 2022.).

Oracle Corporation. 20. Srpanj 2022. https://en.wikipedia.org/wiki/Oracle_Corporation (pokušaj pristupa Srpanj 2022.).

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

Sciore, Edward. *Understanding Oracle APEX 20 Application Development.* Newton: Apress, 2020.

8. Popis slika

Slika 1: Dijagram entiteti-veze poslovanja restorana.....	7
Slika 2: Zahtjev za kreiranje radnog okruženja.....	9
Slika 3: Uspješno kreiranje radnog okruženja.....	9
Slika 4: Kreiranje tablice MJESTO.....	10
Slika 5: Kreiranje primarnog ključa tablice MJESTO	11
Slika 6: Tablica MJESTO	11
Slika 7: Vanjski ključ tablice RESTORAN	12
Slika 8: Atributi tablice REZERVACIJA	13
Slika 9: Kreiranje primarnog ključa tablice REZERVACIJA.....	13
Slika 10: Kreiranje aplikacije	14
Slika 11: Kreiranje aplikacije Restoran.....	14
Slika 12: Odabir izgleda aplikacije	15
Slika 13: Kreiranje stranice tipa <i>Blank Page</i>	15
Slika 14: Postavke stranice tipa <i>Blank Page</i>	16
Slika 15: Postavke regije <i>Jelovnik</i>	17
Slika 16: Padajuća lista iz prozora <i>Item Gallery</i>	17
Slika 17: SQL upit za padajuću listu <i>Vrste menija</i>	18
Slika 18: Gumb za filtriranje vrsta menija	18
Slika 19: Postavke gumba za filtriranje vrsta menija	19
Slika 20: SQL upit za filtriranje stavka menija	19
Slika 21: <i>Home</i> stranica – slikovni prikaz i gumbi	20
Slika 22: HTML kod za prikaz slike	20
Slika 23: Gumb za preusmjerenje na stranicu u aplikaciju.....	21
Slika 24: Kreiranje Interactive Grid izvješća za tablicu MJESTO.....	22
Slika 25: Informacije Interactive Grid izvješća za tablicu MJESTO	23
Slika 26: Postavljanje primarnog ključa u Interactive Grid izvješću za tablicu MJESTO.....	24
Slika 27: Odabir stila Master Detail izvješća	25
Slika 28: Kreiranje Master Detail izvješća za tablicu REZERVACIJA.....	26
Slika 29: Okidač za izračun vrijednosti <i>Iznos_stavki</i>	27
Slika 30: Izračunavanje vrijednosti <i>Ukupan_iznos</i> koristeći <i>Computation</i>	27
Slika 31: Navigation Menu	28
Slika 32: Uređivanje pojedinog elementa navigacije	28
Slika 33: Kreirana tablica Users.....	29
Slika 34: Kreiranje korisnika unutar <i>SQL Scripts</i>	29
Slika 35: Pregled kreiranih korisnika	29
Slika 36: Postavke regije <i>Pregled korisnika</i>	30
Slika 37: Proces za unos novog korisnika.....	30
Slika 38: Username validation	31
Slika 39: Password validation	32
Slika 40: Authentication scheme.....	33
Slika 41: Authorization scheme – Guest.....	33
Slika 42: Authorization scheme – Administrator.....	34
Slika 43: Primjena autorizacije na razini aplikacije	34
Slika 44: Postavljanje stranice na javno.....	35
Slika 45: Početna stranica	36
Slika 46: Prikaz navigacije.....	37

Slika 47: Prijava kao administrator	37
Slika 48: Stranica <i>O restoranu</i>	38
Slika 49: Stranica <i>Vrste menija</i>	38
Slika 50: Stranica <i>Popis jela/pica</i>	38
Slika 51: Stranica <i>Jelovnik</i>	39
Slika 52: Stranica <i>Rezervacija</i>	39
Slika 53: Prikaz Master Detail-a <i>Stavke računa</i>	40
Slika 54: Kreiranje novog računa (Master Detail)	40
Slika 55: Kreiranje nove stavke računa (Master Detail)	41
Slika 56: Prikaz Interactive Gride-a <i>Restoran</i>	41
Slika 57: Kreiranje novog restorana.....	42
Slika 58: Prikaz stranice <i>Korisnici</i>	42

9. Popis priloga

Poveznica na poslovnu aplikaciju za restoran:

<https://apex.oracle.com/pls/apex/r/restoran/restoran/home?session=6978733455181>