

Rezervacija autobusnih karata - poslovna aplikacija nad relcijskom bazom podataka (Oracle APEX)

Nikolić, Mario

Undergraduate thesis / Završni rad

2024

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:195:746473>

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

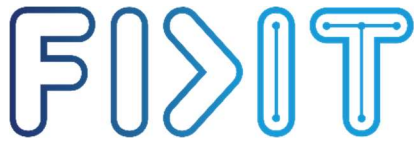
Download date / Datum preuzimanja: **2024-08-28**



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 prijediplomski studij Informatika

Mario Nikolić

**Rezervacija autobusnih karata –
poslovna aplikacija nad relacijskom
bazom podataka (ORACLE APEX)**

Završni rad

Mentor: dr. sc. Kristian Stančin

Rijeka, srpanj 2024.

Rijeka, 14. lipanj 2024.

Zadatak za završni rad

Pristupnik: Mario Nikolić

Naziv završnog rada: Rezervacija autobusnih karata - poslovna aplikacija nad relacijskom bazom podataka (Oracle APEX)

Naziv završnog rada na engleskom jeziku: Reservation of bus tickets - business application with relational database (Oracle APEX)

Sadržaj zadatka: Cilj je završnog rada izgraditi aplikaciju za potrebe rezervacije autobusnih karata. 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. Cjelokupan proces izrade praktičnog rada bit će opisan kroz prikaz koda, ekrana i funkcionalnosti. Na kraju rada bit će dan kritički osvrt te izneseni prigodni zaključci.

Mentor
Dr. sc. Kristian Stančin

Voditelj za završne radove
Izv. prof. dr. sc. Miran Pobar

Zadatak preuzet: 14. lipanj 2024.

Mario Nikolić
(potpis pristupnika)

Sažetak

Tema ovog završnog rada izrada je aplikacije unutar platforme Oracle APEX. U radu je dan pregled postupka izrade aplikacije s detaljnim opisima i snimkama zaslona. Konkretno, unutar alata Oracle APEX izrađena je poslovna aplikacija namijenjena za rezervaciju autobusnih karata. Najprije je dan opis aplikacije i njezinih ciljeva pomoću korisničkih priča, potom je demonstrirana ciljana grupa korisnika te je predstavljeno nekoliko profila konkretnih budućih korisnika aplikacije (modeli korisnika). Nakon toga, korištenjem metodologije MIRIS detaljno je opisan postupak izgradnje modela baze podataka koji se potom prevodi u relacijski model koji je neophodan za samu implementaciju aplikacije. Potom slijedi pregled realizacije projekta unutar platforme Oracle APEX. Naposljetku je dan prikaz kreirane aplikacije sa svojim CRUD (Create, Read, Update, Delete) mogućnostima koje nudi krajnjim korisnicima.

Ključne riječi: Oracle APEX; Korisničke priče; Modeli korisnika; Rezervacija autobusnih karata; MIRIS; Relacijske baze podataka; CRUD

SADRŽAJ

| | |
|---|----|
| 1. Uvod | 2 |
| 2. Korisničke priče..... | 3 |
| 3. Korisničke persone | 4 |
| 4. Modeliranje podataka | 6 |
| 4.1. Specijalizirana metodologija MIRIS..... | 7 |
| 4.2. Dijagram entiteti-veze..... | 10 |
| 4.3. Relacijska shema..... | 12 |
| 5. Implementacija aplikacije u ORACLE Apex 23.2 | 13 |
| 5.1. Izrada baze podataka (SQL Workshop)..... | 14 |
| 5.2. Izrada aplikacije (App Builder) | 20 |
| 5.2.1. Izrada stranice „Popis ruta“ | 24 |
| 5.2.2. Izrada stranice za unos nove rezervacije..... | 27 |
| 5.2.3. Izrada stranice za prikaz svih rezervacija | 33 |
| 5.2.4. Implementacija potvrde o rezervaciji..... | 38 |
| 5.2.5. Izrada preostalih stranica | 43 |
| 5.2.6. Implementacija registracije novog korisnika | 45 |
| 5.2.7. Autentifikacija korisnika..... | 48 |
| 5.2.8. Autorizacija korisnika..... | 49 |
| 6. Prikaz aplikacije | 53 |
| 6.1. Prikaz stranice „POPIS RUTA“..... | 53 |
| 6.2. Prikaz stranica „REGISTRACIJA“ i „LOG IN“ | 53 |
| 6.3. Prikaz stranica „Nova rezervacija“ i „REZERVACIJA“ | 54 |
| 7. Zaključak | 57 |
| 8. Literatura | 58 |
| 9. Popis slika..... | 59 |
| 10. Prilozi..... | 61 |

1. Uvod

Rapidnim razvojem informacijske i komunikacijske tehnologije svakodnevno dolazi do digitalizacije poslovanja tvrtki različitih profila. Tvrtke koje nude usluge korisnicima sve više i više svoje poslovanje djelomično ili u potpunosti digitaliziraju s ciljem optimizacije vremena i samog poslovanja.

Upravo time se ovaj završni rad bavi, postupkom digitalizacije na prvi pogled jednostavnog procesa rezervacije autobusnih karata. Postoje razne platforme koje omogućuju brzu i jednostavnu izradu funkcionalnih aplikacija, a u ovome radu fokus je na platformi Oracle APEX 23.2.

Oracle APEX 23.2 je platforma za razvoj aplikacija niskog koda tvrtke Oracle Corporation. Koristi se za razvoj i implementaciju cloud, mobilnih i desktop aplikacija. APEX je u principu razvojno okruženje bazirano na webu s nizom značajki koje pojednostavljaju razvoj sigurnih i skalabilnih poslovnih aplikacija. Pojednostavljuje proces učitavanja podataka, upravljanja bazom podataka, razvoj REST (engl. Representational State Transfer) sučelja i izradu aplikacija visokih performansi koje je moguće koristiti putem stolnih računala i mobilnih uređaja.

Oracle APEX 23.2 koristi SQL na način da sve što se može izraziti SQL jezikom može se lako upotrijebiti u Oracle APEX aplikaciji. Također nudi gotove komponente pomoću kojih je moguće upravljati podacima, vizualizirati podatke te razvijati responzivne aplikacije. Umjesto samostalnog pisanja programskog koda, razvoj aplikacije unutar platforme Oracle APEX temelji se na korištenju inteligentnih čarobnjaka koji programera softvera vode kroz brzo stvaranje aplikacija i komponenti (Oracle APEX, bez dat. a).

Kao što je prethodno navedeno, cilj je rada izrada aplikacije za rezervaciju autobusnih karata što omogućuje efikasnije poslovanje i bolje korisničko iskustvo prilikom kupnje autobusnih karata. Najpotrebnije operacije koje će krajnji korisnici zahtijevati su CRUD (engl. Create, Read, Update, Delete) operacije, čiju implementaciju Oracle APEX uveliko pojednostavljuje. Aplikacija za rezervaciju autobusnih karata svojim korisnicima će omogućiti brojne funkcionalnosti, a one glavne su:

- registracija i prijava korisnika,
- prikaz rasporeda autobusa i cijena karata,
- pretraga autobusnih prijevoza po polazištu i odredištu i po datumu polaska,
- pregled detalja autobusa (tip, kapacitet, sadržaji),
- generiranje potvrda o rezervaciji,
- mogućnost otkazivanja rezervacije,
- za administratore sustava, mogućnost upravljanja autobusnim rutama.

2. Korisničke priče

Kako bi se zahtjevi aplikacije jasno definirali i shvatili, vrlo su nam korisne korisničke priče.

Korisničke priče su tehnika u agilnom razvoju softvera koja se koristi za opisivanje funkcionalnosti ili zahtjeva koje korisnici imaju u vezi sa softverskim proizvodom. Tehnika se fokusira na razumijevanje potreba korisnika i na kreiranje rješenja s ciljem zadovoljavanja tih potreba.

Za stvaranje korisničkih priča u obzir uzimamo 3 važna faktora (Čandrić, 2023a):

1. Uloga: predstavlja osobu, sustav, podsustav ili bilo koji drugi subjekt koji će komunicirati sa sustavom koji će se implementirati radi postizanja cilja (dobit će vrijednost u interakciji sa sustavom).
2. Akcija: predstavlja korisnikovo očekivanje koje se može postići interakcijom sa sustavom.
3. Korist: predstavlja vrijednost koja stoji iza interakcije sa sustavom.

Korisničke priče vrijedi koristiti upravo iz razloga što one naglašavaju verbalnu komunikaciju, lako su razumljive i korisnicima i programerima, mogu se koristiti za planiranje iteracija te su se pokazale korisnima unutar procesa iterativnog razvoja softvera (Cohn, 2009).

Zapis korisničke priče je kratak i jednostavan, iskazan poslovnim, a ne tehničkim jezikom. To je opis budućih značajki softvera na prirodnom jeziku, ispričan iz perspektive korisnika ili kupca (Čandrić, 2023a). Predložak je u sljedećem formatu:

Kao <vrsta korisnika> želim <neki cilj> [tako da <neki razlog>]

Za aplikaciju koja se bavi rezervacijom autobusnih karata, neke od korisničkih priča bi bile:

1. Kao potencijalni korisnik želim se registrirati na platformi tako da mogu pristupiti svojem korisničkom računu i koristiti sve funkcionalnosti aplikacije.
2. Kao putnik želim vidjeti raspored autobusa i cijene karata kako bih mogao planirati svoje putovanje i odabrati najpovoljniju opciju.
3. Kao putnik želim pretraživati autobusne rute po polazištu i odredištu kako bih brzo pronašao željenu rutu i informacije o dostupnim terminima.
4. Kao putnik želim pregledati detalje autobusa (tip, kapacitet, sadržaj) kako bih znao što mogu očekivati tijekom putovanja.
5. Kao putnik želim filtrirati putovanja prema datumu, vremenu polaska i dolaska i kako bih pronašao najbolju opciju za svoje potrebe.
6. Kao putnik želim dobiti potvrdu o rezervaciji kako bih imao dokaz o svojoj kupnji.
7. Kao putnik želim imati mogućnost otkazivanja rezervacije kako bih se prilagodio promjenama u planovima i situacijama.
8. Kao administrator želim upravljati autobusnim rutama kako bih mogao ažurirati informacije, dodavati nove rute ili mijenjati postojeće.

3. Korisničke persone

Modeli korisnika ili persone su arhetipovi korisnika, predstavnici različitih skupina ponašanja, stavova, sklonosti, ciljeva i motivacija koje su uočene i identificirane tijekom faze istraživanja. Njihovom upotrebom programerima i menadžerima je lakše razumjeti dizajn i dati prioritet određenim karakteristikama na temelju potreba korisnika (Čandrlić, 2023b).

Da bismo kreirali bilo kakvu uslugu ili proizvod, moramo znati za koga tu uslugu ili proizvod kreiramo, tko su naši korisnici, kako se ponašaju i zašto, što ih motivira, što ih frustrira, što su njihove ključne potrebe i želje (Pranjić, 2019).

Pet značajki dobro definiranih persona (Pranjić, 2019):

1. Predstavljaju glavne skupine vaših korisnika.
2. Opisuju osobe sa stvarnim vrijednostima i ciljevima.
3. Izražavaju glavne potrebe i očekivanja vaših glavnih skupina korisnika.
4. Daju jasnu sliku očekivanja korisnika i kakvu vrstu interakcije očekuju da će imati s vama.
5. Pružaju pomoć pri definiranju glavnih značajki i funkcionalnosti usluge.

Fokus je potrebno postaviti na prave korisnike, odnosno na one čije potrebe najbolje predstavljaju veći skup ključnih korisnika (Čandrlić, 2023b).

Budući da je aplikacija za rezervaciju autobusnih karata namijenjena širokom spektru potencijalnih korisnika, ključno je definirati nekoliko korisničkih persona čije će potrebe i želje kreatora aplikacije uveliko usmjeriti na ono što je doista važno, odnosno na što se treba fokusirati.

Na slikama 1., 2. i 3. prikazani su modeli korisnika koji bi koristili aplikaciju za rezervaciju autobusnih karata.

Marko Barić

Scenariji korištenja aplikacije

- Koristi aplikaciju za nadgledanje performansi i brzo identifikiranje eventualnih problema u sustavu.
- Koristi aplikaciju za ažuriranje i održavanje baze podataka autobusnih ruta, vremena polazaka i dostupnih sjedala.

Važnost kod aplikacije

| | |
|-------------------|------------|
| Dizajn | ██████████ |
| Lakoća korištenja | ██████████ |
| Interaktivnost | ██████████ |

Ciljevi

- Optimizacija performansi aplikacije radi poboljšanja brzine i učinkovitosti korisničkog iskustva.
- Implementacija sigurnosnih mjera radi zaštite korisničkih podataka i osiguranja povjerenja korisnika.
- Poboljšanje korisničkog sučelja i funkcionalnosti aplikacije radi lakšeg i intuitivnijeg korištenja.
- Brzo identifikiranje i rješavanje bilo kakvih tehničkih problema ili prekića u radu aplikacije.

Biografija

Kristijan Novak, 35-godišnji administrator aplikacije za rezervaciju autobusa iz Splita, odgovoran je za održavanje i optimizaciju platforme. Fokusiran je na osiguravanje brzine, stabilnosti i sigurnosti korisničkog iskustva. Potreban mu je alat koji omogućuje efikasno nadgledanje performansi aplikacije, brzo rješavanje potencijalnih problema i optimizaciju korisničkog sučelja.

Godine: **33**
Posao: **Administrator sustava**
Obitelj: **Samac**
Lokacija: **Zagreb, Hrvatska**

Slika 1 - Persona Marko Barić

Marko Horvat



Godine: **35**
Posao: **Menadžer prodaje**
Obitelj: **Oženjen, dvoje djece**
Lokacija: **Zagreb, Hrvatska**

Scenariji korištenja aplikacije

- Koristi aplikaciju za pretragu autobusnih ruta, usporedbu cijena i rezervaciju karte
- Koristi aplikaciju za brzu izmjenu rezervacije nakon neplaniranih promjena putovanja
- Koristi aplikaciju za brzo otkazivanje rezervacije zbog nepredviđenih obiteljskih obveza.

Važnost kod aplikacije

| | |
|-------------------|------------|
| Dizajn | ██████████ |
| Lakoća korištenja | ██████████ |
| Interaktivnost | ██████████ |

Biografija

Marko Horvat rođen je 15. svibnja 1989. godine u Zagrebu. Diplomirao je na Ekonomskom fakultetu Sveučilišta u Zagrebu i trenutno je menadžer prodaje u međunarodnoj IT kompaniji. Oženjen je i otac je dvoje dece, a živi i radi u Zagrebu. Zbog svoje pozicije menadžera prodaje, Marko često putuje unutar zemlje i u susjedne zemlje radi sastanaka, konferencija i pregovora. Stoga mu je važno da ponude pouzdanu i efikasnu uslugu za rezervaciju autobusnih karata koja mu omogućava brzo i jednostavno planiranje i prilagodbu putovanja poslovnim obavezama i promjenama u rasporedu.

Ciljevi

- Planira poslovna putovanja unutar zemlje i u susjedne zemlje
- Želi brzo i jednostavno pronaći autobusne rute, usporediti cijene i napraviti rezervaciju.
- Voli organizirati putovanja unaprijed kako bi se prilagodio svojem radnom rasporedu i obiteljskim obavezama.

Slika 2 - Persona Marko Horvat

Veronika Kovač



Godine: **21**
Posao: **Studentica kineziologije**
Obitelj: **Samac**
Lokacija: **Zagreb, Hrvatska**

Scenariji korištenja aplikacije

- Koristi aplikaciju za pretragu autobusnih ruta i usporedbu cijena kako bi pronašla najpovoljniju opciju za putovanje kući u svoj rodni grad.
- Koristi aplikaciju za rezervaciju autobusne karte unaprijed za sportska natjecanja, seminare ili druge posebne događaje kako bi osigurala mjesto i izbjegla gužve.

Važnost kod aplikacije

| | |
|-------------------|------------|
| Dizajn | ██████████ |
| Lakoća korištenja | ██████████ |
| Interaktivnost | ██████████ |

Biografija

Veronika Kovač je 21-godišnja studentica kineziologije koja živi i studira u Zagrebu. Kao mlada i ambiciozna studentica, Veronika često putuje između Zagreba i svog rodnog grada kako bi provodila vikende s obitelji i prijateljima te sudjelovala u raznim sportskim natjecanjima i seminarima. Budući da je samac i živi na studentskom budžetu, Veronika cijeni ekonomične i pouzdane načine putovanja koji joj omogućuju da optimalno iskoristi svoje vrijeme i novac.

Ciljevi

- Cilj je koristiti aplikaciju kako bi brzo i učinkovito planirala svoja putovanja između Zagreba i svog rodnog grada, s minimalnim gubicima vremena i stresa.
- Cilj je iskoristiti aplikaciju za pronalaženje najpovoljnijih cijena karata i mogućih popusta, kako bi optimizirala svoj studentski budžet i uštedjela novac na putovanjima.
- Cilj je imati mogućnost brze i jednostavne izmjene ili otkazivanja rezervacija u slučaju promjena u rasporedu ili neočekivanih situacija.

Slika 3 - Persona Veronika Kovač

4. Modeliranje podataka

Model podataka skup je međusobno povezanih podataka koji opisuju entitete, veze i atribute poslovnog sustava. On je reprezentacija skupa podataka koji se modelom interpretiraju preko aspekta: strukture, ograničenja i operatora (Pavlič, 2011).

Tijekom razvoja informacijskog sustava potrebno je izraditi adekvatan model podataka. Time će se spriječiti negativne posljedice loše organizacije podataka od kojih su neke (Panev, Kaluža, 2022):

- Troškovi i vrijeme razvoja informacijskog sustava premašuju planove ako ne postoji unaprijed izrađena dokumentacija s prikazanim modelom podataka na temelju kojeg će se izgraditi budući informacijski sustav.
- Bez dokumentiranog modela podataka informacijskog sustava, njegovo održavanje je mnogo složenije/zahajevnije te postaje dominantan posao inženjera.
- Promjene u poslovnom sustavu sporije se i teže implementiraju ako ne postoji dokumentacija s modelom podataka na kojem je vidljiva organizacija njegova informacijskog sustava.
- Bez dokumentacije s modelom podataka, zbog loše organizacije podataka, može se dogoditi nekontrolirana redundancija podataka i problemi u njihovu ažuriranju.

Metoda za modeliranje podataka je definirani postupak nalaženja i prikazivanja informacijskih objekata i njihovog međusobnog odnosa (Pavlič, 2011).

Postoji više specijaliziranih metodologija za razvoj informacijskog sustava, a neke od njih su:





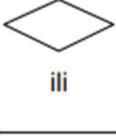
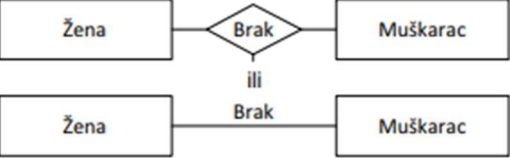






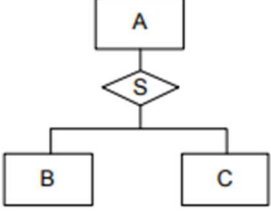
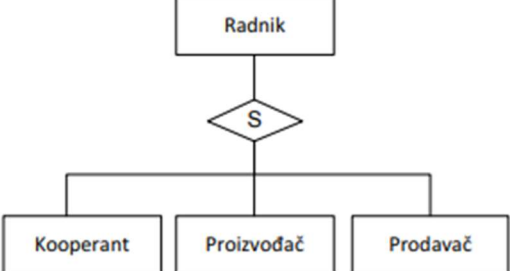
- SSADM metodologija
- Specijalizirana metodologija MIRIS
- Metodologija IDEA
- Metodologija ORACLE*CASE
- Metodologija IDEF
- Systems Development Life Cycle
- Business Systems Planning
- Information Engineering Methodology
- Agile Software Development
- Extreme Programming

Za modeliranje podataka informacijskog sustava namijenjenog za rezervaciju autobusnih karata korištena je specijalizirana metodologija MIRIS koja je opisana u nastavku. Metodologija MIRIS je odabrana upravo radi svoje jednostavnosti i intuitivnosti, odnosno

konceptata koji su bliski „prirodnom jeziku“ koji uveliko olakšavaju modeliranje baze podataka i sami razvoj softvera.

4.1. Specijalizirana metodologija MIRIS

Model podataka metodom entiteti-veze (EV) gradi se upotrebom grafičkih simbola. Na slici 4 dani su osnovni koncepti za gradnju dijagrama entiteti-veze (DEV) prema metodologiji MIRIS, a svaki koncept dijagrama prikazan je sebi svojstvenim oblikom (simbolom) te su prikazani konkretni primjeri za svaki od tih konceptata.

| KONCEPT | SIMBOL | PRIMJER |
|-------------------|---|--|
| TIP ENTITETA |  |  |
| SLAB TIP ENTITETA |  |  |
| TIP VEZE |  |  |
| ATRIBUT |  |  |
| AGREGACIJA |  |  |
| POVRATNA VEZA |  |  |
| GENERALIZACIJA |  |  |

Slika 4 - Grafički oblik konceptata strukture EV metode (Pavlič, 2011.)

Struktura modela podataka skup je entiteta i veza koji interpretira podatke klasificirajući ih u tipove entiteta i tipove veza među tipovima entiteta, zajedno s relevantnim svojstvima tipova entiteta (Pavlič, 2011).

Ograničenja modela podataka koncepti su strukture podataka, a omogućuju daljnju interpretaciju podataka razdvajajući dopuštena od zabranjenih stanja skupa podataka preko: dopuštenih podataka u okviru jednoga tipa entiteta, dopuštenih vrijednosti podataka pojedinoga svojstva tipa entiteta, dopuštenih povezivanja među tipovima entiteta (Pavlič, 2011).

Operatori modela podataka čine skup koncepata koji omogućuju interpretaciju dinamičkih karakteristika skupa podataka. Koncept strukture i ograničenja reprezentira statička svojstva poslovnoga sustava, a operatori omogućuju izmjenu stanja podataka u bazi podataka u skladu s promjenom stanja u poslovnome sustavu (Pavlič, 2011).

Struktura modela podataka gradi se od temeljnih koncepata: entitet, veza i atribut (Pavlič, 2011).

Entitet sustava je neki njegov realni ili konceptualni element, to je neka posebnost što u poslovnome sustavu postoji i jasno se razlikuje od drugih entiteta. Na primjer: muškarci, narudžbe, organizacijske jedinice i dr. (Pavlič, 2011).

Tip entiteta je skup sličnih pojedinačnih entiteta dobiven procesom klasifikacijske apstrakcije (Pavlič, 2011).

Slab tip entiteta je tip entiteta koji je na neki način ovisan o nekom drugom tipu entiteta, a ta se ovisnost prikazuje specijalnim tipom veze među jakim i slabim tipom entiteta. (Pavlič, 2011).

Veza je koncept koji predstavlja neku interakciju među entitetima u sustavu, odnosno predstavlja znanje o njihovoj povezanosti. Na primjer: brak je veza između muškaraca i žena (Pavlič, 2011).

Tip veze je skup veza između istih tipova entiteta (Pavlič, 2011).

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).

Atribut entiteta je neko svojstvo entiteta. Na primjer: ime radnika, datum prispjeća narudžbe, faktor uvećanja, brzina vozila, šifra općine i dr. (Pavlič, 2011).

U modelu podataka, kao i u nizu drugih modela (matematičkome sustavu jednadžbi putanje rakete, geografskoj karti teritorija, nacrtu broda, shemi električnih strujnih krugova pojačala) koji predstavljaju neki sustav, ne opisuje se potpuni skup znanja o sustavu, već se obavlja odabir relevantnih karakteristika sustava te se u model uključuju samo relevantni podaci (Pavlič, 2011).

Relevantni podatak (do koga nam je stalo, od interesa) je onaj podatak za koga smo zainteresirani te ulazi u model sustava, odnosno onaj koji pri analizi podataka smatramo važnim s aspekta namjene informacijskoga sustava (Pavlič, 2011).

Ključ tipa entiteta je takav skup atributa koji neovisno o vremenu zadovoljava uvjet jedinstvenosti i uvjet neredundantnosti (Pavlič, 2011).

Primarni ključ je jedan od ključeva tipa entiteta koji je odabran zbog nekog značajnijeg razloga za zamjenika entiteta toga tipa (za ključ). Jedan od razloga je njegovo često korištenje. U EV metodi uvodimo samo jedan ključ za svaki tip entiteta i to primarni (glavni) ključ. Ovaj će ključni atribut postati osnova za fizičko zapisivanje podataka u bazu podataka (Pavlič, 2011).

Uz primarni ključ, moguće je sadržavati i vanjski ključ.

Ako neki skup atributa u promatranoj relaciji nije ključ, ali je ključ u nekoj drugoj relaciji u modelu, onda se on naziva vanjski ključ (Pavlič, 2011).

Ključne razlike između primarnog i vanjskog ključa (Brown, 2024):

- Ograničenje primarnog ključa je stupac koji jedinstveno identificira svaki red u tablici sustava za upravljanje relacijskom bazom podataka, dok je vanjski ključ stupac koji stvara odnos između dvije tablice.
- Primarni ključ nikada ne prihvaća null vrijednosti, dok vanjski ključ može prihvatiti više null vrijednosti.
- Moguće je imati samo jedan primarni ključ u tablici, dok je s druge strane moguće imati imati više vanjskih ključeva u tablici.
- Vrijednost primarnog ključa ne može se ukloniti iz nadređene tablice, dok se vrijednost vanjskog ključa može ukloniti iz podređene tablice.
- Dva retka ne mogu imati identične vrijednosti za primarni ključ, dok vrijedi obratno za vanjski ključ.

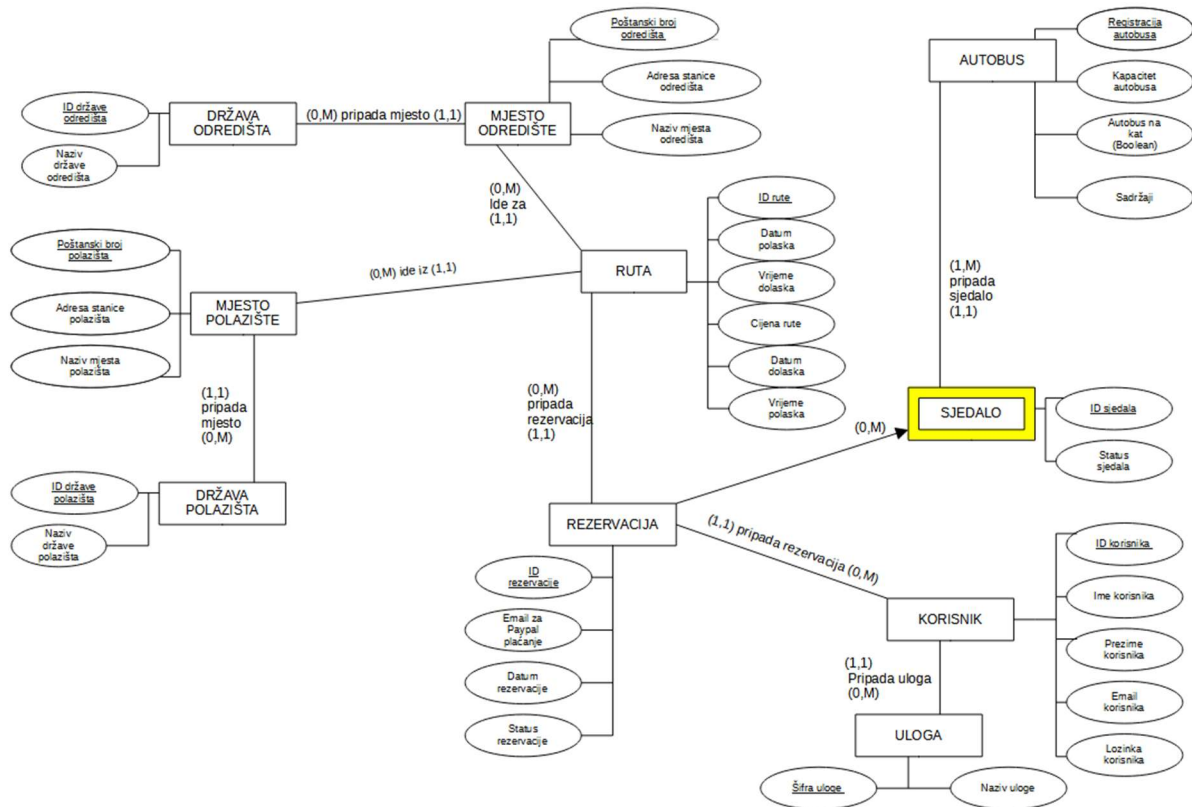
Projektiranje baze podataka predstavlja ključni dio njezina razvoja. Cilj projektiranja je da se na osnovi utvrđenih potreba za podacima oblikuje pogodna građa baze. Ta građa u pravilu ne bi smjela biti optimizirana za jednu određenu aplikaciju, već bi trebala održavati smisao i unutrašnju povezanost samih podataka. Na taj način, baza bi dugoročno trebala biti pogodna za promjene i evoluciju u skladu sa zahtjevima budućih aplikacija (Manger, 2010).

Imajući sve ovo u obzir, izuzetno je važno temeljito i pažljivo projektirati bazu podataka za aplikaciju namijenjenu za rezervaciju autobusnih karata. Baza podataka predstavlja srce i temelj same aplikacije, pružajući infrastrukturu za pohranu, upravljanje i pristup podacima koji su ključni za sve operacije aplikacije. Pri projektiranju baze podataka, nužno je uzeti u obzir sve potrebne funkcionalnosti aplikacije, uključujući stvaranje, ažuriranje, brisanje i pretraživanje podataka o korisnicima, autobusima, rasporedima vožnje, cijenama karata te informacijama o rezervacijama. Također, važno je osigurati da baza podržava složene operacije radi brzog i pouzdanog rada. Stoga, detaljno planiranje i projektiranje baze podataka ključno je za uspjeh aplikacije za rezervaciju autobusnih karata, jer će kvalitetno dizajnirana baza

podataka omogućiti stabilno i efikasno funkcioniranje aplikacije te pružiti osnovu za njezinu buduću evoluciju i skaliranje.

4.2. Dijagram entiteti-veze

Na slici 5. prikazan je dijagram entiteti-veze (DEV) za sustav rezervacije autobusnih karata.



Slika 5 - Dijagram entiteti-veze za sustav rezervacije autobusnih karata

U dijagramu entiteti-veze za sustav namijenjen za rezervaciju autobusnih karata nalazi se 10 tipova entiteta, od kojih je:

- Devet jakih tipova entiteta: DRŽAVA ODREDIŠTA, DRŽAVA POLAZIŠTA, MJESTO ODREDIŠTE, MJESTO POLAZIŠTE, RUTA, AUTOBUS, REZERVACIJA, KORISNIK i ULOGA
- Jedan slabi tip entiteta: SJEDALO

Tip entiteta DRŽAVA POLAZIŠTA sadrži informacije o državi iz koje polazi autobus. Atribut "ID države polazišta" je jedinstveni identifikator za svaku državu polazišta. Atribut „Naziv države polazišta" sadrži ime te države.

Tip entiteta DRŽAVA ODREDIŠTA sadrži informacije o državi u koju autobus dolazi. Atribut „ID države odredišta" je jedinstveni identifikator za svaku državu odredišta. Atribut „Naziv države odredišta" sadrži ime te države.

Tip entiteta MJESTO POLAZIŠTE sadrži podatke o mjestu polaska autobusa. Atribut „Poštanski broj polazišta" predstavlja poštanski broj mjesta polaska. Atribut „Adresa stanice polazišta" sadrži adresu autobusne stanice. Atribut „Naziv mjesta polazišta" sadrži ime mjesta polaska.

Tip entiteta MJESTO ODREDIŠTE sadrži podatke o mjestu dolaska autobusa. Atribut „Poštanski broj odredišta" predstavlja poštanski broj mjesta dolaska. Atribut „Adresa stanice odredišta" sadrži adresu autobusne stanice. Atribut „Naziv mjesta odredišta" sadrži ime mjesta dolaska.

Tip entiteta AUTOBUS opisuje autobuse u sustavu. Atribut „Registracija autobusa" je jedinstveni identifikator autobusa. Atribut „Naziv autobusa" sadrži naziv ili model autobusa. Atribut „Kapacitet autobusa" specificira broj sjedišta u autobusu. Atribut „Autobus na kat" označava je li autobus na kat. Atribut „Sadržaji" opisuje dodatne sadržaje u autobusu, poput WiFi-a, klima uređaja i sl.

Slabi tip entiteta SJEDALO sadrži informacije o sjedalima unutar autobusa. Atributi „ID sjedala" i „ID rezervacije" čine jedinstveni identifikator svakog sjedala.

Tip entiteta RUTA sadrži podatke o rutama koje autobusi voze. Atribut „ID rute" je jedinstveni identifikator svake rute. Atribut „Datum polaska" označava datum kada ruta započinje. Atribut „Datum dolaska" označava datum kada ruta završava. Atribut „Vrijeme polaska" označava vrijeme kada autobus kreće. Atribut „Vrijeme dolaska" označava vrijeme kada autobus stiže na odredište. Atribut „Cijena rute" specificira cijenu karte za tu rutu.

Tip entiteta ULOGA definira različite uloge korisnika u sustavu. Atribut „Šifra uloge" je jedinstveni identifikator svake uloge. Atribut „Naziv uloge" sadrži naziv uloge, npr. administrator ili korisnik (putnik).

Tip entiteta KORISNIK sadrži podatke o korisnicima sustava. Atribut „ID korisnika" je jedinstveni identifikator svakog korisnika. Atribut „Ime korisnika" sadrži ime korisnika. Atribut „Prezime korisnika" sadrži prezime korisnika. Atribut „Email korisnika" sadrži email adresu korisnika. Atribut „Lozinka korisnika" sadrži šifriranu lozinku korisnika. Atribut „Šifra uloge" povezuje korisnika s njegovom ulogom u sustavu.

Tip entiteta REZERVACIJA sadrži podatke o rezervacijama koje su korisnici napravili. Atribut „ID rezervacije" je jedinstveni identifikator svake rezervacije. Atribut „Datum rezervacije" označava datum kada je rezervacija napravljena. Atribut „Email za Paypal plaćanje" sadrži email adresu korisnika za plaćanje putem Paypala. Atribut „Status rezervacije" označava status rezervacije (npr. potvrđena, otkazana). Atribut „Iznos rezervacije" odnosi se na cijenu rezervirane autobusne karte.

4.3. Relacijska shema

Kako bi se EV model preveo u relacijsku shemu, prate se sljedeća osnovna pravila (Pavlić, 2011):

1. Atribut u DEV-u (na primjer Ime radnika) odgovara konceptu Atribut u relacijskom modelu (RM).
2. Svaki tip entiteta iz DEV-a postaje relacija (tablica) u RM-u i to tako da atributi tipa entiteta postaju atributi relacije, primarni (glavni) ključ tipa entiteta postaje primarni (glavni) ključ relacije.
3. Svaka agregacija postaje relacija u RM-u, ključ relacije je složen od ključeva tipova entiteta koji sudjeluju u vezi. Atributi agregacije postaju atributi relacije.
4. Svaki tip veze s brojnostima (1,1):(0,M) (tip veze gdje je strana s brojnošću po gornjoj granici 1 totalna) ne prevodi se u relaciju, već se ključ tipa entiteta koji ulazi u vezu sa strane s brojnošću M umeće kao atribut (vanjski ključ) u relaciju dobivenu prevođenjem tipa entiteta s brojnošću (1,1). Tip veze ne smije imati attribute jer, ako ih ima, to nije tip veze već agregirani tip entiteta.
5. Tip slabog entiteta prevodi se u relaciju. Atributi tipa slabog entiteta postaju atributi relacije. Ključ relacije je složen i od ključa jakoga tipa entiteta i ključa slaboga tipa entiteta tako da jedinstveno identificira n-torke relacije u okviru relacijskoga modela.

Prateći pravila za prevođenje dobivena je sljedeća relacijska shema baze podataka:

DRŽAVA POLAZIŠTA (**ID države polazišta**, Naziv države polazišta)

DRŽAVA ODREDIŠTA (**ID države odredišta**, Naziv države odredišta)

MJESTO POLAZIŠTE (**Poštanski broj polazišta**, Adresa stanice polazišta, Naziv mjesta polazišta, *ID države polazišta*)

MJESTO ODREDIŠTE (**Poštanski broj odredišta**, Adresa stanice odredišta, Naziv mjesta odredišta, *ID države odredišta*)

AUTOBUS (**Registracija autobusa**, Naziv autobusa, Kapacitet autobusa, Autobus na kat, Sadržaji)

SJEDALO (**ID sjedala**, **ID rezervacije**, Status sjedala, *Registracija autobusa*)

RUTA (**ID rute**, Datum polaska, Datum dolaska, Vrijeme polaska, Vrijeme dolaska, Cijena rute, *Poštanski broj polazišta*, *Poštanski broj odredišta*)

ULOGA (**Šifra uloge**, Naziv uloge)

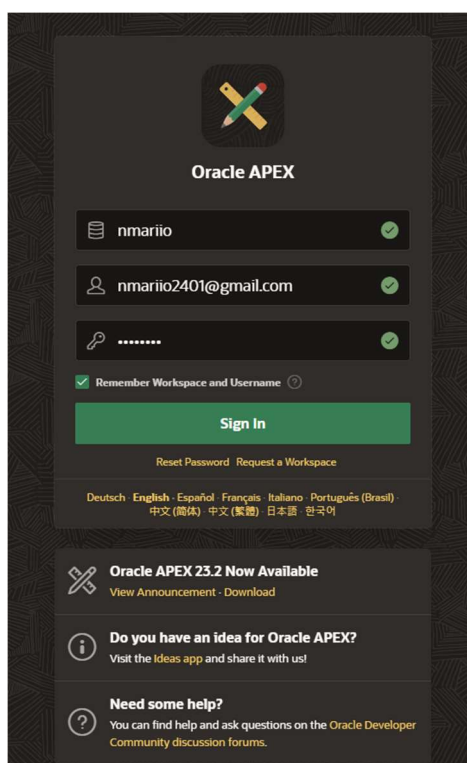
KORISNIK (**ID korisnika**, Ime korisnika, Prezime korisnika, Email korisnika, Lozinka korisnika, *Šifra uloge*)

REZERVACIJA (**ID rezervacije**, Datum rezervacije, Email za Paypal plaćanje, Status rezervacije, *ID rute*, *ID korisnika*)

5. Implementacija aplikacije u ORACLE Apex 23.2

U nastavku je dan kompletan postupak implementacije aplikacije za rezervaciju autobusnih karata unutar platforme Oracle Apex 23.2.

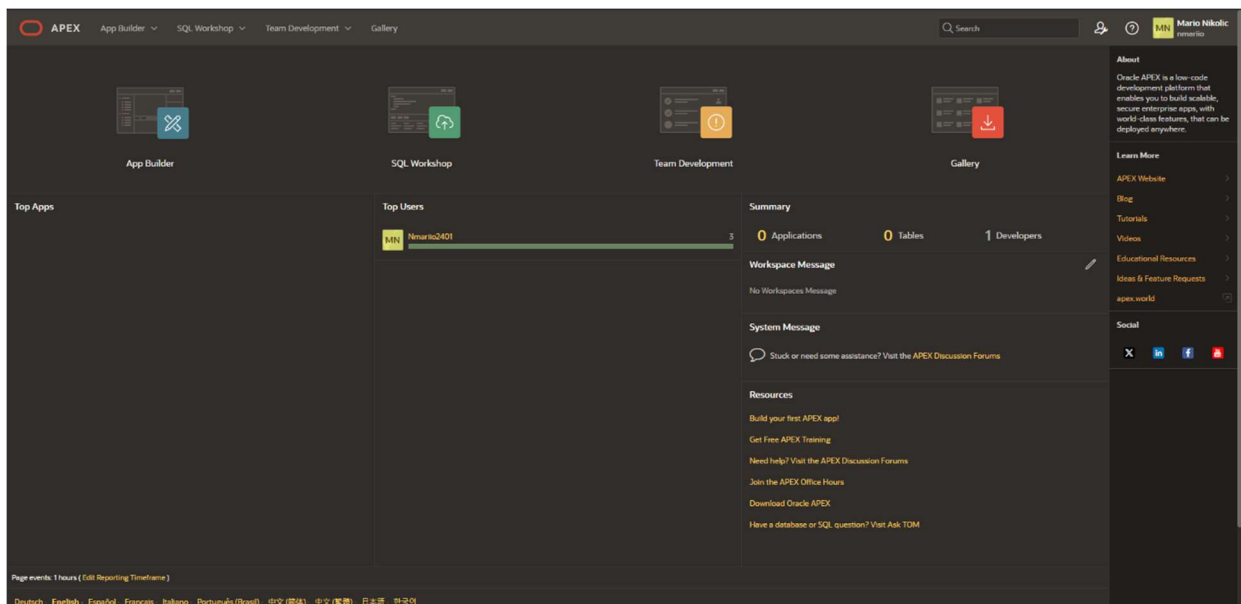
Kako bi se implementirala sama aplikacija, najprije je potrebno kreirati radno okruženje unutar Oracle APEX platforme, a nakon generiranja radnog okruženja potrebno se prijaviti sa svojim korisničkim podacima. Na Slici 6. prikazan je ekran za prijavu.



Slika 6 - Prijava u radno okruženje

Ako su korisnički podaci (naziv radnog okruženja, email korisnika i lozinka) ispravno uneseni, korisnika se preusmjerava na glavni ekran na kojemu su vidljive sve mogućnosti koje Oracle APEX nudi razvojnim inženjerima (Slika 7.) (Oracle APEX, bez dat. b):

- App Builder je centralno mjesto za kreiranje i upravljanje aplikacijama. Omogućava kreiranje stranica, izvještaja, formi i drugih komponenti aplikacije.
- SQL Workshop je alat za rad sa SQL i PL/SQL jezikom. Omogućava izvršavanje SQL upita, skripti te samo upravljanje bazom podataka. Također, omogućava i izradu objekata baze podataka kao što su tablice, indeksi i sl.
- Team Development je alat za kolaboraciju i upravljanje projektima unutar tima. Omogućuje praćenje zadataka, grešaka i funkcionalnosti aplikacija.
- App Gallery je biblioteka predložaka i uzoraka aplikacija. Omogućuje razvojnim inženjerima preuzimanje i prilagodbu postojeće aplikacije za svoje potrebe.



Slika 7 - Radno okruženje

Za potrebe izrade aplikacije za rezervaciju autobusnih karata korišteni su alati App Builder i SQL Workshop.

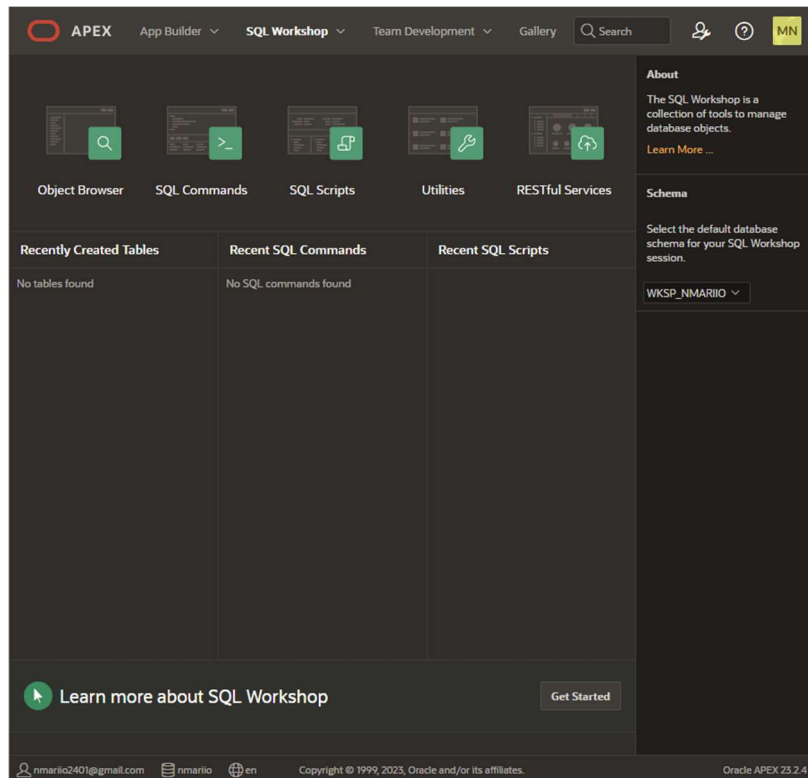
5.1. Izrada baze podataka (SQL Workshop)

Na temelju prethodno izrađene sheme relacijske baze podataka unutar alata SQL Workshop kreirana je baza podataka (koja je temelj same aplikacije). Budući da se radi o relacijskoj bazi podataka, SQL Workshop je prikladan za potrebe ove aplikacije jer omogućuje brzo i jednostavno kreiranje relacijske baze podataka pružajući programerima najbitnije mogućnosti, kao što su kreiranje i manipuliranje tablicama, pisanje SQL naredbi, izvršavanje upita i sl.

SQL Workshop sadrži sljedeće alate (Oracle APEX, bez dat. c):

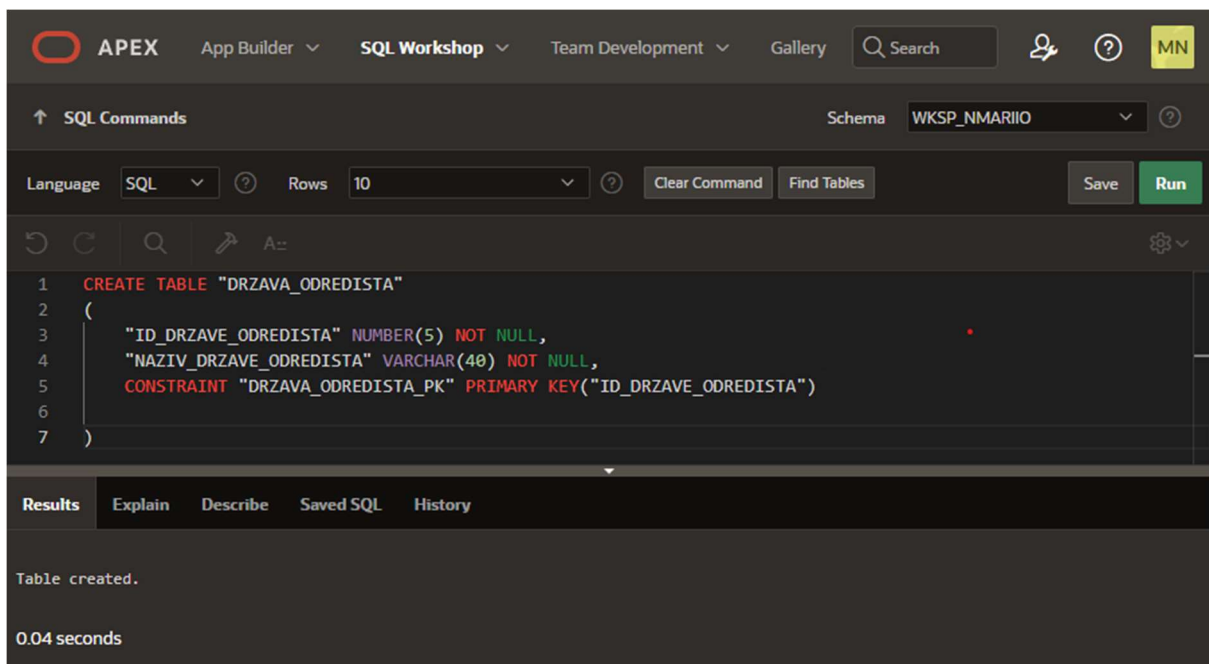
- Object Browser: omogućava pregled, kreiranje, izmjene i brisanje objekata baze podataka kao što su tablice, pogledi, indeksi, procedure itd.
- SQL Commands je interaktivni editor za izvršavanje SQL naredbi i upita. Omogućava pisanje, izvršavanje i pregled rezultata SQL upita. Također podržava formatiranje i pregled povijesti prethodno izvršenih upita za lakše praćenje i eventualnu ponovnu upotrebu.
- SQL Scripts je alat za kreiranje, izvršavanje i upravljanje SQL skriptama. Omogućava organizaciju skripti u okviru projekta ili nekih specifičnih zadataka. Osim toga, podržava i verzioniranje skripti i suradnju između korisnika.
- Utilities je skup pomoćnih alata koji olakšavaju administraciju i razvoj baze podataka.
- RESTful Services omogućava kreiranje i upravljanje RESTful web servisima za pristup bazi podataka.

Na slici 8. prikazan je ekran alata SQL Workshop.



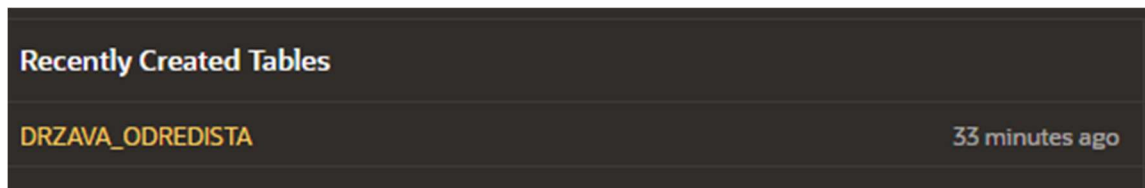
Slika 8 - SQL Workshop sučelje

Za potrebe kreiranja tablice DRZAVA_ODREDISTE pomoću sintakse SQL jezika, korišten je alat SQL Commands. Na slici 9. prikazan je postupak kreiranja prve tablice (DRZAVA_ODREDISTE) koja sadrži atribute ID_DRZAVE_ODREDISTA i NAZIV_DRZAVE_ODREDISTA koji imaju uvjet da ne mogu biti prazni (NOT NULL) te sadrži ograničenje na atribut ID_DRZAVE_ODREDISTA koji je primarni ključ tablice.



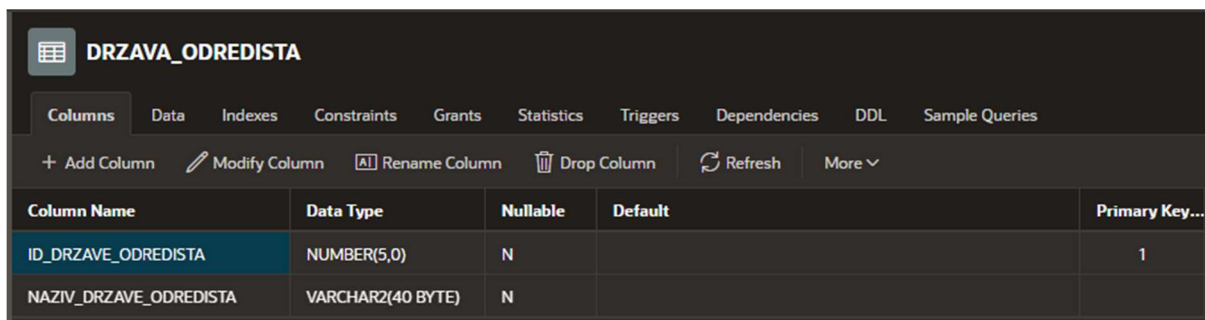
Slika 9 - Izrada tablice DRZAVA_ODREDISTA

Nakon kreiranja tablice, pod sekcijom „Recently Created Tables“ vidljivo je da je tablica kreirana, što je i prikazano na slici 10.



Slika 10 - Nedavno kreirane tablice

Klikom na tablicu otvara se ekran na kojemu su vidljivi stupci, indeksi, ograničenja itd. Na slici 11. prikazana su svojstva tablice DRZAVA_ODREDISTA.



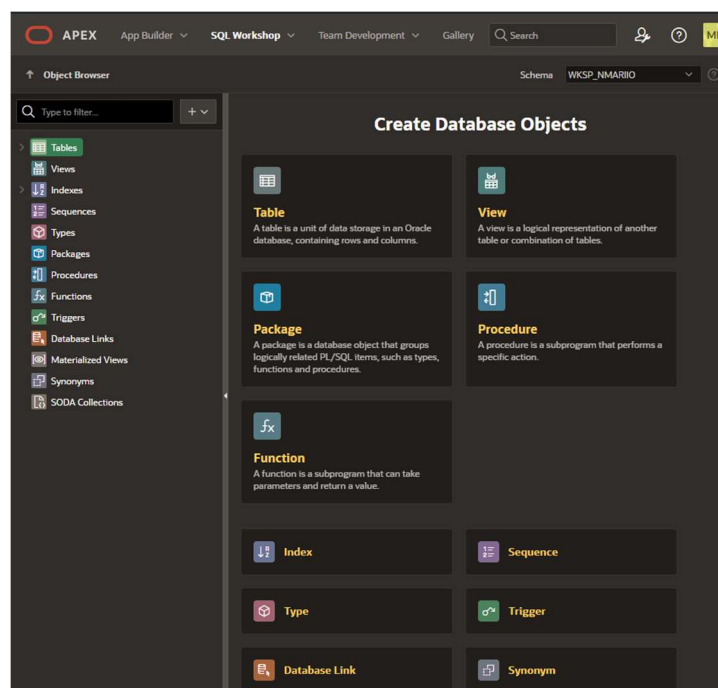
| Column Name | Data Type | Nullable | Default | Primary Key... |
|------------------------|-------------------|----------|---------|----------------|
| ID_DRZAVE_ODREDISTA | NUMBER(5,0) | N | | 1 |
| NAZIV_DRZAVE_ODREDISTA | VARCHAR2(40 BYTE) | N | | |

Slika 11 - Tablica DRZAVA_ODREDISTA

Na isti način kreirana je i tablica DRZAVA_POLAZISTA.

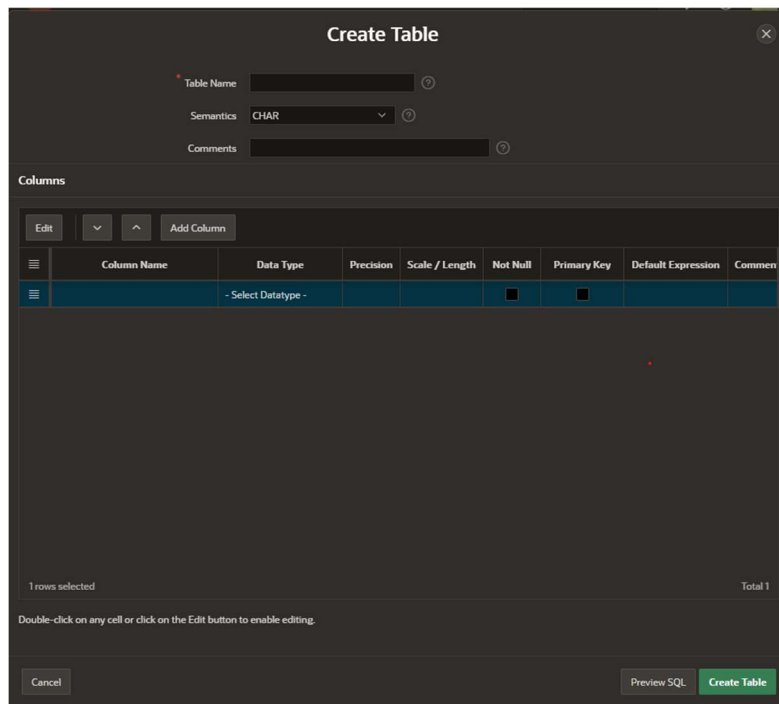
Osim alata SQL Commands, postoji i drugi način kreiranja tablica, unutar alata Object Browser, koji nudi grafički prikaz unutar kojega se mogu kreirati tablice (nije potrebno koristiti SQL sintaksu).

Na slici 12. prikazane su opcije koje alat Object Browser nudi razvojnim inženjerima.



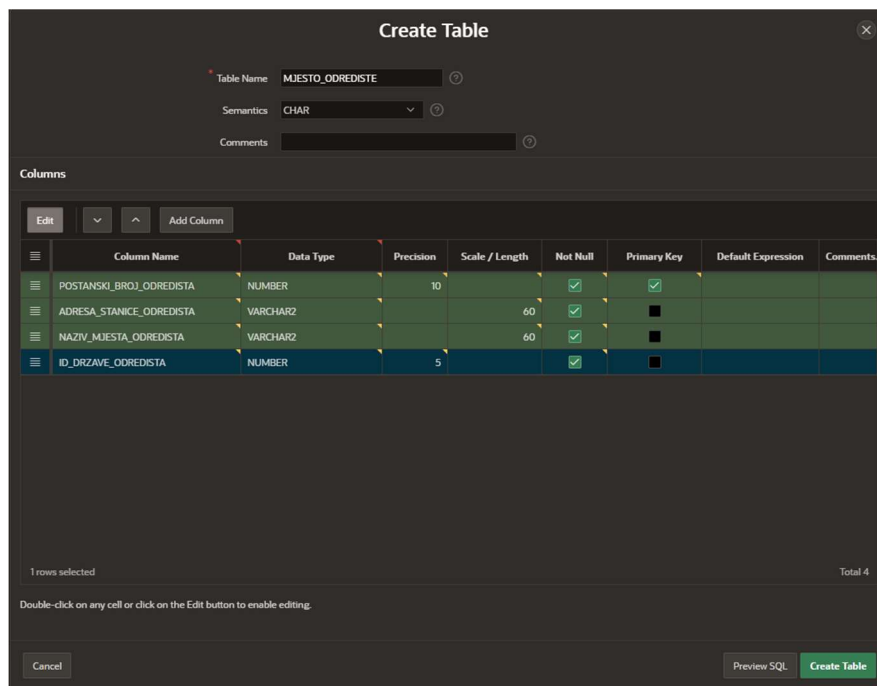
Slika 12 - Ekran alata Object Browser

Klikom na opciju Table otvara se ekran koji omogućuje kreiranje nove tablice, koji je vidljiv na slici 13.



Slika 13 - Ekran Create Table

Dakle, postupak kreiranja tablice na ovaj način vrlo je intuitivan jer je korisniku jasno označeno gdje se definira naziv tablice, gdje se definiraju stupci, gdje se postavljaju ograničenja (kao što su primarni ključ i sl). Zbog efikasnosti i jednostavnosti, putem ovog alata kreirane su sve preostale tablice baze podataka. Na slici 14. prikazan je postupak izrade tablice MJESTO_ODREDISTE.



Slika 14 - Izrada tablice MJESTO_ODREDISTE

Klikom na gumb „Create Table“ u donjem desnom kutu kreira se nova tablica sa svim definiranim svojstvima.

Klikom na „Constraints“ moguće je dodavati ograničenja na stupce tablice.

Budući da tablica MJESTO_ODREDISTE sadrži vanjski ključ (ID_DRZAVE_ODREDISTA) potrebno je postaviti ograničenje na taj stupac, a taj postupak vidljiv je na slici 15.

Create Constraint

Owner: WKSP_NMARIIO

Table: MJESTO_ODREDISTE

Name: ID_DRZ_ODR_MJESTO_ODR_FK

Constraint Type: Foreign Key

Delete Behavior: Disallow Delete
 Cascade Delete
 Set Null on Delete

Foreign Key Column(s): POSTANSKI_BROJ_ODREDISTA, ADRESA_STANICE_ODREDISTA, NAZIV_MJESTA_ODREDISTA

Reference Table Name: DRZAVA_ODREDISTA

Reference Table Column(s): NAZIV_DRZAVE_ODREDISTA

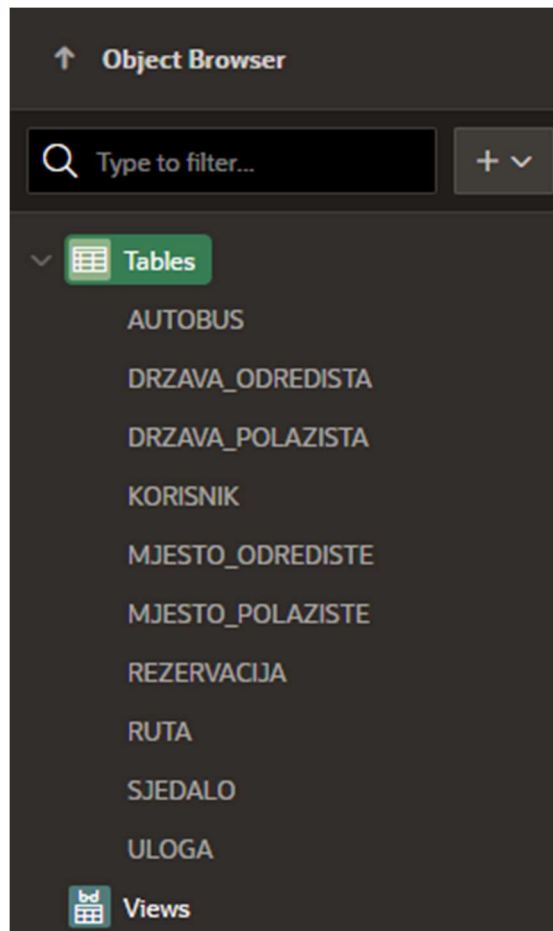
SQL

```
alter table "WKSP_NMARIIO"."MJESTO_ODREDISTE" add constraint "ID_DRZ_ODR_MJESTO_ODR_FK" foreign key ( "ID_DRZAVE_ODREDISTA" ) references "DRZAVA_ODREDISTA" ( "ID_DRZAVE_ODREDISTA" );
```

Slika 15 - Kreiranje ograničenja vanjski ključ unutar tablice MJESTO_ODREDISTE

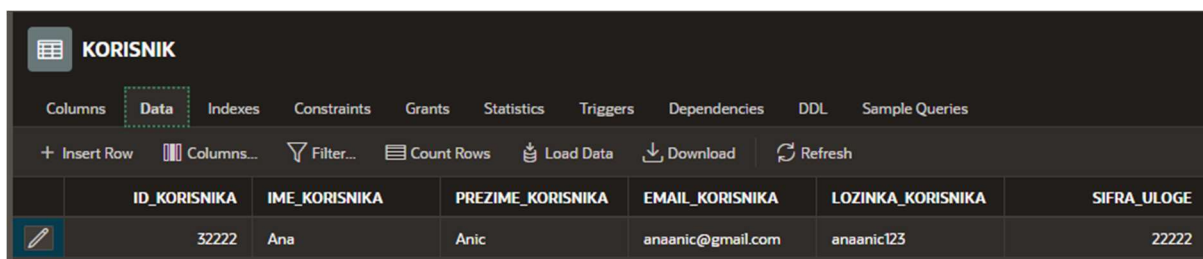
Na isti način izrađene su sve preostale tablice neophodne za ispravan rad naše aplikacije (MJESTO_POLAZISTE, RUTA, AUTOBUS, SJEDALO, REZERVACIJA, KORISNIK i ULOGA).


Na slici 16. moguće je vidjeti kako su tablice doista kreirane te se uvjeriti u njihovo postojanje unutar same baze podataka.



Slika 16 - Ekran sa svim kreiranim tablicama

Zbog provjere ispravnosti tablica, moguće je unositi podatke kako bismo provjerili jesu li stupci ispravno definirani. Na slici 17. moguće je vidjeti primjer unosa novog zapisa u tablicu KORISNIK s ciljem provjere ispravnosti. Provjera je napravljena za sve tablice s ciljem prevencije grešaka i problema vezanih uz tablice iz bazu podataka prilikom same izrade aplikacije.



| | ID_KORISNIKA | IME_KORISNIKA | PREZIME_KORISNIKA | EMAIL_KORISNIKA | LOZINKA_KORISNIKA | SIFRA_ULOGA |
|---|--------------|---------------|-------------------|-------------------|-------------------|-------------|
|  | 32222 | Ana | Anic | anaanic@gmail.com | anaanic123 | 22222 |

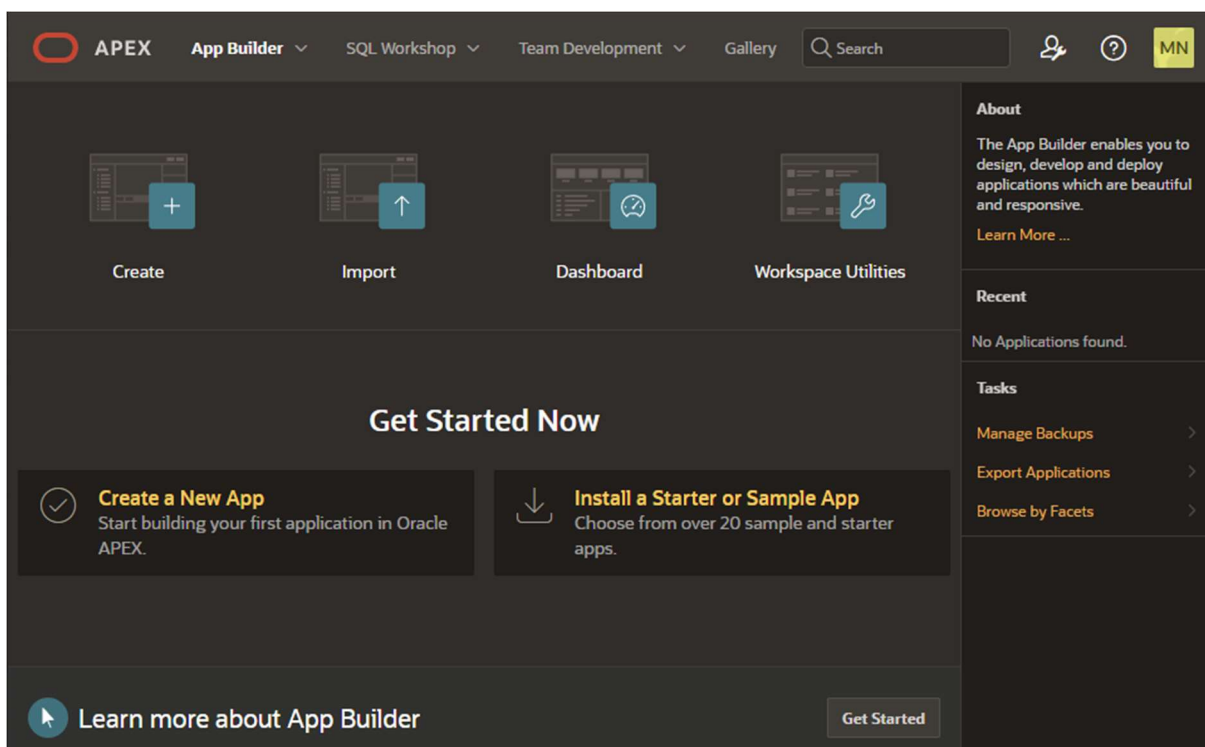
Slika 17 - Unos novog zapisa u tablicu KORISNIK

5.2. Izrada aplikacije (App Builder)

Nakon kreiranja svih tablica unutar alata *SQL Workshop* slijedi izrada same aplikacije.

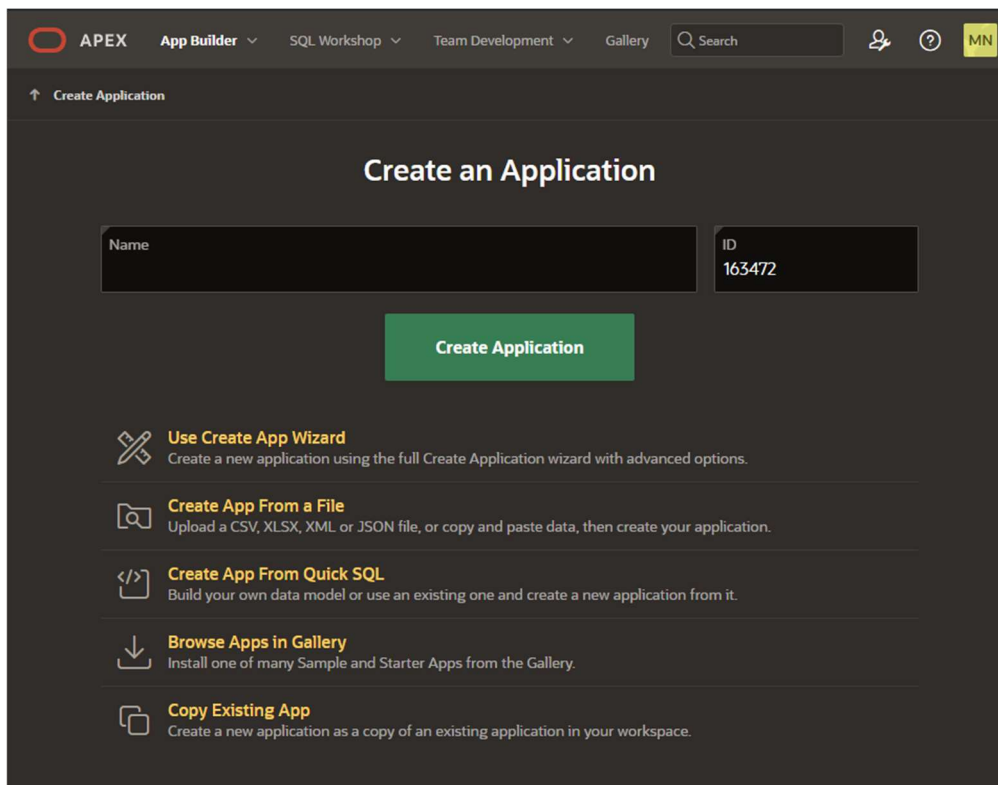
Aplikacija se kreira unutar alata App Builder. App Builder je centralno mjesto unutar Oracle APEX-a gdje korisnici mogu dizajnirati i razvijati aplikacije te upravljati istima. Pruža jednostavne vizualne alate za kreiranje različitih vrsta aplikacija, od jednostavnih izvještaja i obrazaca do složenih poslovnih aplikacija. Također pruža jednostavnu integraciju s Oracle bazama podataka za pristup i manipuliranje podacima. Podržava razvoj bez coda (engl. no code development) i s malo koda (engl. low code development), pružajući pritom širokom rasponu korisnika (od poslovnih analitičara pa sve do profesionalnih programera) mogućnost razvijanja aplikacija (Oracle APEX, bez dat. d).

Na slici 18. mogu se vidjeti sve komponente App Builder alata.



Slika 18 - App Builder sučelje

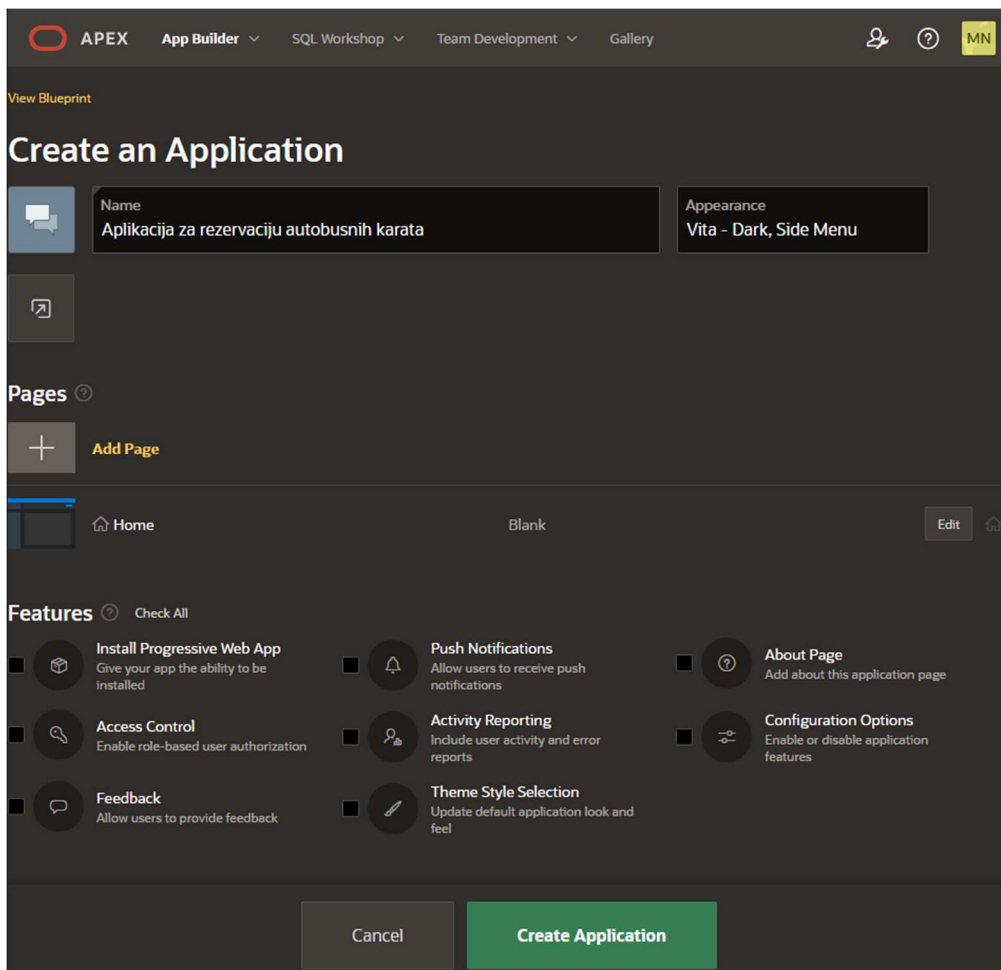
Klikom na opciju Create korisnika se preusmjerava na ekran unutar kojega može kreirati novu aplikaciju. Postoji više načina na koje se može kreirati nova aplikacija, što se može vidjeti na slici 19.



Slika 19 - Create an Application sučelje

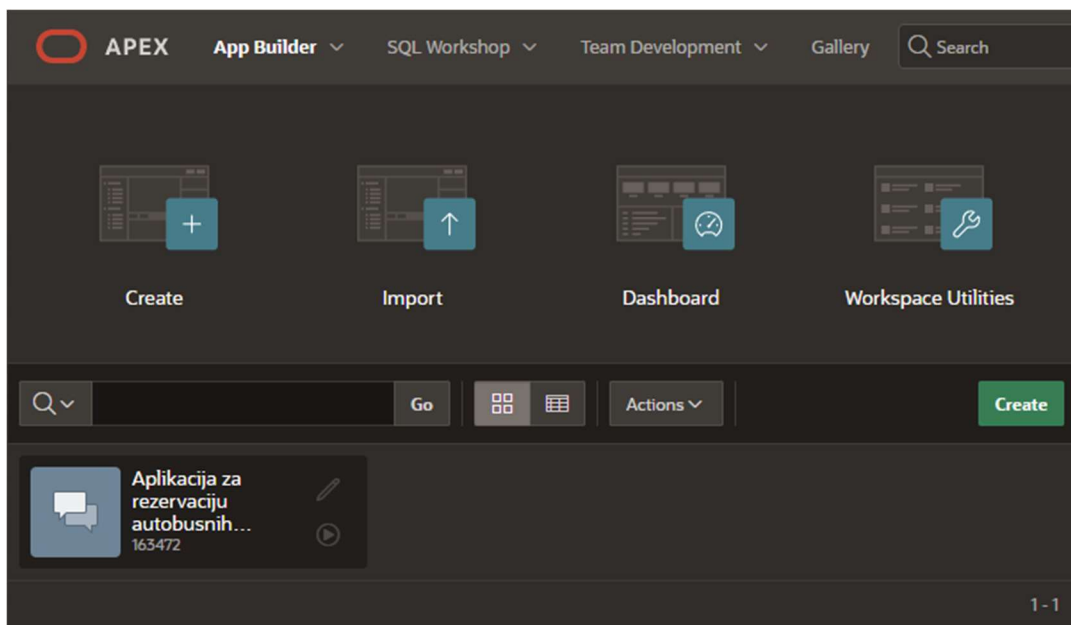
Za kreiranje aplikacije iskorištena je opcija „Use Create App Wizard“. Ta opcija korisnicima nudi intuitivan i strukturiran način za brzo kreiranje web aplikacija. Posebno je koristan zbog svoje jednostavnosti, fleksibilnosti i brzine, omogućavajući korisnicima svih razina tehničkog znanja kreiranje funkcionalne aplikacije bez potrebe za opsežnim programiranjem.

Na slici 20. prikazan je postupak izrade aplikacije za rezervaciju autobusnih karata.



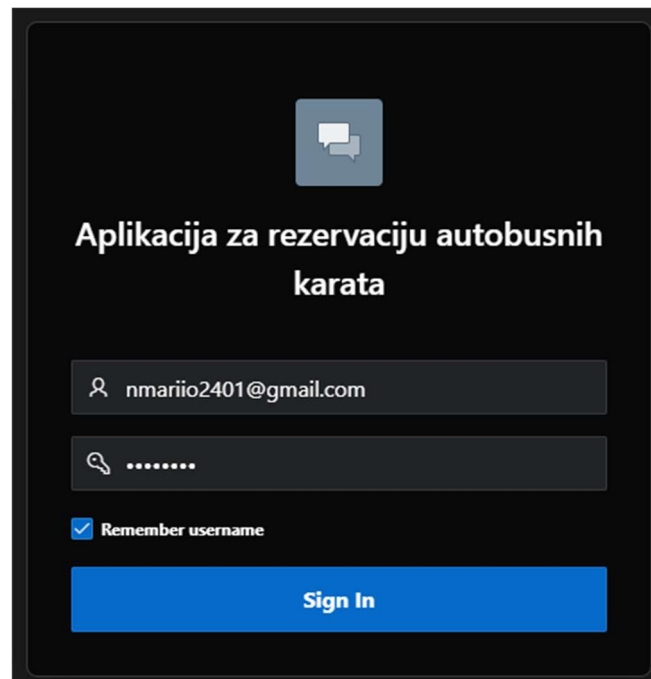
Slika 20 - Kreiranje aplikacije za rezervaciju autobusnih karata

Klikom na Create Application stvara se nova aplikacija, a potom se ista može pronaći unutar alata App Builder (kao što je prikazano na slici 21).



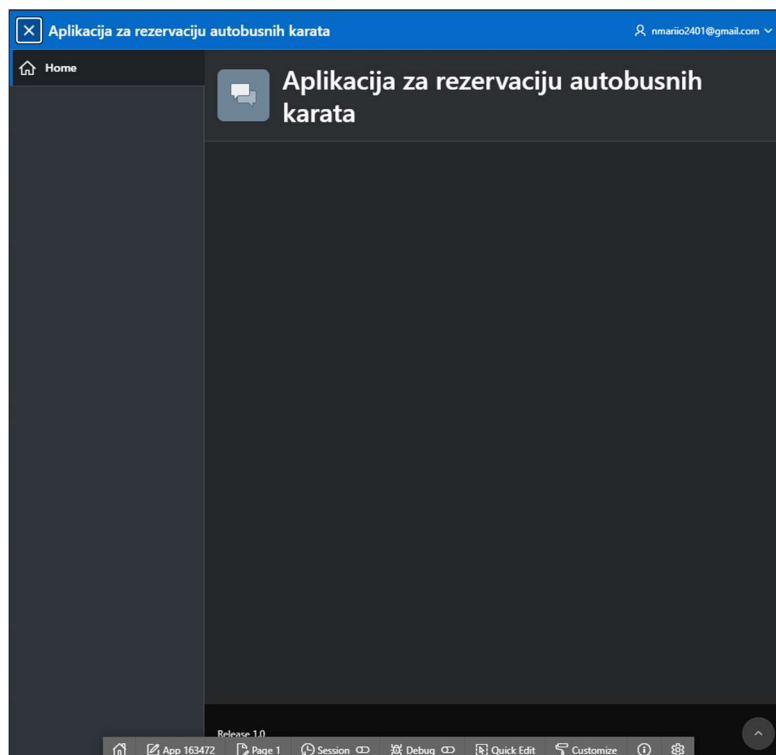
Slika 21 - Sučelje App Builder sa novo kreiranom aplikacijom

Klikom na strelicu koja simbolizira pokretanje aplikacije (run) aplikacija se pokreće i otvara u novom prozoru, što se može vidjeti na slici 22.



Slika 22 - Pokrenuta aplikacija

Nakon unosa korisničkih podataka (u ovom slučaju to je moguće samo za kreatora aplikacije jer drugi korisnici još nisu definirani), korisnika se preusmjerava na glavnu stranicu aplikacije (slika 23). Aplikacija za sada nema nikakve druge stranice niti mogućnosti, te je stoga prikazana samo Home stranica (koja se stvara prilikom izrade aplikacije).



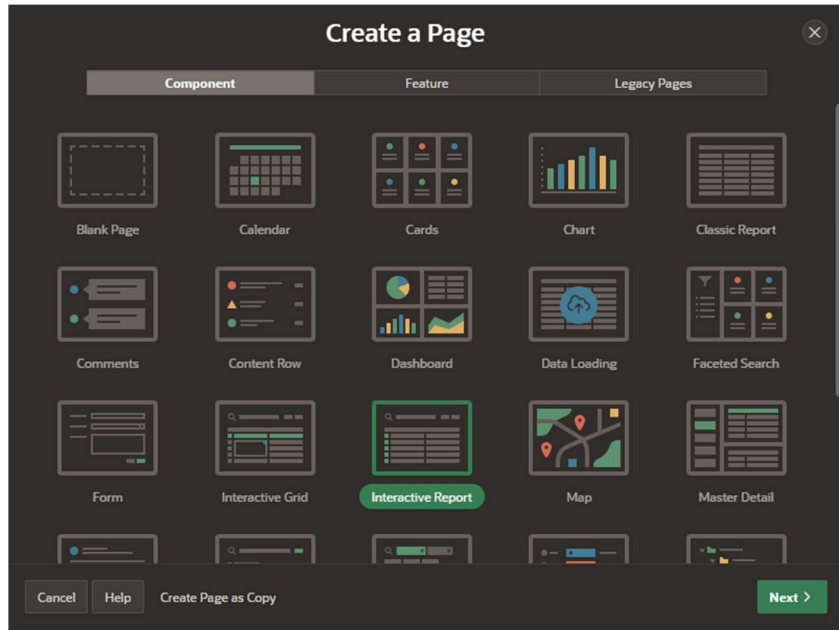
Slika 23 - Glavna stranica aplikacije

5.2.1. Izrada stranice „Popis ruta“

Sada kada je aplikacija kreirana moguće je dodavati nove stranice i značajke, odnosno nadograđivati samu aplikaciju. U nastavku je dan opis i prikaz kreiranja stranice s ciljem prikazivanja svih ruta iz baze podataka.

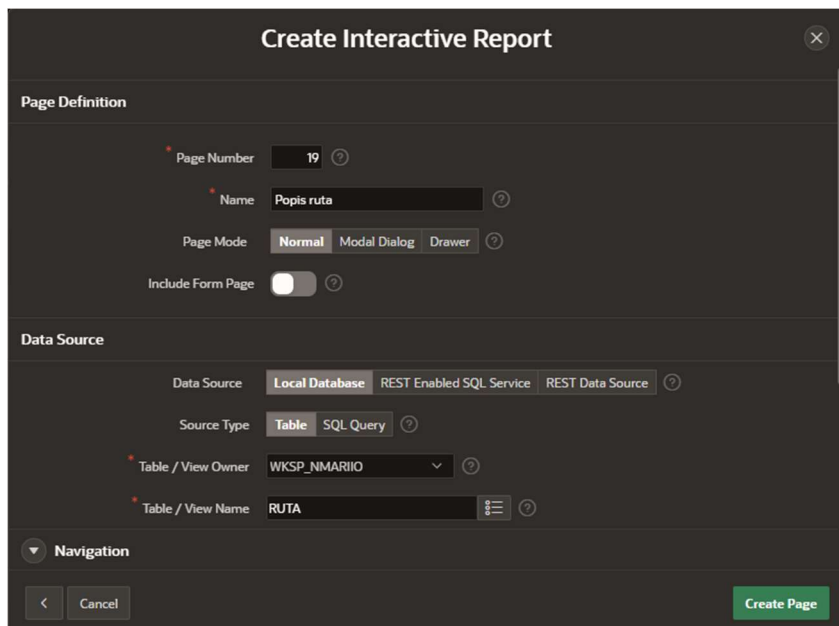
Klikom na Create Page otvara se ekran za kreiranje nove stranice.

Budući da je u specifikaciji zahtjeva jasno definirano da se rute mogu filtrirati po određenim značajkama, najprikladnije je odabrati Interactive Report, a na slici 24. može se vidjeti ekran kod stvaranja nove stranice.



Slika 24 - Stvaranje nove stranice tipa "Interactive Report"

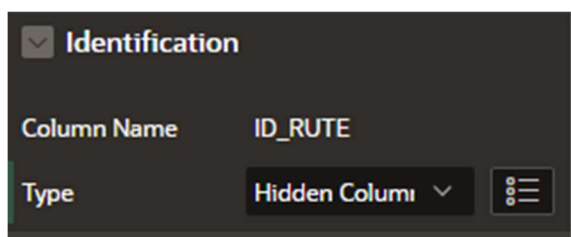
Na slici 25. moguće je vidjeti sve postavke koje se odnose na tu stranicu.



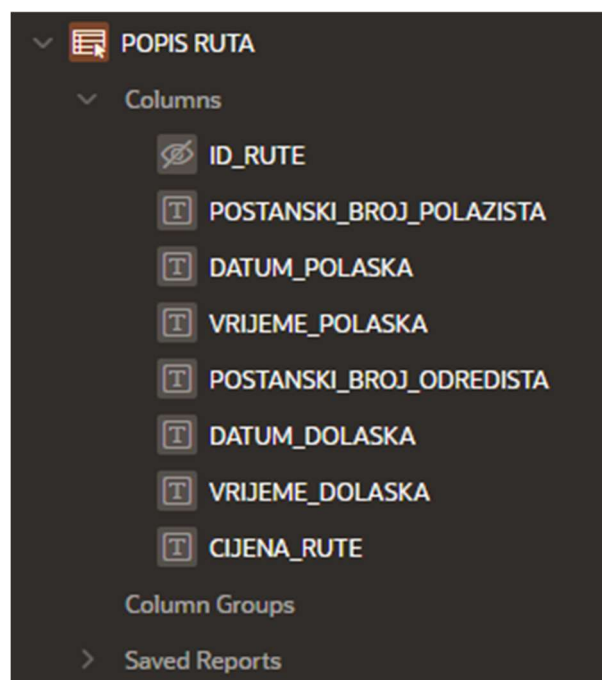
Slika 25 - Stvaranje stranice "Popis ruta"

Nakon kreiranja stranice otvara se Page Designer za prethodno stvorenu stranicu. Unutar Page Designer-a moguće je upravljati stranicom, uređivati je, dodavati elemente itd.

Kako bi prikaz ruta bio što pristupačniji, moguće je preimenovati nazive stupaca u tablici, promijeniti njihov raspored i sl. Na slici 26. može se vidjeti da je polje `id_rute` postavljeno kao „hidden“ kako se ono ne bi prikazivalo u Report-u. Na slici 27. može se vidjeti novi raspored stupaca, koji je više prilagođen korisniku.



Slika 26 - Sakrivanje stupca ID_RUTE



Slika 27 - Izmijenjen raspored stupaca tablice RUTA

Nadalje, za vanjske ključeve (`POSTANSKI_BROJ_POLAZISTA`) i `POSTANSKI_BROJ_ODREDISTA` potrebno je promijeniti vrijednost koja je prikazana, jer korisnika zanima naziv mjesta polaska i dolaska. To je postignuto uređivanjem List of Values za te stupce (na slici 27. nalazi se prikaz istoga za `POSTANSKI_BROJ_POLAZISTA`).



Slika 28 - List of Values za POSTANSKI_BROJ_POLAZISTA

Na slici 28. moguće je vidjeti da je Display vrijednost sada naziv mjesta polaska, a ne poštanski broj.

List of Values:
MJESTO_POLAZISTE.ADRESA_STANICE_POLAZISTA

Cancel Delete **Apply Changes**

Show All Name Source Subscription Comments

Source

Data Source: Local Database

Source Type: Table SQL Query Function Body Returning SQL

Table / View Owner: WKSP_NMARIIO

Table / View Name: MJESTO_POLAZISTE (table)

Where Clause:

Column Mapping

Return: POSTANSKI_BROJ_POLAZISTA

Display: NAZIV_MJESTA_POLAZISTA

Default Sort: ADRESA_STANICE_POLAZISTA

Sort Direction: Ascending (Nulls Last)

Group: - Not Defined -

Group Sort Direction: Ascending (Nulls Last)

Icon: - Not Defined -

Oracle Text Column: - Not Defined -

Additional Display Columns **Select Columns >**

Slika 29 - Izmijenjen List of Values za POSTANSKI_BROJ_POLAZAZISTA

Analogno je napravljeno i za stupac POSTANSKI_BROJ_ODREDISTA, a na slici 30. moguće je vidjeti izgled konačne stranice za popis ruta.

POPIS RUTA

Q Go Actions

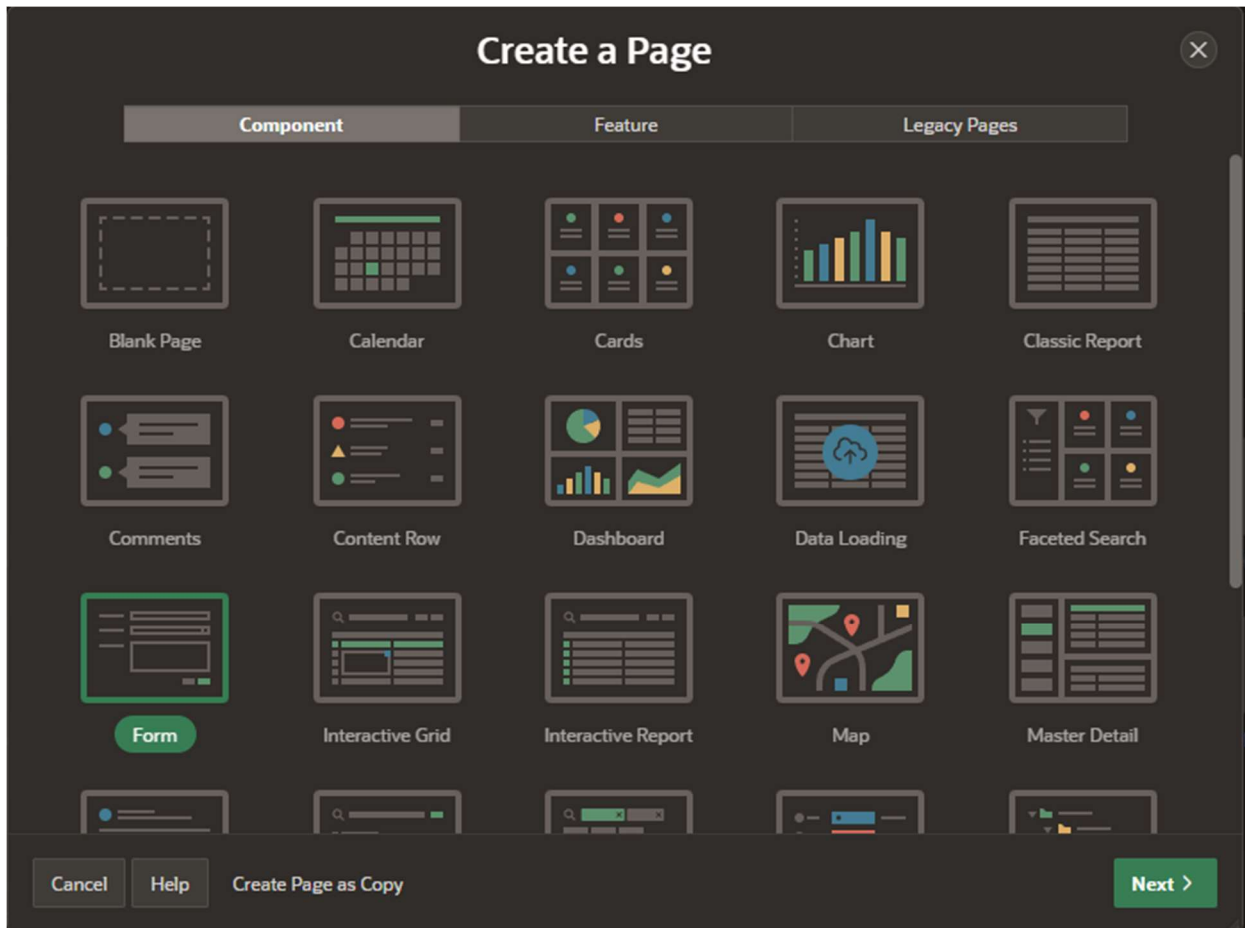
| Polaziste | Datum polaska | Polazak | Odrediste | Datum dolaska | Dolazak | Cijena (€) |
|----------------|---------------|---------|-----------|---------------|---------|------------|
| Slavonski Brod | 6/6/2024 | 17:30 | Rijeka | 6/6/2024 | 22:30 | 45 |

1 - 1

Slika 30 - Stranica "Popis ruta"

5.2.2. Izrada stranice za unos nove rezervacije

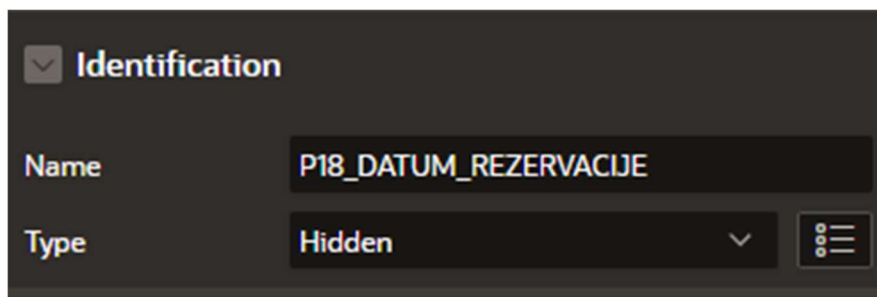
Nakon kreiranja stranice na kojoj je moguće pregledati sve rute (koje je moguće i filtrirati po željenim vrijednostima), slijedi izrada stranice za unos rezervacije. Budući da se radi o stranici čija je svrha unos podataka, odabiremo tip stranice Form, što se može vidjeti na slici 31.



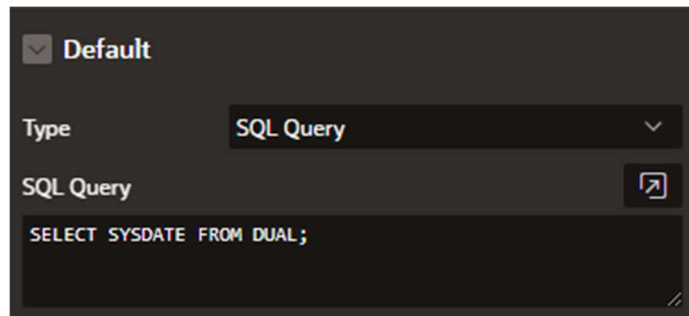
Slika 31 - Stvaranje forme za unos podataka

Nakon stvaranja forme (koja je prethodno povezana s tablicom REZERVACIJA iz baze podataka), otvara se Page Designer, unutar kojega se uređuje forma.

Najprije je potrebno „sakriti“ sva polja osim polja „id_rute“, jer u ovoj formi korisnik odabire isključivo rutu, a ostali podaci imaju zadanu vrijednost koja se sprema. (što je vidljivo na slikama 32. i 33. za stupac datum_rezervacije).

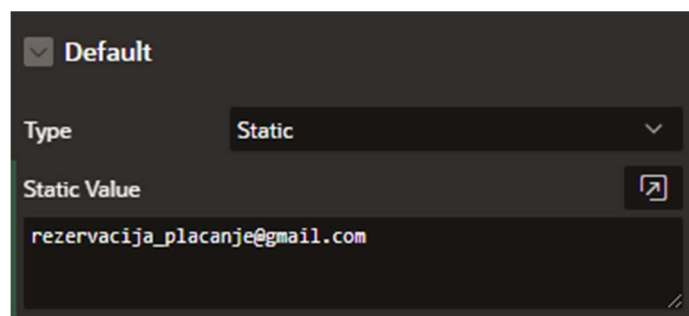


Slika 32 - Skriveni stupac "datum_rezervacije"

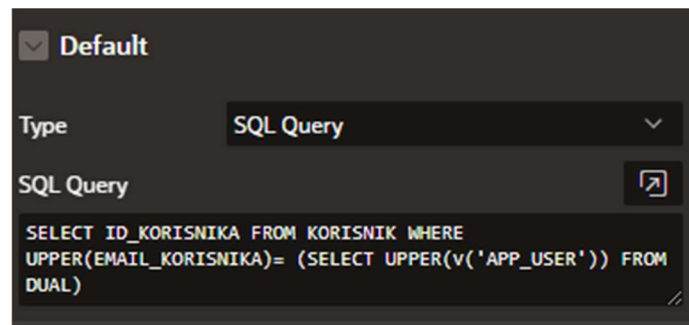


Slika 33 - Postavljanje defaultne vrijednosti na stupac "datum_rezervacije"

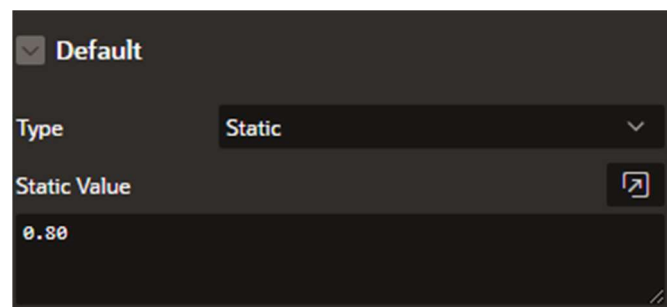
Isto je napravljeno i za preostale stupce koji su sakriveni od korisnika (email_za_paypal_placanje, id_korisnika i iznos_rezervacije), a zadana vrijednost postavljena za ta polja može se vidjeti na slikama 34., 35. i 36.



Slika 34 - Default vrijednost za polje EMAIL_ZA_PAYPAL_PLACANJE

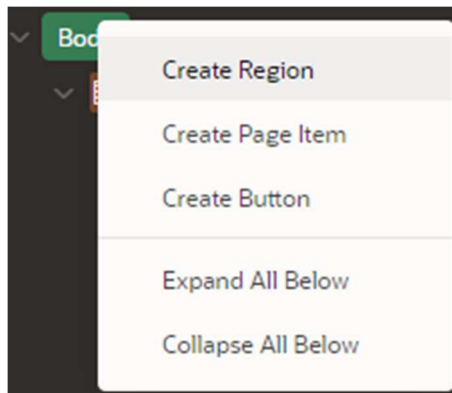


Slika 35 - Default vrijednost za polje ID_KORISNIKA



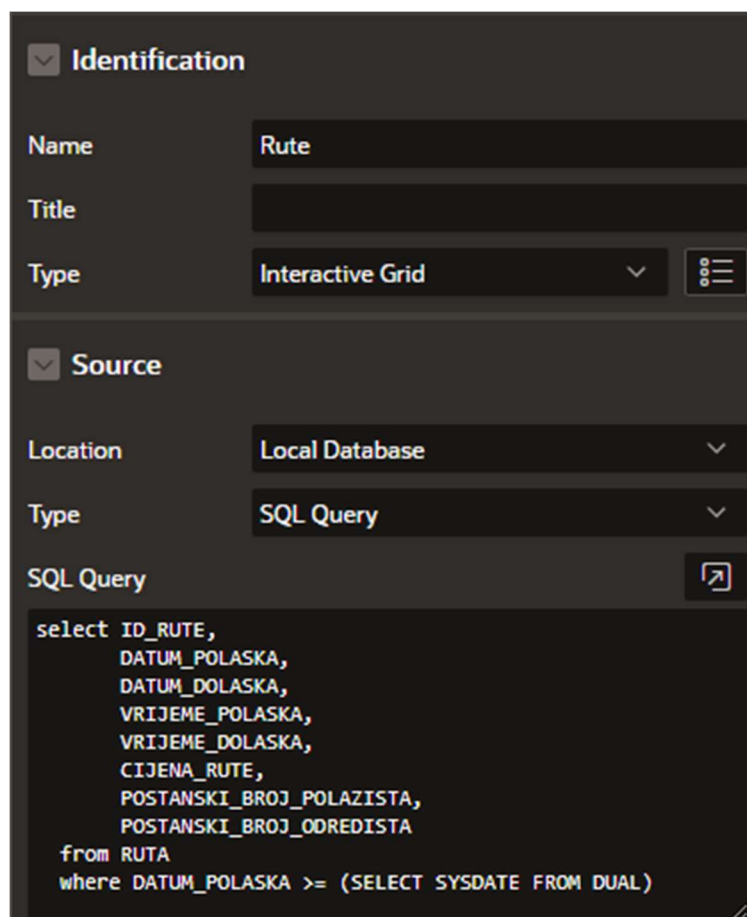
Slika 36 - Default vrijednost za polje ID_KORISNIKA

Nakon sakrivanja prethodno navedenih stupaca i postavljanja zadanih vrijednosti za iste, slijedi implementacija logike za odabir i unos rute za rezervaciju. Najprije je potrebno implementirati prikaz svih ruta (koje korisnik može filtrirati i od kojih može odabrati jednu za rezervaciju). Za to je kreirana nova regija (desnim klikom na Body regiju), što je i prikazano na slici 37.



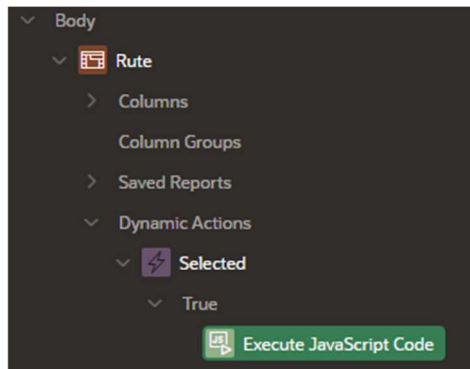
Slika 37 - Stvaranje nove regije

Naziv kreirane regije mijenja se u „Rute“, a tip regije postavlja se na „Interactive Grid“. Pod Source za tip je odabran „SQL Query“, a na slici 38. vidljive su sve postavke za kreiranu regiju.



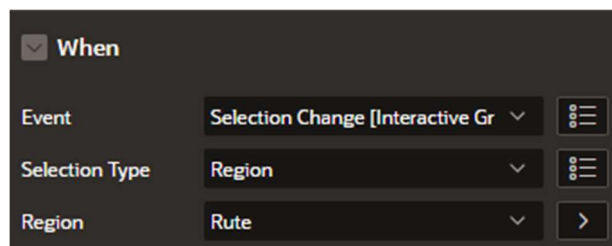
Slika 38 - Postavke regije „Rute“

Nakon izlistavanja svih ruta potrebno je povezati te dvije regije (regiju za prikaz ruta i regiju za unos nove rezervacije). Najprije je potrebno u ID_RUTE iz forme za unos rezervacije postaviti vrijednost primarnog ključa odabranog retka iz tablice sa popisom ruta. Desnim klikom na regiju „Rute“ potrebno je kreirati novu dinamičku akciju koja je prikazana na slici 39.



Slika 39 - Dinamička akcija za prosljeđivanje primarnog ključa u formu

Prethodno kreirana dinamička akcija se izvršava kada je odabran neki redak iz tablice sa popisom svih ruta (slika 40.)



Slika 40 - Postavke za izvršavanje dinamičke akcije „Selected“

Kada je neki redak odabran, u ID_RUTE forme za unos rezervacije sprema se ID_RUTE odabranog retka pomoću JavaScript programskog koda (slika 41.)

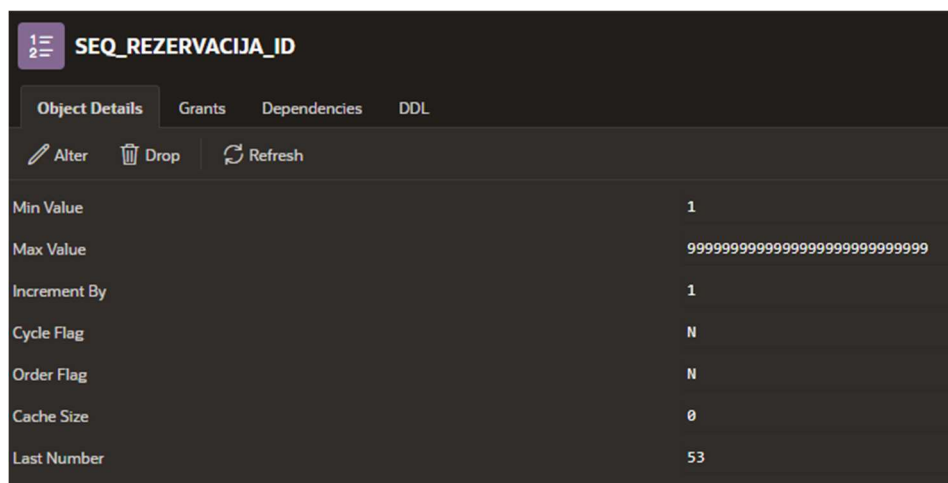


Slika 41 - Postavke akcije za prosljeđivanje ID_RUTE u formu

Sada je korisniku omogućeno odabrati neku od ruta, a samim odabirom ju proslijediti u formu za rezervaciju te klikom na gumb iz forme kreirati novu rezervaciju.

Budući da rezervacija ima svoj primarni ključ (ID_REZERVACIJE) potrebno je implementirati funkcionalnost automatskog generiranja vrijednosti za to polje jer nije predviđeno za korisnika da unosi takve podatke.

U Oracle APEX bazi podataka, sekvenca je objekt koji se koristi za generiranje jedinstvenih numeričkih vrijednosti. Često su korišteni za automatsko generiranje primarnih ključeva u tablicama. Na slici 42. prikazane su postavke sekvence koja je kreirana s ciljem automatskog generiranja polja ID_REZERVACIJE.

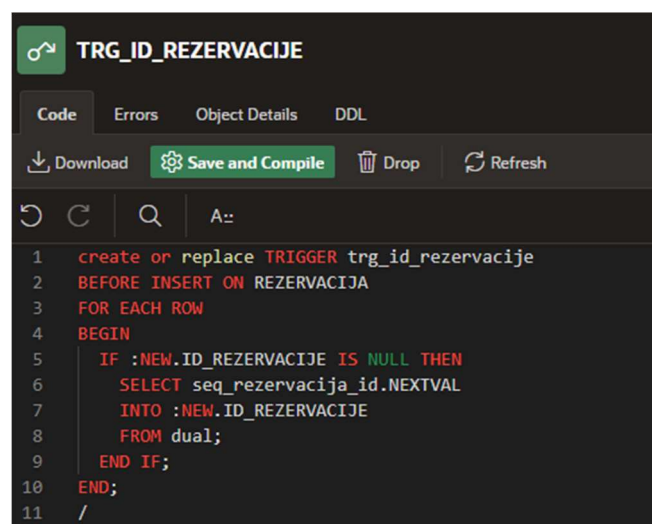


| SEQ_REZERVACIJA_ID | |
|--------------------|-----------------------------------|
| Min Value | 1 |
| Max Value | 999999999999999999999999999999999 |
| Increment By | 1 |
| Cycle Flag | N |
| Order Flag | N |
| Cache Size | 0 |
| Last Number | 53 |

Slika 42 - Sekvenca SEQ_REZERVACIJA_ID

Osim sekvenca, u Oracle APEX bazama podataka moguće je kreirati i okidač (eng. Trigger). Okidač je posebna vrsta PL/SQL programa koja se automatski izvršava kao odgovor na određene događaje koji se događaju unutar određene tablice (ili prikaza). Često se koriste za provođenje poslovnih pravila, održavanja integriteta podataka i sl.

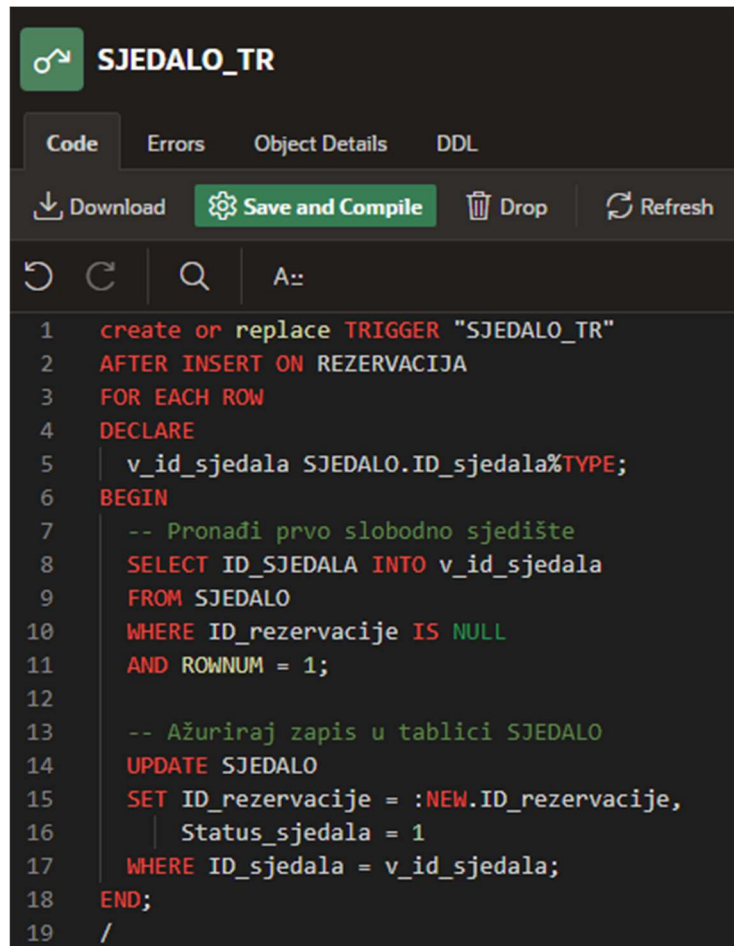
Na slici 43. prikazan je okidač koji svakoj novoj rezervaciji dodjeljuje ID_REZERVACIJE generiran u prethodno opisanoj sekvenci.



```
1 create or replace TRIGGER trg_id_rezervacije
2 BEFORE INSERT ON REZERVACIJA
3 FOR EACH ROW
4 BEGIN
5     IF :NEW.ID_REZERVACIJE IS NULL THEN
6         SELECT seq_rezervacija_id.NEXTVAL
7         INTO :NEW.ID_REZERVACIJE
8         FROM dual;
9     END IF;
10 END;
11 /
```

Slika 43 - Okidač TRG_ID_REZERVACIJE

Sada je još potrebno implementirati automatsko dodjeljivanje nove rezervacije nekom od dostupnih sjedala. To je također napravljeno pomoću okidača, koji pronalazi prvo sjedalo kojemu je vrijednost polja ID_REZERVACIJE jednaka null vrijednosti, te mijenja vrijednost toga polja u vrijednost primarnog ključa nove rezervacije. Okidač za taj proces prikazan je na slici 44.

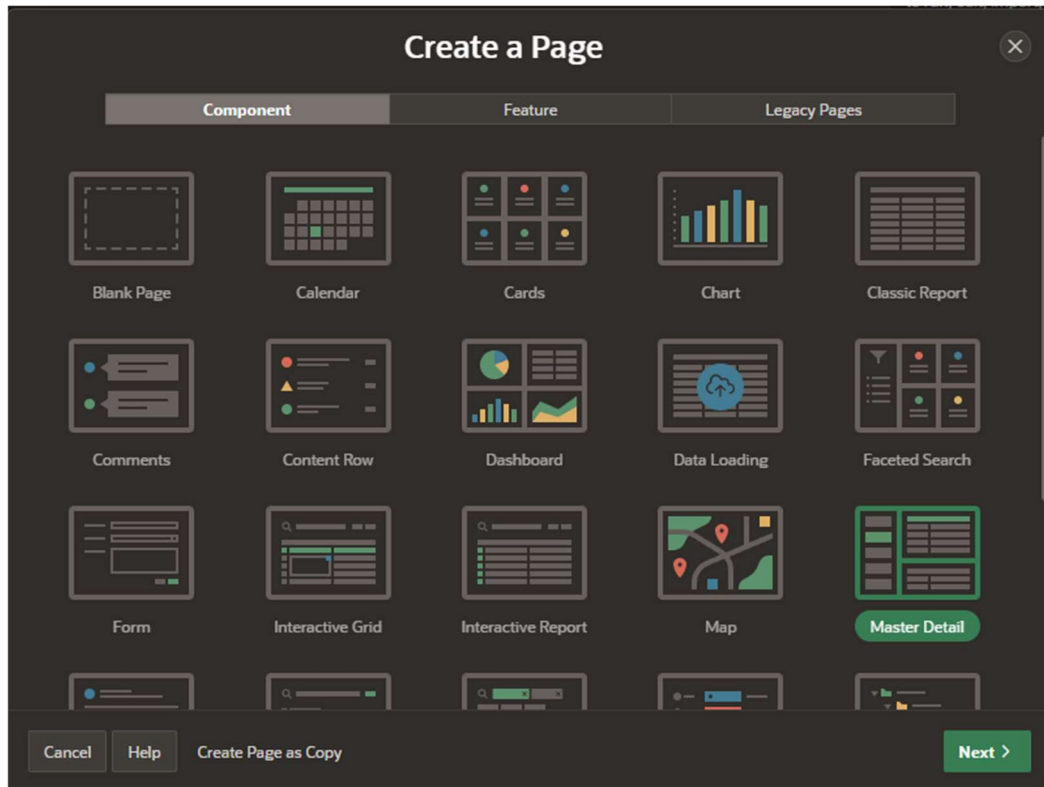


```
1 create or replace TRIGGER "SJEDALO_TR"
2 AFTER INSERT ON REZERVACIJA
3 FOR EACH ROW
4 DECLARE
5     v_id_sjedala SJEDALO.ID_sjedala%TYPE;
6 BEGIN
7     -- Pronađi prvo slobodno sjedište
8     SELECT ID_SJEDALA INTO v_id_sjedala
9     FROM SJEDALO
10    WHERE ID_rezervacije IS NULL
11    AND ROWNUM = 1;
12
13    -- Ažuriraj zapis u tablici SJEDALO
14    UPDATE SJEDALO
15    SET ID_rezervacije = :NEW.ID_rezervacije,
16        Status_sjedala = 1
17    WHERE ID_sjedala = v_id_sjedala;
18 END;
19 /
```

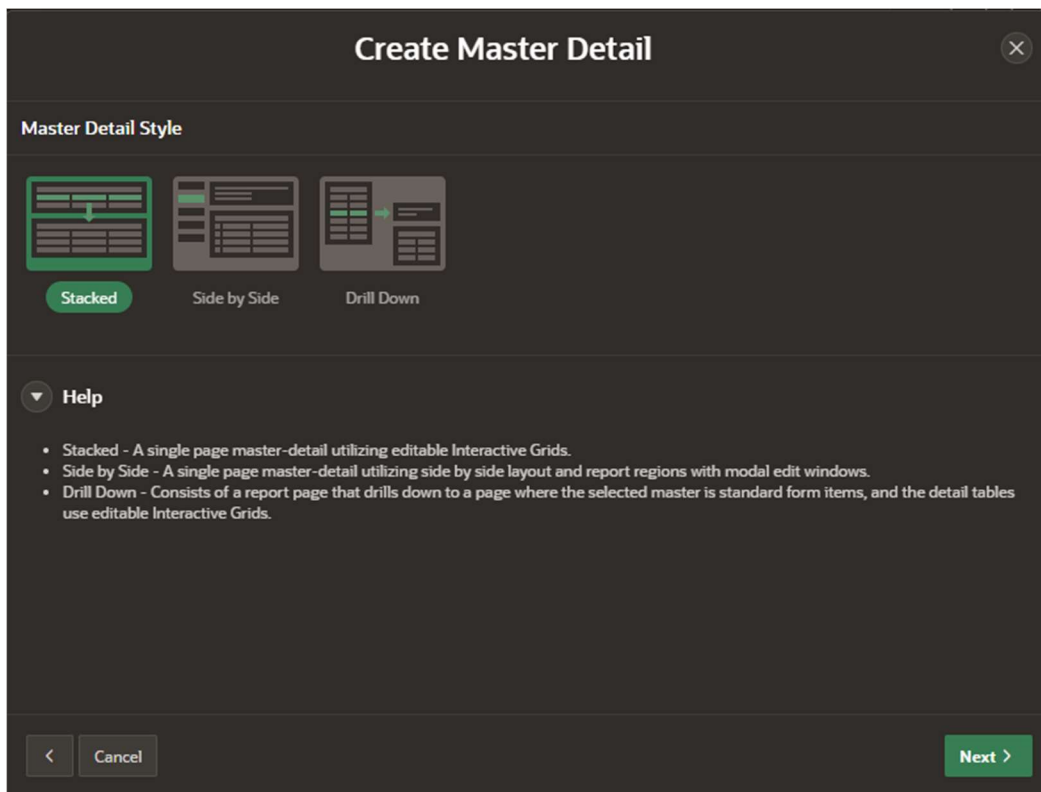
Slika 44 - Okidač SJEDALO_TR

5.2.3. Izrada stranice za prikaz svih rezervacija

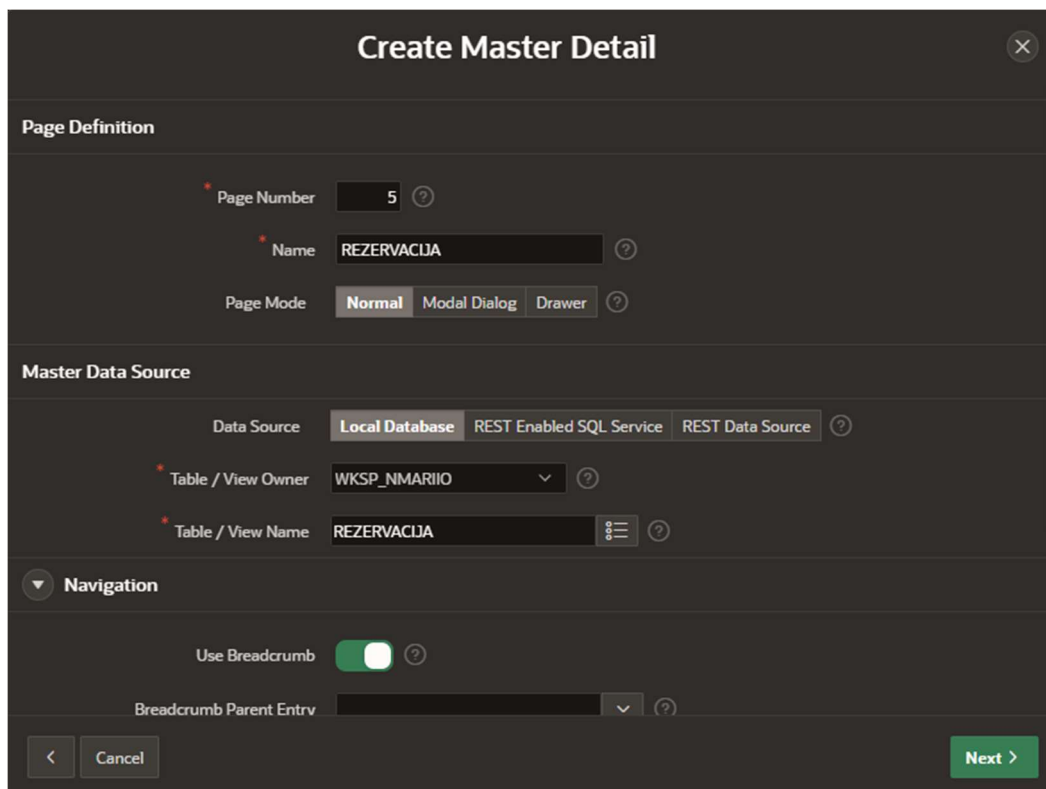
Sada kada je korisniku omogućen unos nove rezervacije, potrebno je kreirati stranicu na kojoj će korisnik moći vidjeti svoje rezervacije, detalje o istima te ih brisati. Za to se koristi Master-Detail stranica, koja korisniku pruža mogućnosti upravljanja dvjema (ili više) povezanih tablica gdje „master“ tablica sadržava osnove podatke, a „detail“ tablica sadrži povezane detalje. Na slikama 45., 46., 47. i 48. prikazani su postupci izrade Master-Detail stranice, gdje je master tablica REZERVACIJA, a detail SJEDALO.



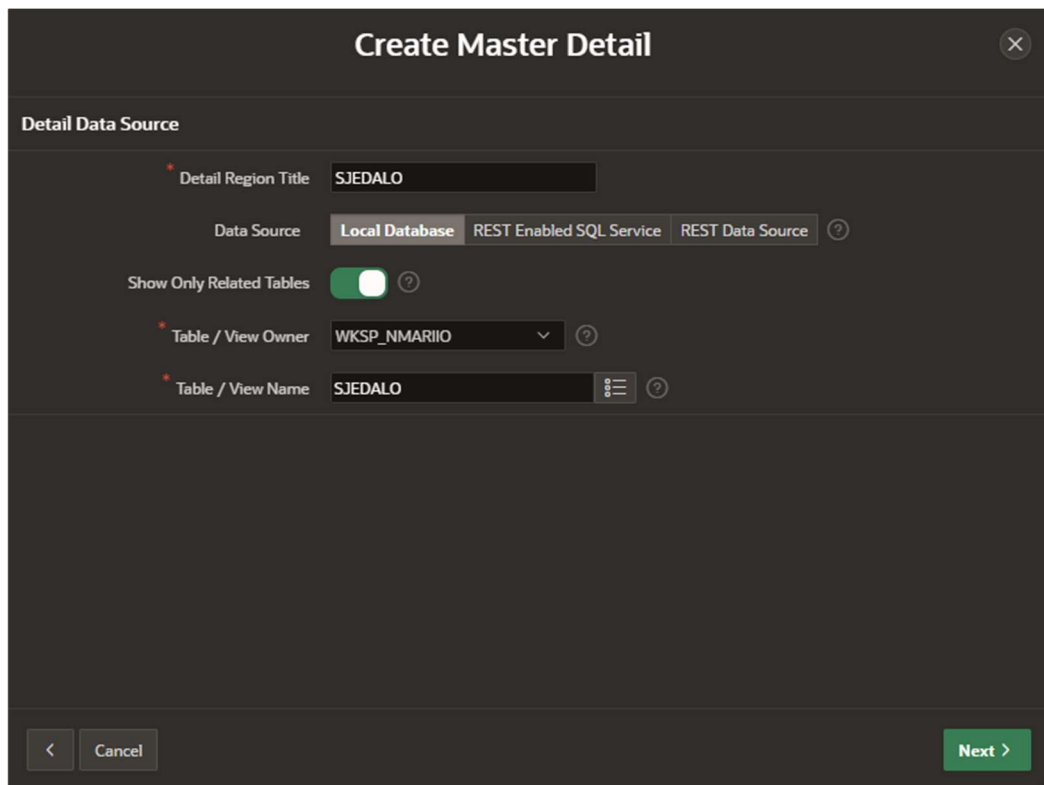
Slika 45 - Kreiranje nove Master-Detail stranice



Slika 46 - Odabir stila Master-Detail stranice

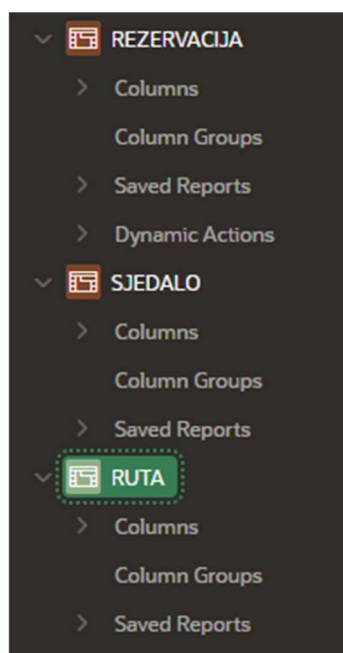


Slika 47 - Odabir master tablice

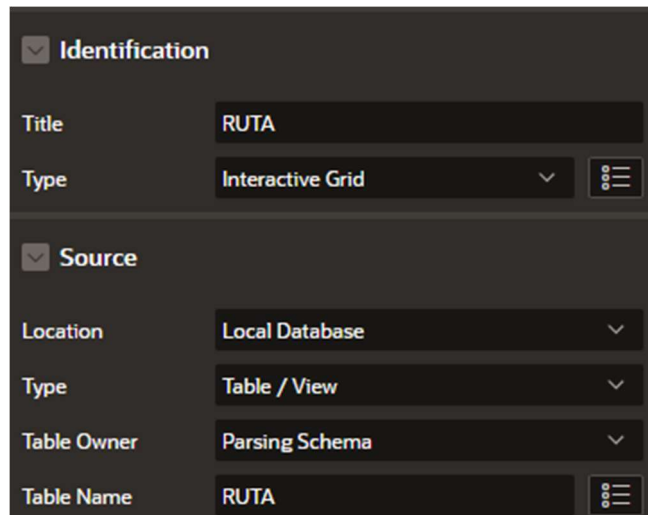


Slika 48 - Odabir detail stranice

Nakon postavljanja svih postavki Master-Detail stranice, otvara se Page Designer. Prvo što je potrebno napraviti je dodavanje još jedne „detail“ stranice, jer korisnika osim detalja o sjedalu zanimaju i detalji o ruti koja je rezervirana. To je najlakše napraviti na način da se „detail“ tablica SJEDALO duplicira i nakon toga promijenjene postavke nove „detail“ tablice, što je prikazano na slikama 49. i 50.



Slika 49 - Prikaz master i detail tablica



Slika 50 - Prikaz postavni nove detail tablice

Nadalje, potrebno je promijeniti izvor podatka za master tablicu. Naime, sada su prikazane sve rezervacije svih korisnika, ali potrebno je prikazati samo rezervacije trenutno prijavljenog korisnika. To se modificira na način da se u postavkama master tablice Type promijeni na SQL Query i postavi SQL upit. Na slici 51. prikazan je SQL upit za dohvaćanje podataka master tablice REZERVACIJA.

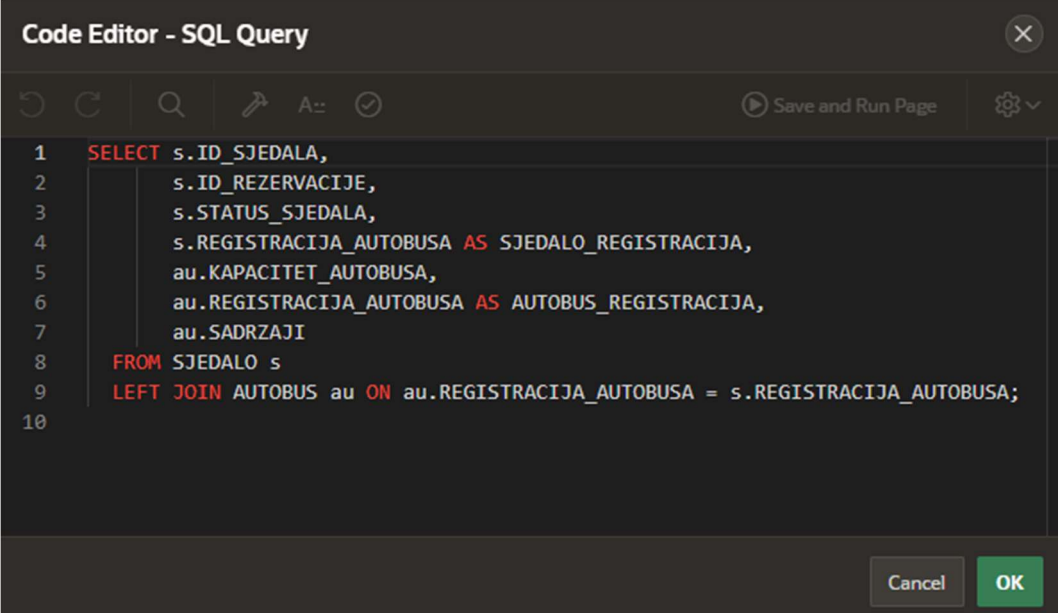
```

Code Editor - SQL Query
1  SELECT
2     R.ID_REZERVACIJE,
3     R.DATUM_REZERVACIJE,
4     R.EMAIL_ZA_PAYPAL_PLACANJE,
5     R.STATUS_REZERVACIJE,
6     R.ID_RUTE,
7     R.ID_KORISNIKA,
8     R.IZNOS_REZERVACIJE,
9     RT.CIJENA_RUTE,
10    R.IZNOS_REZERVACIJE + RT.CIJENA_RUTE AS UKUPNA_CIJENA
11  FROM
12    REZERVACIJA R
13  JOIN
14    RUTA RT ON R.ID_RUTE = RT.ID_RUTE
15  WHERE
16    R.ID_KORISNIKA = (
17      SELECT ID_KORISNIKA FROM KORISNIK
18      WHERE UPPER(EMAIL_KORISNIKA)= (SELECT UPPER(v('APP_USER')) FROM DUAL)
19    )
20
21
Cancel OK

```

Slika 51 - SQL upit za master tablicu

Isto je potrebno napraviti i za „detail“ tablicu SJEDALO. Korisnika osim detalja o sjedalu zanimaju i detalji o autobusu (registracija, sadržaji i sl.), a za realizaciju toga korišten je LEFT JOIN tablice SJEDALO i AUTOBUS koji omogućuje da se prikažu svi detalji o sjedalima, uključujući i detalje o autobusu kojemu određeno sjedalo pripada (što je i prikazano na slici 52.).



```
Code Editor - SQL Query
1 SELECT s.ID_SJEDALA,
2       s.ID_REZERVACIJE,
3       s.STATUS_SJEDALA,
4       s.REGISTRACIJA_AUTOBUSA AS SJEDALO_REGISTRACIJA,
5       au.KAPACITET_AUTOBUSA,
6       au.REGISTRACIJA_AUTOBUSA AS AUTOBUS_REGISTRACIJA,
7       au.SADRZAJI
8 FROM SJEDALO s
9 LEFT JOIN AUTOBUS au ON au.REGISTRACIJA_AUTOBUSA = s.REGISTRACIJA_AUTOBUSA;
10
Cancel OK
```

Slika 52 - SQL upit za detail tablicu SJEDALO

Sada je korisniku omogućeno pregledavanje rezervacija te detalja o sjedalu i ruti za pojedinu rezervaciju.

5.2.4. Implementacija potvrde o rezervaciji

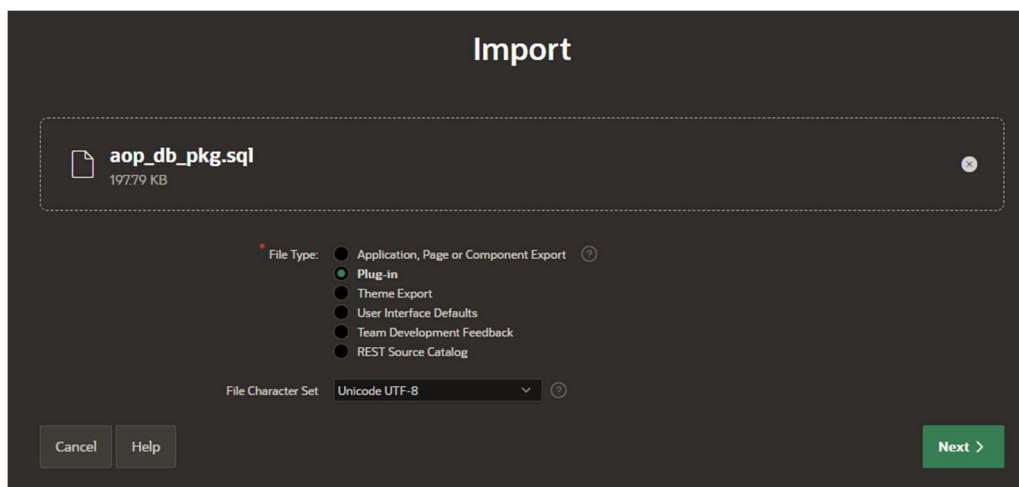
Nakon prikaza svih rezervacija sa pripadnim detaljima, potrebno je implementirati funkcionalnost generiranja potvrde o rezervaciji.

Apex Office Print (AOP) je alat koji se integrira sa Oracle APEX-om za generiranje različitih vrsta dokumenata kao što su PDF, Word, Excel i PowerPoint iz Oracle APEX aplikacija. Omogućuje korisnicima lako ispisivanje dokumenata koristeći podatke iz Oracle APEX aplikacija. Podaci za dokumente mogu dolaziti iz različitih izvora, uključujući SQL upite, PL/SQL procedure, RESTful service i druge izvore podataka dostupne u Oracle APEX-u.

Najprije je na službenoj web stranici (<https://www.apexofficeprint.com/>) potrebno kreirati korisnički račun (za generiranje API ključa koji je potreban za korištenje AOP-a unutar Oracle APEX-a).

Nakon registracije/prijave, potrebno je locirati se u Downloads panel i unutar njega preuzeti Cloud Package. Preuzetu datoteku je potom potrebno raspakirati kako bi se pristupilo datotekama koje su potrebne za umetanje u Oracle APEX.

Nadalje, potrebno je otvoriti Plug-ins (Shared Components→Other Components→Plug-ins) te klikom na import učitati datoteku čiji naziv je prikazan na slici 53. U nastavku instalacije potrebno je upisati prethodno generirani API ključ te kompletno učitati dodatak.



Slika 53 - Učitavanje AOP datoteke

Nakon instalacije, u Plug-ins je vidljivo da je dodatak učitani, što je prikazano na slici 54.



Slika 54 - Učitani AOP dodatak

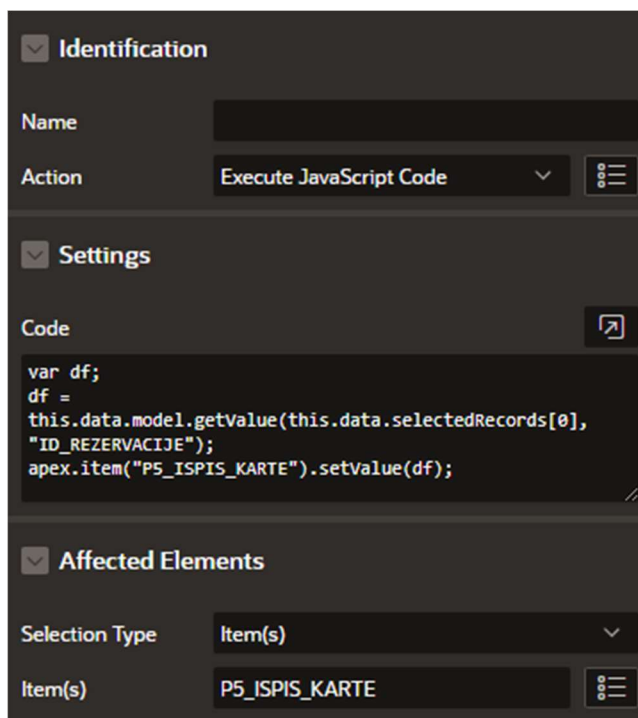
Zadnji korak za kompletnu instalaciju AOP dodatka je pokretanje SQL skripte. Lociranjem na SQL Scripts (SQL-Workshop→SQL Scripts) otvaraju se SQL skripte te je potrebno kliknuti na Run kako bi se skripta izvršila (slika 55.).

| <input type="checkbox"/> | Edit | Owner | Name | Created | Updated By | Updated | Bytes | Results | Run |
|--------------------------|------|-----------------------|----------------|------------|-----------------------|------------|---------|---------|-----|
| <input type="checkbox"/> | | NMARIIO2401@GMAIL.COM | aop_db_pkg.sql | 2 days ago | NMARIIO2401@GMAIL.COM | 2 days ago | 202,535 | 1 | |

Slika 55 – Izvršena SQL skripta

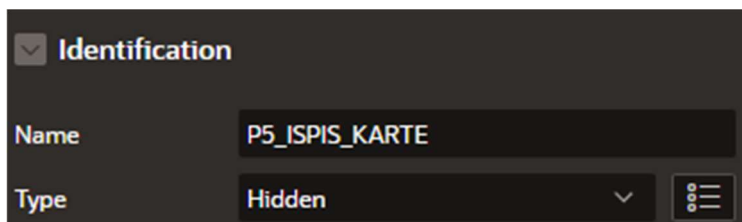
Sada kada je dodatak instaliran, moguće je implementirati funkcionalnosti ispisivanja podataka o rezervaciji u PDF formatu.

Najprije je potrebno kreirati novi Page Item unutar kojega će se pohraniti trenutno selektirana rezervacija od strane korisnika. Kako bi se to i realiziralo, potrebno je kreirati novu dinamičku akciju. Akcija je „Execute Javascript Code“, a u Code se pohranjuje JavaScript program koji u Page Item (P5_ISPIS_KARTE) sprema vrijednost primarnog ključa trenutno odabrane rezervacije (slika 56.).



Slika 56 - Dinamička akcija „Execute Javascript Code“

Stvoreni Page Item može se promatrati kao posrednika između rezervacije i pdf potvrde. Putem njega se prenosi ID selektirane rezervacije AOP-u, te se stoga može sakriti od korisnika (postavlja se Type na Hidden, što je prikazano na slici 57.).



Slika 57 - Hidden Page Item

Na slici 58. prikazan je predložak za PDF potvrdu o rezervaciji stvoren unutar alata Microsoft Word. Isti predložak je potrebno učitati u Shared Components → Static Application Files kako bi se mogao iskoristiti u AOP potvrdi.

ID REZERVACIJE: {id_rezervacije}
DATUM REZERVACIJE: {datum_rezervacije}



KARTA

IME I PREZIME: {ime_korisnika} {prezime_korisnika}

DATUM I VRIJEME POLASKA: {datum_polaska} {vrijeme_polaska}h

DATUM I VRIJEME DOLASKA: {datum_dolaska} {vrijeme_dolaska}h

POLAZIŠTE: {mjesto_polaziste}

ODREDIŠTE: {mjesto_odrediste}

BROJ SJEDALA: {id_sjedala }

EMAIL ZA PAYPAL PLAĆANJE: {email_za_paypal_placanje}

CIJENA RUTE: {cijena_rute}€

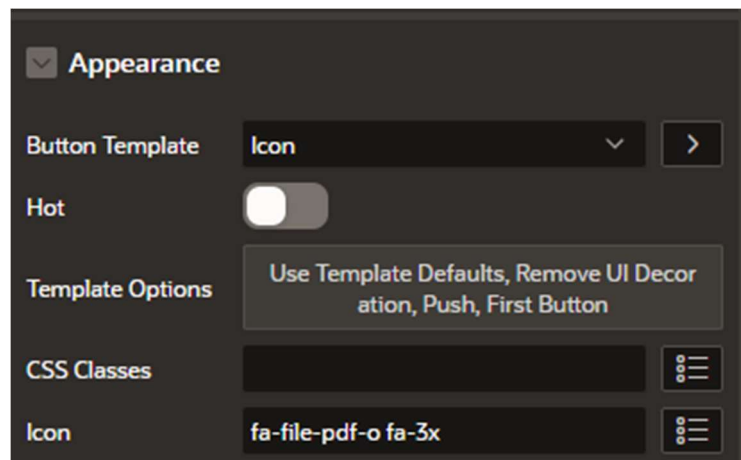
IZNOS REZERVACIJE: {iznos_rezervacije}€

UKUPNA CIJENA: {ukupna_cijena}€

Slika 58 - Predložak za PDF potvrdu o rezervaciji

Nakon stvaranja i učitavanja predloška slijedi implementacija generiranja potvrde o rezervaciji u PDF formatu.

Najprije se stvara novi gumb, naziva PRINT, klikom na koji se generira nova potvrda. Unutar Appearance postavki gumba (koje su prikazane na slici 59.) Button Template postavlja se na Icon, a za Icon vrijednost stavlja se 'fa-file-pdf-o fa-3x'. Na slici 60. prikazan je konačni izgled gumba.



Slika 59 - Appearance postavke gumba PRINT



Slika 60 - Izgled gumba PRINT

Desnim klikom na gumb PRINT stvara se nova dinamička akcija naziva AOP, koja se aktivira klikom na gumb PRINT. Kada je uvjet ispunjen (odnosno kada je gumb pritisnut) pokreće se AOP akcija „UC – APEX Office Print (AOP) – DA [Plug-in]“ (prikazano na slici 61.). Za postavku Template Type stavlja se Static Application Files, a pod Template Source naziv učitano predloška (prikazano na slici 63.). Potom je potrebno u Data Type odabrati SQL, te u Data Source navesti SQL sintaksu za dohvaćanje podataka koji se spremaju u potvrdu (prikazano na slici 62.). Output Type stavlja se na PDF.



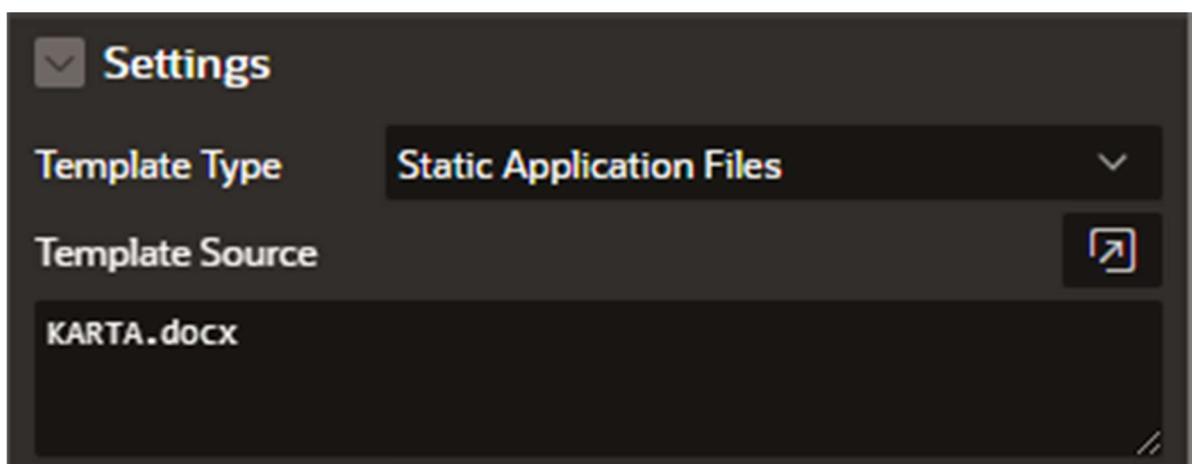
Slika 61 - Akcija za generiranje PDF potvrde o rezervaciji

```
Code Editor - Data Source
Save and Run Page

1 SELECT 'file1' as "karta",
2 CURSOR(SELECT
3 r.id_rezervacije as "id_rezervacije",
4 to_char(r.datum_rezervacije, 'DD-MM-YYYY') as "datum_rezervacije",
5 r.email_za_paypal_placanje as "email_za_paypal_placanje",
6 r.iznos_rezervacije as "iznos_rezervacije",
7 r.id_korisnika as "id_korisnika",
8 k.ime_korisnika as "ime_korisnika",
9 s.id_rezervacije,
10 s.id_sjedala as "id_sjedala",
11 k.prezime_korisnika as "prezime_korisnika",
12 mp.postanski_broj_polazista,
13 mo.postanski_broj_odredista,
14 mo.naziv_mjesta_polazista as "mjesto_polaziste",
15 mo.naziv_mjesta_odredista as "mjesto_odrediste",
16 to_char(ru.datum_polaska, 'DD-MM-YYYY') as "datum_polaska",
17 to_char(ru.datum_dolaska, 'DD-MM-YYYY') as "datum_dolaska",
18 ru.vrijeme_polaska as "vrijeme_polaska",
19 ru.vrijeme_dolaska as "vrijeme_dolaska",
20 ru.cijena_rute as "cijena_rute",
21 r.iznos_rezervacije + ru.cijena_rute as "ukupna_cijena",
22 r.id_rute as "id_rute"
23 FROM REZERVACIJA r, RUTA ru, SJEDALO s, KORISNIK k, MJESTO_ODREDISTE mo, MJESTO_POLAZISTE mp
24 WHERE
25 k.ID_KORISNIKA=r.id_korisnika AND
26 ru.id_rute=r.id_rute AND
27 s.id_rezervacije=r.id_rezervacije AND
28 mp.postanski_broj_polazista=ru.postanski_broj_polazista AND
29 mo.postanski_broj_odredista=ru.postanski_broj_odredista AND
30 r.id_rezervacije=:P5_ISPIS_KARTE
31 ) AS "data" FROM dual

Cancel OK
```

Slika 62 - SQL izvor za dohvaćanje podataka za potvrdu o rezervaciji



Slika 63 - Postavke za izvor predloška potvrde

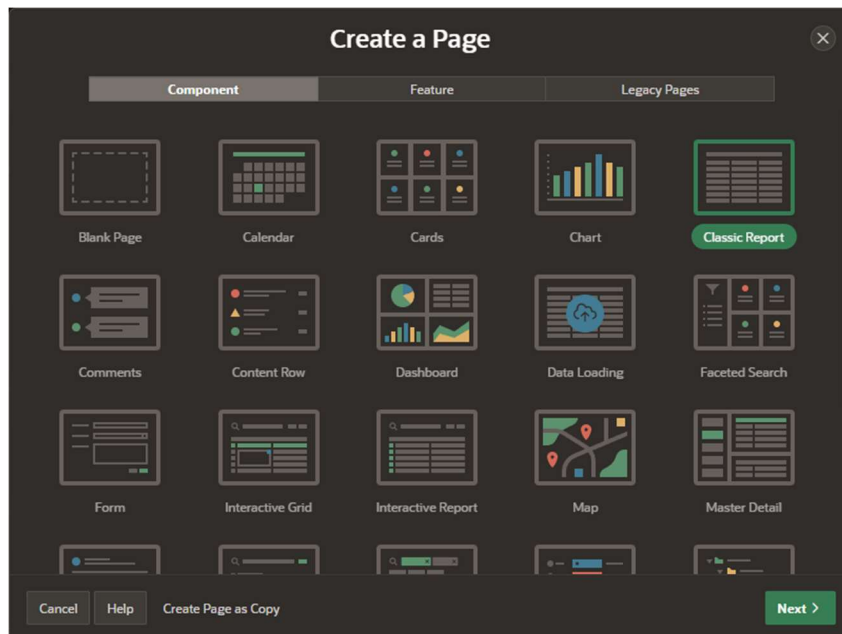
5.2.5. Izrada preostalih stranica

Nakon kreiranja stranica namijenjenih za krajnjeg korisnika (koji ima ulogu putnika) slijedi izrada stranica namijenjena za samog administratora aplikacije. Radi se o stranicama koje služe za unos novih podataka (npr. nove rute), izmjenu i pregled podataka.

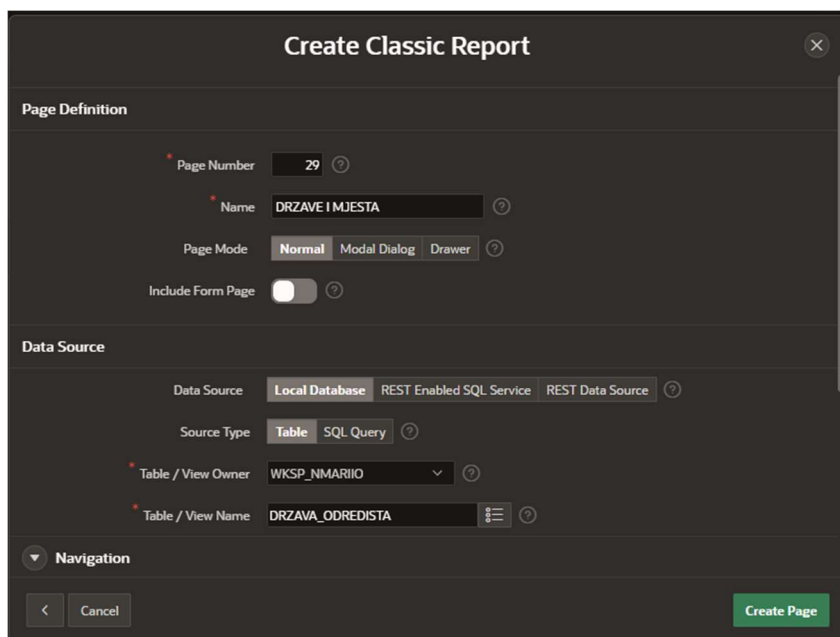
Postoje 4 grupe stranica koje se trebaju izraditi a to su:

1. DRŽAVE I MJESTA – pregled i unos novih država polazišta/odredišta i mjesta polazišta/odredišta,
2. AUTOBUS I SJEDALO – pregled autobusa i sjedala te unos novih,
3. UNOS RUTE,
4. PREGLED SVIH REZERVACIJA.

Za pregled država odredišta/polazišta i mjesta odredišta/polazišta koristimo tip stranice Classic Report. Na slici 64. prikazan je ekran za stvaranje nove stranice Classic Report, a na slici 65. postavke te stranice.

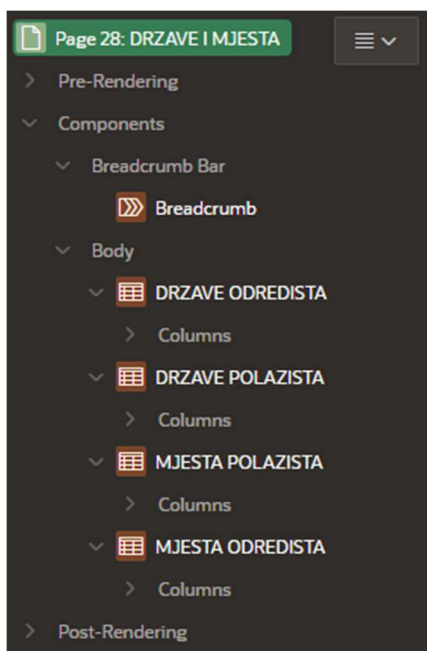


Slika 64 - Stvaranje nove Classic Report stranice



Slika 65 - Postavke stranice za pregled država i mjesta

U Page Designer alatu tablicu DRZAVA_ODREDISTA dupliciramo te postavke nove tablice promijenimo da odgovara tablici DRZAVA_POLAZISTA, te isto učinimo za tablice MJESTO_ODREDISTA i MJESTO_POLAZISTA. Na slici 66. prikazane su te tablice kao regije.



Slika 66 - Prikaz svih regija za pregled država i mjesta

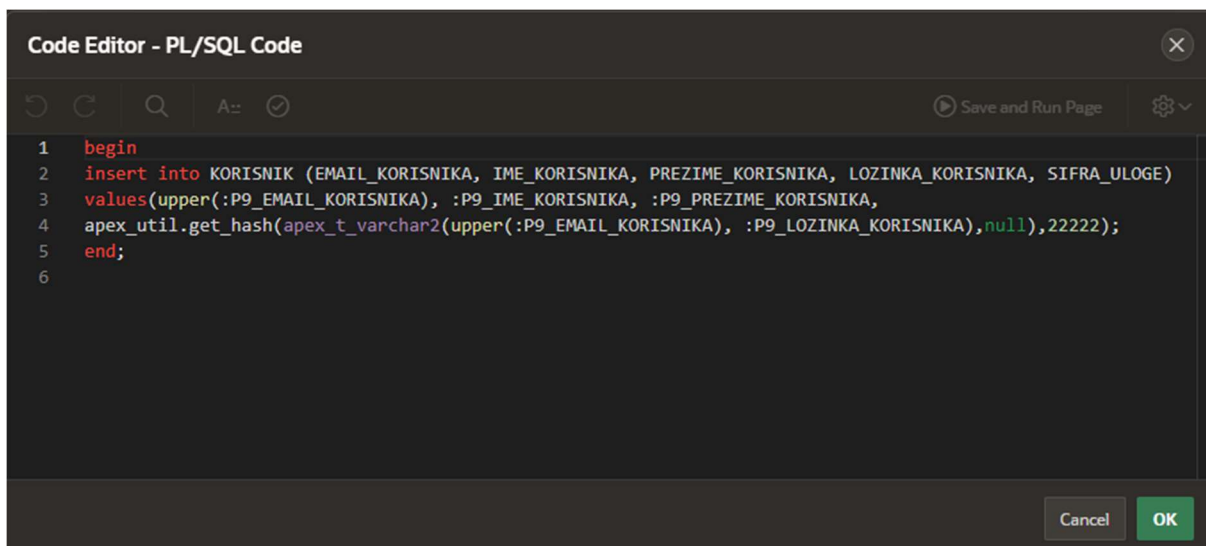
Sada je administratoru sustava na jednoj stranici omogućen pregled podataka o svim državama i mjestima. Preostalo je omogućiti još i unos novih podataka za istoimene tablice. Za potrebe toga kreirane su forme za unos podataka (4 forme).

Za pregled autobusa i sjedala korištena je Master-Detail stranica, a za pregled svih rezervacija korištena je stranica tipa Interactive Report. Za unos nove rute iskorišten je Form tip stranice.

Kada je to osigurano stvara se nova stranica (tipa Form) koja služi za samu registraciju.

Unutar Page Designer alata potrebno je kreirati novi proces. Naziv mu je „Registracija“, a Type „Execute Code“. U Source postavkama postavljaju se sljedeće vrijednosti:

1. Location: Local Database
2. Language: PL/SQL
3. PL/SQL Code prikazan je na slici 69.

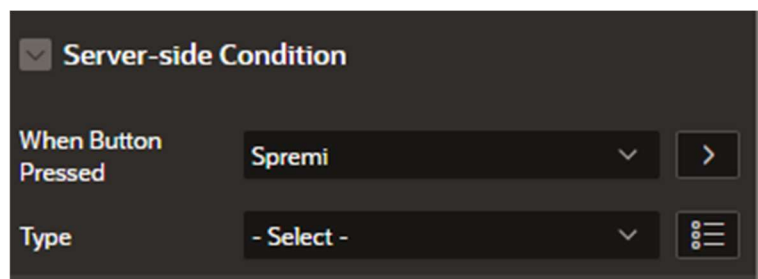


```
Code Editor - PL/SQL Code
1 begin
2 insert into KORISNIK (EMAIL_KORISNIKA, IME_KORISNIKA, PREZIME_KORISNIKA, LOZINKA_KORISNIKA, SIFRA_ULOGA)
3 values(upper(:P9_EMAIL_KORISNIKA), :P9_IME_KORISNIKA, :P9_PREZIME_KORISNIKA,
4 apex_util.get_hash(apex_t_varchar2(upper(:P9_EMAIL_KORISNIKA), :P9_LOZINKA_KORISNIKA),null),22222);
5 end;
6
```

Slika 69 - PL/SQL kod procesa Registracija

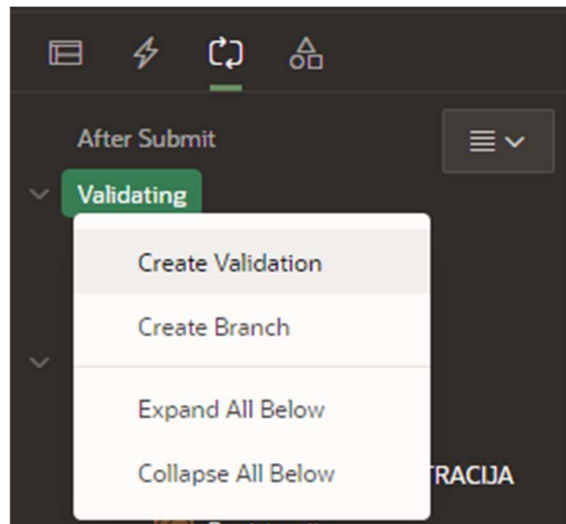
Dakle, PL/SQL kod započinje procesom begin, potom u tablicu KORISNIK koristeći klauzulu INSERT pohranjuje vrijednosti koje je korisnik unio u formi za registraciju. Pritom se koristi funkcija apex_util.get_hash() kako bi se generirala hash lozinka korisnika. Ulaz za ovu funkciju je apex_t_varchar2 koji kombinira email korisnika i njegovu lozinku. Šifru uloge ne unosi korisnik, već je ona postavljena na vrijednost 22222 jer je to šifra putnika, a registracija i je namijenjena samo korisnicima koji imaju tu ulogu.

Konačno, za ovaj proces je potrebno definirati i kada se pokreće, a to je pritiskom na gumb (što je prikazano na slici 70.).



Slika 70 - Server-side Condition postavke procesa Registracija

Naposlijetku je potrebno dodati validaciju s ciljem provjere unesene e-mail adrese. Potrebno je onemogućiti registraciju korisnika ako je korisnik za registraciju unio već postojeću e-mail adresu. Desnim klikom na regiju Validating potrebno je odabrati „Create Validation“. Na slici 71. prikazan je postupak stvaranja nove validacije.



Slika 71 - Stvaranje nove validacije

Ime validacije mijenja se u E-mail validation, a za Validation postavke pod Type je potrebno odabrati No Rows Returned te unijeti odgovarajući SQL upit. Na slici 72. moguće je vidjeti postavke za validaciju.

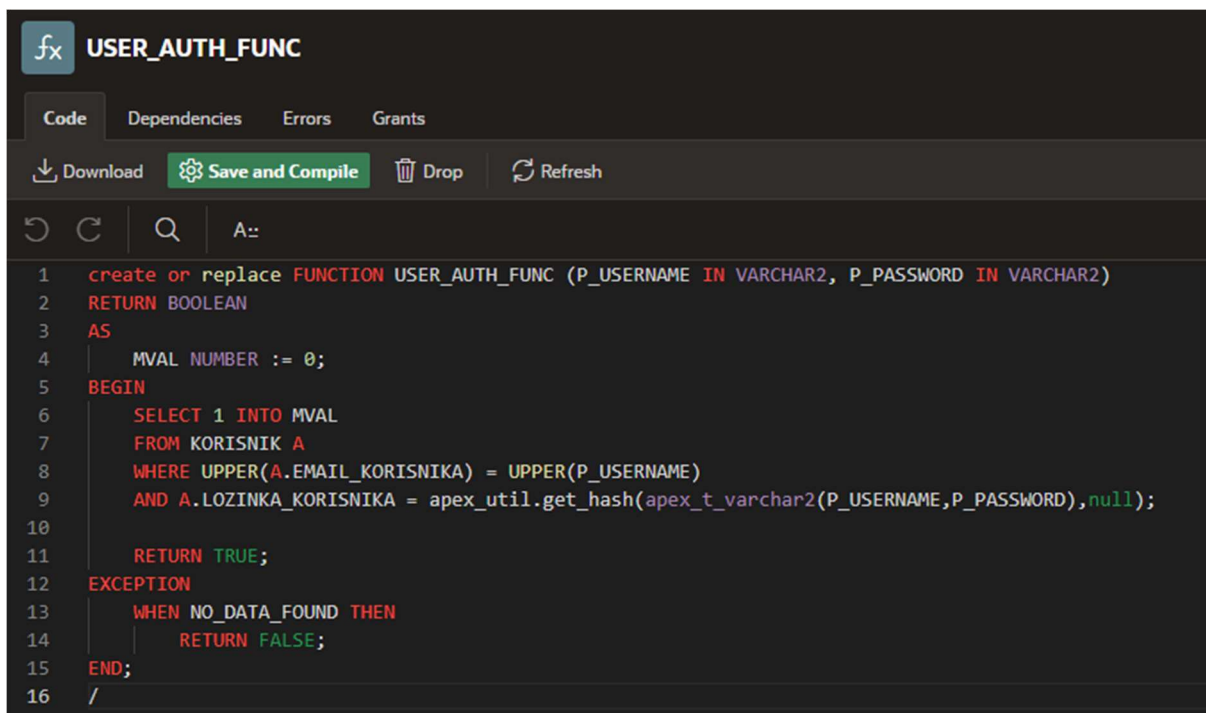
A screenshot of the configuration settings for an 'E-mail validation' rule. The settings are organized into sections: 'Identification' (Name: E-mail validation), 'Execution' (Sequence: 10), 'Validation' (Editable Region: - Select -, Type: No Rows returned, SQL Query: SELECT * FROM KORISNIK WHERE UPPER(EMAIL_KORISNIKA)=UPPER(:P9_EMAIL_KORISNIKA), Always Execute: off), and 'Error' (Error Message: Korisnik s ovom e-mail adresom već postoji, molimo pokušajte ponovno., Display Location: Inline with Field and in Notification, Associated Item: - Select -).

Slika 72 - Postavke validacije za provjeru e-mail adrese

5.2.7. Autentifikacija korisnika

Za prijavu korisnika (login) koristi se već postojeća stranica (9999 – Login Page), ali uz nekoliko promjena.

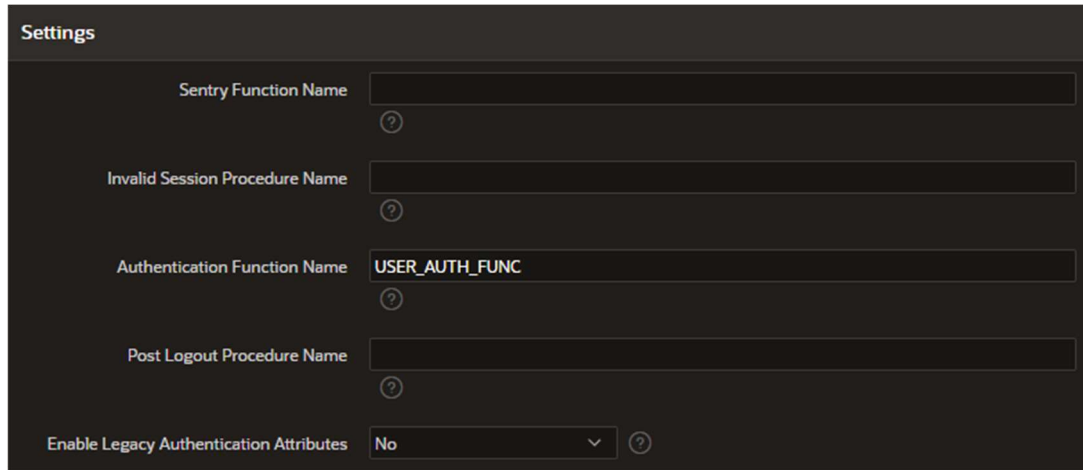
Najprije je potrebno kreirati novu PL/SQL funkciju unutar SQL Workshop → Object Browser → Functions koja se naziva USER_AUTH_FUNC , a prikazana je na slici 73. Ova PL/SQL funkcija ima svrhu autentifikacije korisnika na temelju korisničkog imena (e-mail adrese) i lozinke. Najprije se deklarira sama funkcija koja prihvaća dva ulazna parametra (P_USERNAME i P_PASSWORD), a sama funkcija vraća logičku vrijednost (true ili false). Deklarira se privremena varijabla MVAL brojanog tipa i inicijalizira se na vrijednost 0. Koristit će se za pohranu privremene vrijednosti tijekom provjere korisničkih podataka. Potom se SELECT naredbom dohvaća vrijednost iz tablice KORISNIK gdje EMAIL_KORISNIKA odgovara vrijednosti P_USERNAME (uneseni e-mail u formi za prijavi) i hashirana LOZINKA_KORISNIKA odgovara hashu koji se generira pomoću apex_util.get_hash() funkcije s ulaznim parametrima P_USERNAME i P_PASSWORD. Ako je SELECT naredba uspješna (e-mail i lozinka su ispravni) funkcija vraća vrijednost True, a u protivnom False (odnosno autentifikacija nije uspjela).



```
fx USER_AUTH_FUNC
Code Dependencies Errors Grants
Download Save and Compile Drop Refresh
A::
1 create or replace FUNCTION USER_AUTH_FUNC (P_USERNAME IN VARCHAR2, P_PASSWORD IN VARCHAR2)
2 RETURN BOOLEAN
3 AS
4     MVAL NUMBER := 0;
5 BEGIN
6     SELECT 1 INTO MVAL
7     FROM KORISNIK A
8     WHERE UPPER(A.EMAIL_KORISNIKA) = UPPER(P_USERNAME)
9     AND A.LOZINKA_KORISNIKA = apex_util.get_hash(apex_t_varchar2(P_USERNAME,P_PASSWORD),null);
10
11     RETURN TRUE;
12 EXCEPTION
13     WHEN NO_DATA_FOUND THEN
14         RETURN FALSE;
15 END;
16 /
```

Slika 73 - Funkcija USER_AUTH_FUNC

Zatim je potrebno tu funkciju i postaviti kao zadanu za autentifikaciju. Otvori se Shared Components→Security→Authentication Schemes te unutar njega se pronade Oracle APEX Accounts i otvori. U Authentication Function Name stavci odabere se prethodno kreirana funkcija i pohrane se promjene. Prikaz modificirane sheme autentifikacije prikazan je na slici 74.



Slika 74 - Modificirana shema autentifikacije

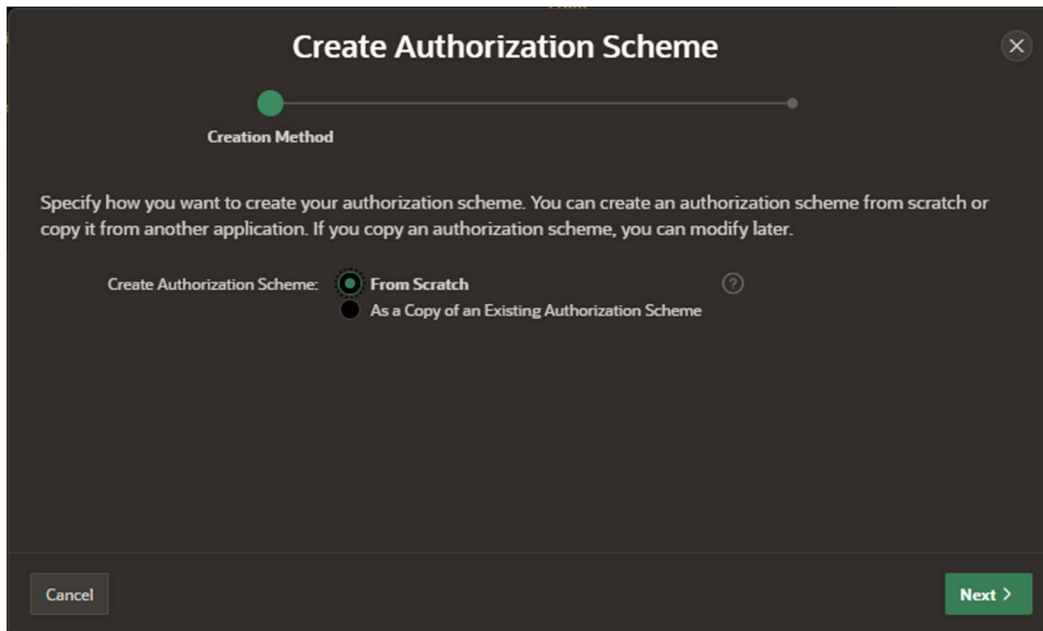
5.2.8. Autorizacija korisnika

Autorizacija korisnika je proces kojim se određuje koji resursi i usluge unutar sustava su dostupni korisnicima nakon što je njihov identitet već prethodno potvrđen, odnosno nakon što su prijavljeni sa svojim korisničkim podacima.

Oracle APEX uveliko olakšava definiranje sheme autorizacije korisnika. Za sustav rezervacije autobusnih karata potrebno je definirati 3 sheme autorizacije:

1. Autorizacija korisnika kojemu je uloga admin
2. Autorizacija korisnika kojemu je uloga putnik
3. Autorizacija korisnika kojemu je uloga gost

Za stvaranje nove autorizacijske sheme potrebno se premjestiti unutar Shared Components→Security→Authorization Schemes. Klikom na Create otvara se ekran za kreiranje nove autorizacijske sheme. Na slici 75. prikazan je ekran za izradu nove autorizacijske sheme.



Slika 75 - Izrada nove autorizacijske sheme

Klikom na Next otvara se ekran unutar kojega je moguće definirati sve nužne postavke nove autorizacijske sheme. Na slici 76. prikazane su postavke za autorizacijsku shemu „Administracija“.

Create Authorization Scheme

Use this page to define an authorization scheme. By creating an authorization schemes, you can protect applications, pages, and application components and extend the security provided by your application authentication scheme. You can use authorization schemes to identify additional security beyond simple user authentication. For example a user with administration rights may need access to more navigation bar icons, pages, and tabs than other users.

Application: **163472 Aplikacija za rezervaciju autobusnih karata**

Name: **Administrator**

Scheme Type: **Exists SQL Query**

SQL Query: `SELECT * FROM KORISNIK WHERE UPPER(EMAIL_KORISNIKA)=UPPER(:APP_USER) AND SIFRA_ULOGA=11111;`

Identify error message displayed when scheme violated: **Pristup je dozvoljen samo administratoru aplikacije!**

Validate authorization scheme: **Once per session**

Once per page view

Once per component

Always (No Caching)

Comments

Create Authorization Scheme

Slika 76 - Izrada autorizacijske sheme "Administrator"

Na ekvivalentan način kreirane su i autorizacijske sheme „GUEST“ i „PUTNIK“, razlika je samo u šifri uloge. Na slici 77. moguće je vidjeti popis svih autorizacijskih shema.

| Name | Type | Caching | Subscribed From | Subscribers | Updated | Copy |
|-----------------------|-----------------------------------|--------------------|-----------------|-------------|-------------|------|
| Administration Rights | PL/SQL Function Returning Boolean | Once per page view | | | 2 weeks ago | |
| Administrator | Exists SQL Query | Once per session | | | 6 days ago | |
| GUEST | Exists SQL Query | Once per session | | | 5 days ago | |
| PUTNIK | Exists SQL Query | Once per session | | | 5 days ago | |

Slika 77 - Popis svih autorizacijskih shema

Sada kada su autorizacijske sheme kreirane, može ih se dodijeliti stranicama ovisno o tome kome je stranica namijenjena.

Potrebno se premjestiti u Shared Components → Navigation and Search → Navigation Menu. Otvori se postojeći Navigation Menu unutar kojega se može vidjeti popis svih stranica. Pod Authorization Scheme moguće je promijeniti autorizacijsku shemu određene stranice, a kompletni popis svih stranica s pripadnim autorizacijskim shemama moguće je vidjeti na slici 78.

| Sequence | Name | Target | Icon | Authorization Scheme |
|----------|-----------------------|---|-------------------|----------------------|
| 100 | REZERVACIJA | f?p=&APP_ID.:5.&APP_SESSION.:&DEBUG::: | fa-layout-head... | PUTNIK |
| 120 | REGISTRACIJA | f?p=&APP_ID.:9.&APP_SESSION.:&DEBUG::: | fa-forms | GUEST |
| 180 | DRZAVE I MJESTA | f?p=&APP_ID.:28.&APP_SESSION.:&DEBUG::: | fa-table | Administrator |
| 200 | AUTOBUSI I SJEDALA | f?p=&APP_ID.:32.&APP_SESSION.:&DEBUG::: | fa-layout-head... | Administrator |
| 230 | POPIS REZERVACIJA | f?p=&APP_ID.:14.&APP_SESSION.:&DEBUG::: | fa-table | Administrator |
| 240 | POPIS RUTA | f?p=&APP_ID.:1.&APP_SESSION.:&DEBUG::: | fa-table | |
| 50 | Unos rezervacije | f?p=&APP_ID.:18.&SESSION.:&DEBUG::: | fa-forms | PUTNIK |
| 130 | UNOS DRZAVE POLAZISTA | f?p=&APP_ID.:22.&SESSION.:&DEBUG::: | fa-forms | Administrator |
| 140 | UNOS DRZAVE ODREDISTA | f?p=&APP_ID.:23.&SESSION.:&DEBUG::: | fa-forms | Administrator |
| 150 | UNOS MJESTA POLAZISTA | f?p=&APP_ID.:24.&SESSION.:&DEBUG::: | fa-forms | Administrator |
| 160 | UNOS MJESTA ODREDISTA | f?p=&APP_ID.:26.&SESSION.:&DEBUG::: | fa-forms | Administrator |
| 170 | UNOS AUTOBUSA | f?p=&APP_ID.:27.&SESSION.:&DEBUG::: | fa-forms | Administrator |
| 210 | UNOS SJEDALA | f?p=&APP_ID.:34.&SESSION.:&DEBUG::: | fa-forms | Administrator |
| 220 | UNOS RUTE | f?p=&APP_ID.:35.&SESSION.:&DEBUG::: | fa-forms | Administrator |

Slika 78 - Popis svih stranica sa pripadnim autorizacijskim shemama

6. Prikaz aplikacije

6.1. Prikaz stranice „POPIS RUTA“

Na slici 79. moguće je vidjeti prikaz stranice „POPIS RUTA“, unutar koje su vidljive sve rute kojima je datum polaska veći ili jednak današnjem datumu i koje je potom moguće i filtrirati po željenim vrijednostima.

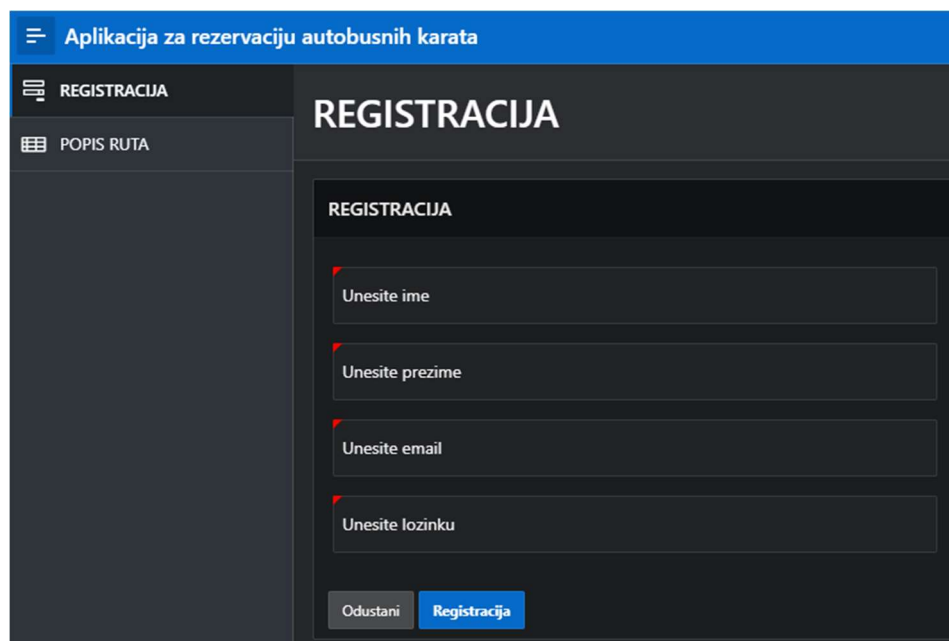


| DATUM POLASKA | DATUM DOLASKA | VRIJEME POLASKA | VRIJEME DOLASKA | CIJENA (€) | POLAZIŠTE | ODREDIŠTE |
|---------------|---------------|-----------------|-----------------|------------|----------------|-----------|
| 7/4/2024 | 7/4/2024 | 12:00 | 13:48 | 20 | Slavonski Brod | Osijek |
| 6/28/2024 | 6/28/2024 | 12:00 | 22:30 | 45 | Osijek | Rijeka |
| 6/30/2024 | 6/30/2024 | 8:00 | 11:30 | 40 | Osijek | Zagreb |
| 7/2/2024 | 7/2/2024 | 17:30 | 23:30 | 40 | Slavonski Brod | Rijeka |
| 6/26/2024 | 6/26/2024 | 11:00 | 13:30 | 20 | Rijeka | Zagreb |
| 7/4/2024 | 7/4/2024 | 6:00 | 9:00 | 35 | Zagreb | Graz |

Slika 79 - Stranica „POPIS RUTA“

6.2. Prikaz stranica „REGISTRACIJA“ i „LOG IN“

Ukoliko je unutar aplikacije korisnik prijavljen kao gost (GUEST), utoliko mu se nudi mogućnost registracije kao što je prikazano na slici 80.



REGISTRACIJA

Unesite ime

Unesite prezime

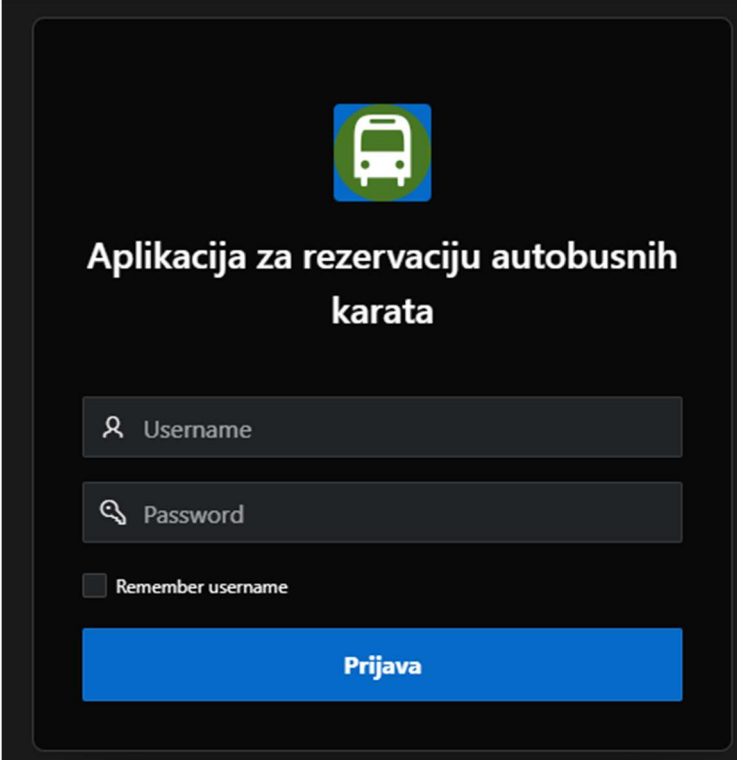
Unesite email

Unesite lozinku

Odustani Registracija

Slika 80 - Stranica „REGISTRACIJA“

Nakon registracije, korisnika se preusmjerava na stranicu za prijavu, koju je moguće vidjeti na slici 81.



Aplikacija za rezervaciju autobusnih karata

Username

Password

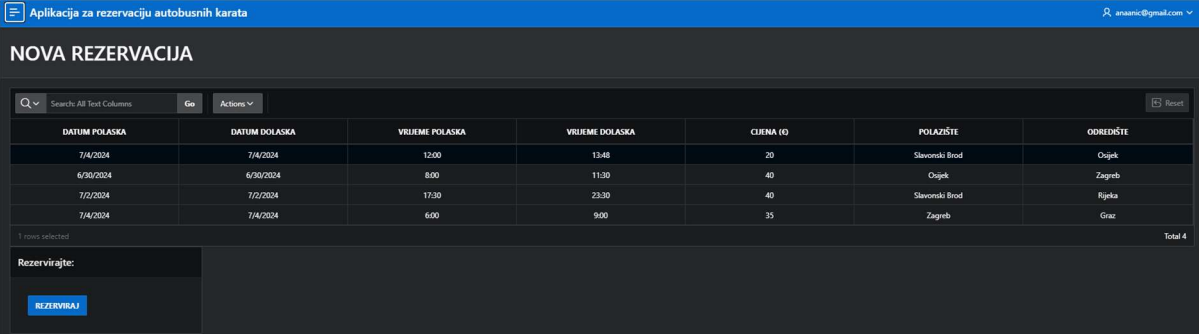
Remember username

Prijava

Slika 81 - Stranica „LOG IN“

6.3. Prikaz stranica „Nova rezervacija“ i „REZERVACIJA“

Nakon prijavljivanja sa korisničkim podacima, korisnik može stvoriti novu rezervaciju, kao što je i prikazano na slici 82. Korisnik može odabrati neku od ruta te klikom na gumb „REZERVIRAJ“ rezervirati odabranu rutu. Također je moguće i filtrirati rute (slika 83.)



NOVA REZERVACIJA

| DATUM POLASKA | DATUM DOLASKA | VRJEME POLASKA | VRJEME DOLASKA | CIJENA (€) | POLAZIŠTE | ODREĐIŠTE |
|---------------|---------------|----------------|----------------|------------|----------------|-----------|
| 7/4/2024 | 7/4/2024 | 12:00 | 13:48 | 20 | Slavonski Brod | Osijek |
| 6/30/2024 | 6/30/2024 | 8:00 | 11:30 | 40 | Osijek | Zagreb |
| 7/2/2024 | 7/2/2024 | 17:30 | 23:30 | 40 | Slavonski Brod | Rijeka |
| 7/4/2024 | 7/4/2024 | 6:00 | 9:00 | 35 | Zagreb | Graz |

Rezervirajte:

REZERVIRAJ

Slika 82 - Stranica „Nova rezervacija“

NOVA REZERVACIJA

Search: All Text Columns Go Actions Reset

Search for 'Zagreb'

| DATUM POLASKA | DATUM DOLASKA | VRIJEME POLASKA | VRIJEME DOLASKA | CIJENA (€) | POLAZIŠTE | ODREĐIŠTE |
|---------------|---------------|-----------------|-----------------|------------|-----------|-----------|
| 6/30/2024 | 6/30/2024 | 8:00 | 11:30 | 40 | Orijek | Zagreb |
| 7/4/2024 | 7/4/2024 | 6:00 | 9:00 | 35 | Zagreb | Graz |

1 rows selected Total 2

Rezervirajte:

REZERVIRAJ

Slika 83 - Filtrirani rezultati tablice RUTA

Na slici 84. prikazan je popis rezervacija trenutno prijavljenog korisnika, a klikom na gumb „PDF“ generira se potvrda o rezervaciji (slika 85.)

Aplikacija za rezervaciju autobusnih karata marib123@gmail.com

REZERVACIJA

Unos rezervacije

POPIS RUTA

PDF Save

| DATUM REZERVACIJE | EMAIL ZA PVPAL PLAĆANJE | IZNOS (€) | CIJENA RUTE (€) | UKUPNA CIJENA (€) |
|-------------------|--------------------------------|-----------|-----------------|-------------------|
| 6/14/2024 | rezervacija_placanje@gmail.com | 0,8 | 35 | 35,8 |

1 rows selected Total 1

| BROJ SIEDALA | REGISTRACIJA AUTOBUSA | KAPACITET AUTOBUSA | SADRŽAJI AUTOBUSA |
|--------------|-----------------------|--------------------|-------------------|
| 101 | FK-1344-SB | 97 | WiFi, Klima |

1 rows selected Total 1

| DATUM POLASKA | DATUM DOLASKA | VRIJEME POLASKA | VRIJEME DOLASKA | POLAZIŠTE | ODREĐIŠTE |
|---------------|---------------|-----------------|-----------------|-----------|-----------|
| 7/4/2024 | 7/4/2024 | 6:00 | 9:00 | Zagreb | Graz |

1 rows selected Total 1

Slika 84 - Stranica "REZERVACIJA"

ID REZERVACIJE: 57

DATUM REZERVACIJE: 14-06-2024



KARTA

IME I PREZIME: Mario Nikolić

DATUM I VRIJEME POLASKA: 04-07-2024 6:00h

DATUM I VRIJEME DOLASKA: 04-07-2024 9:00h

POLAZIŠTE: Zagreb

ODREDIŠTE: Graz

BROJ SJEDALA: 101

EMAIL ZA PAYPAL PLAĆANJE: rezervacija_placanje@gmail.com

CIJENA RUTE: 35€

IZNOS_REZERVACIJE: 0.8€

UKUPNA CIJENA: 35.8€

Slika 85 - PDF potvrda o rezervaciji

7. Zaključak

U ovom radu dan je pregled razvoja aplikacije za rezervaciju autobusnih karata unutar platforme Oracle APEX. Implementirane su sve funkcionalnosti na temelju specifikacije zahtjeva pomoću metodologije MIRIS.

Oracle APEX korisnicima uveliko olakšava proces razvoja poslovnih aplikacija, povećavajući pritom produktivnost rada i mogućnost fokusiranja na poslovnu logiku, a ne na tehničke značajke razvojnog procesa aplikacija.

Low code platforme poput ove imaju svoju primjenu, a to su uglavnom aplikacije kojima je glavna funkcionalnost CRUD (create, read, update, delete). Programer je ipak ograničen u svome radu, jer je određene „složenije“ funkcionalnosti teško implementirati unutar ovakvih platformi, stoga je primarni pristup kod razvijanja ipak „high code“ pristup, kod kojeg programer sam programira cjelokupnu aplikaciju.

Izrada aplikacije korištenjem Oracle APEX platforme ocjenjuje se kao pozitivno iskustvo jer pojednostavljuje cjelokupni proces razvoja aplikacija, od izgradnje baze podataka, kreiranja stranica, autentifikacije i autorizacije korisnika pa sve do samog korištenja aplikacije. Nije potrebno samostalno aplikaciju postavljati na cloud i web server jer je navedeno već integrirano unutar Oracle APEX-a što ga čini iznimno praktičnim.

Aplikaciju za rezervaciju autobusnih karata (čiji je proces izrade dan u radu) u budućnosti bi se moglo dodatno poboljšati i unaprijediti implementirajući nove značajke i funkcionalnosti. Primjerice, u planu je dodati druge vrste plaćanja (kao što je kartično plaćanje), implementirati slanje potvrde o rezervaciju na e-mail adresu korisnika i sl.

8. Literatura

- Čandrić, S. (2023a). Analiza i definicija. Uvod u programsko inženjerstvo [Moodle]. Sveučilište u Rijeci, Fakultet informatike i digitalnih tehnologija, Rijeka.
- Čandrić, S. (2023b). Goal-oriented design. Dizajn korisničkog sučelja i iskustva [Moodle]. Sveučilište u Rijeci, Fakultet informatike i digitalnih tehnologija, Rijeka.
- Pavlić, M. (2011). Oblikovanje baza podataka. Rijeka: Odjel za informatiku Sveučilišta u Rijeci.
- Pranjić, K. (2019). Znete li tko je vaša ciljana persona – te kako je definirati. Preuzeto 14.6.2024. s <https://www.cx.hr/korisnicko-iskustvo/znete-li-tko-je-vasa-ciljana-persona-te-kako-je-definirati/>
- Cohn, M. (2009). User Stories Applied: For Agile Software Development. Boston: Addison-Wesley.
- Oracle Application Express, Preuzeto: 14.6.2024. sa https://en.wikipedia.org/wiki/Oracle_Application_Express
- Panev, I., Kaluža, M. (2022). Modeliranje podataka i procesa. Rijeka: Veleučilište u Rijeci.
- Manger, R. (2010). Osnove projektiranja baza podataka. Zagreb: Sveučilište u Zagrebu.
- Brown, F. (2024). Razlika između primarnog i stranog ključa. Preuzeto 28.4.2024. s <https://www.guru99.com/hr/difference-between-primary-key-and-foreign-key.html>
- Oracle APEX (bez dat. a). About Oracle APEX. Preuzeto 2.7.2024. s <https://docs.oracle.com/en/cloud/paas/autonomous-database/serverless/adbsb/apex-intro.html>
- Oracle APEX (bez dat. b). Understanding the Workspace Home Page. Preuzeto 2.7.2024. s <https://docs.oracle.com/en/database/oracle/application-express/20.2/html/understanding-the-workspace-home-page.html>
- Oracle APEX (bez dat. c). Application Express SQL Workshop Guide. Preuzeto 2.7.2024. s <https://docs.oracle.com/database/apex-5.1/AEUTL/getting-started.htm#AEUTL29181>
- Oracle APEX (bez dat. d). App Builder. Preuzeto 2.7.2024. s <https://apex.oracle.com/en/learn/getting-started/app-builder/>

9. Popis slika

| | |
|--|----|
| Slika 1 - Persona Marko Barić..... | 4 |
| Slika 2 - Persona Marko Horvat | 5 |
| Slika 3 - Persona Veronika Kovač..... | 5 |
| Slika 4 - Grafički oblik konceptata strukture EV metode (Pavlić, 2011.)..... | 7 |
| Slika 5 - Dijagram entiteti-veze za sustav rezervacije autobusnih karata..... | 10 |
| Slika 6 - Prijava u radno okruženje..... | 13 |
| Slika 7 - Radno okruženje..... | 14 |
| Slika 8 - SQL Workshop sučelje | 15 |
| Slika 9 - Izrada tablice DRZAVA_ODREDISTA | 15 |
| Slika 10 - Nedavno kreirane tablice..... | 16 |
| Slika 11 - Tablica DRZAVA_ODREDISTA..... | 16 |
| Slika 12 - Ekran alata Object Browser..... | 16 |
| Slika 13 - Ekran Create Table..... | 17 |
| Slika 14 - Izrada tablice MJESTO_ODREDISTE | 17 |
| Slika 15 - Kreiranje ograničenja vanjski ključ unutar tablice MJESTO_ODREDISTA | 18 |
| Slika 16 - Ekran sa svim kreiranim tablicama | 19 |
| Slika 17 - Unos novog zapisa u tablicu KORISNIK..... | 19 |
| Slika 18 - App Builder sučelje..... | 20 |
| Slika 19 - Create an Application sučelje..... | 21 |
| Slika 20 - Kreiranje aplikacije za rezervaciju autobusnih karata..... | 22 |
| Slika 21 - Sučelje App Builder sa novo kreiranom aplikacijom..... | 22 |
| Slika 22 - Pokrenuta aplikacija | 23 |
| Slika 23 - Glavna stranica aplikacije | 23 |
| Slika 24 - Stvaranje nove stranice tipa "Interactive Report"..... | 24 |
| Slika 25 - Stvaranje stranice "Popis ruta" | 24 |
| Slika 26 - Sakrivanje stupca ID_RUTE..... | 24 |
| Slika 27 - Izmijenjen raspored stupaca tablice RUTA..... | 24 |
| Slika 28 - List of Values za POSTANSKI_BROJ_POLAZAZISTA | 25 |
| Slika 29 - Izmijenjen List of Values za POSTANSKI_BROJ_POLAZAZISTA | 26 |
| Slika 30 - Stranica "Popis ruta" | 26 |
| Slika 31 - Stvaranje forme za unos podataka..... | 27 |
| Slika 32 - Skriveni stupac "datum_rezervacije" | 27 |
| Slika 33 - Postavljanje defaultne vrijednosti na stupac "datum_rezervacije"..... | 28 |
| Slika 34 - Default vrijednost za polje EMAIL_ZA_PAYPAL_PLACANJE | 28 |
| Slika 35 - Default vrijednost za polje ID_KORISNIKA | 28 |
| Slika 36 - Default vrijednost za polje ID_KORISNIKA | 28 |
| Slika 37 - Stvaranje nove regije..... | 29 |
| Slika 38 - Postavke regije „Rute" | 29 |
| Slika 39 - Dinamička akcija za prosljeđivanje primarnog ključa u formu..... | 30 |
| Slika 40 - Postavke za izvršavanje dinamičke akcije „Selected" | 30 |
| Slika 41 - Postavke akcije za prosljeđivanje ID_RUTE u formu..... | 30 |
| Slika 42 - Sekvenca SEQ_REZERVACIJA_ID..... | 31 |
| Slika 43 - Okidač TRG_ID_REZERVACIJE..... | 31 |
| Slika 44 - Okidač SJEDALO_TR..... | 32 |
| Slika 45 - Kreiranje nove Master-Detail stranice | 33 |
| Slika 46 - Odabir stila Master-Detail stranice..... | 34 |
| Slika 47 - Odabir master tablice | 34 |
| Slika 48 - Odabir detail stranice | 35 |

| | |
|--|----|
| Slika 49 - Prikaz master i detail tablica | 35 |
| Slika 50 - Prikaz postavni nove detail tablice | 36 |
| Slika 51 - SQL upit za master tablicu | 36 |
| Slika 52 - SQL upit za detail tablicu SJEDALO..... | 37 |
| Slika 53 - Učitavanje AOP datoteke | 38 |
| Slika 54 - Učitani AOP dodatak | 38 |
| Slika 55 - Izvršena SQL skripta | 39 |
| Slika 56 - Dinamička akcija „Execute Javascript Code“ | 39 |
| Slika 57 - Hidden Page Item..... | 39 |
| Slika 58 - Predložak za PDF potvrdu o rezervaciji..... | 40 |
| Slika 59 - Appearance postavke gumba PRINT | 41 |
| Slika 60 - Izgled gumba PRINT | 41 |
| Slika 61 - Akcija za generiranje PDF potvrde o rezervaciji | 41 |
| Slika 62 - SQL izvor za dohvaćanje podataka za potvrdu o rezervaciji | 42 |
| Slika 63 - Postavke za izvor predloška potvrde..... | 42 |
| Slika 64 - Stvaranje nove Classic Report stranice | 43 |
| Slika 65 - Postavke stranice za pregled država i mjesta | 44 |
| Slika 66 - Prikaz svih regija za pregled država i mjesta | 44 |
| Slika 67 - Sekvenca KORISNIK_SEQ..... | 45 |
| Slika 68 - Okidač TRG_KORISNIK_ID | 45 |
| Slika 69 - PL/SQL kod procesa Registracija | 46 |
| Slika 70 - Server-side Condition postavke procesa Registracija | 46 |
| Slika 71 - Stvaranje nove validacije | 47 |
| Slika 72 - Postavke validacije za provjeru e-mail adrese..... | 47 |
| Slika 73 - Funkcija USER_AUTH_FUNC..... | 48 |
| Slika 74 - Modificirana shema autentifikacije..... | 49 |
| Slika 75 - Izrada nove autorizacijske sheme..... | 50 |
| Slika 76 - Izrada autorizacijske sheme "Administrator" | 51 |
| Slika 77 - Popis svih autorizacijskih shema..... | 52 |
| Slika 78 - Popis svih stranica sa pripadnim autorizacijskim shemama..... | 52 |
| Slika 79 - Stranica „POPIS RUTA“ | 53 |
| Slika 80 - Stranica „REGISTRACIJA“ | 53 |
| Slika 81 - Stranica „LOG IN“ | 54 |
| Slika 82 - Stranica „Nova rezervacija“ | 54 |
| Slika 83 - Filtrirani rezultati tablice RUTA | 55 |
| Slika 84 - Stranica "REZERVACIJA" | 55 |
| Slika 85 - PDF potvrda o rezervaciji | 56 |

10. Prilozi

Podaci za prijavu u sustav:

1. ADMIN: Email: admin@gmail.com, Lozinka: admin
2. GOST: Email: GUEST (lozinka prazna)

Link na aplikaciju:

<https://apex.oracle.com/pls/apex/r/nmario/aplikacija-za-rezervaciju-autobusnih-karata/login?session=110524386430780>