Implementacija aplikacije za potrebe dječjeg vrtića

Đud, Magdalena

Master's thesis / Diplomski rad

2023

Degree Grantor / Ustanova koja je dodijelila akademski / stručni stupanj: **University of Zagreb, Faculty of Organization and Informatics / Sveučilište u Zagrebu, Fakultet organizacije i informatike**

Permanent link / Trajna poveznica: https://urn.nsk.hr/urn:nbn:hr:211:299944

Rights / Prava: Attribution-NonCommercial-NoDerivs 3.0 Unported/Imenovanje-Nekomercijalno-Bez prerada 3.0

Download date / Datum preuzimanja: **2024-05-10**

SVEUČILIŠTE U ZAGREBU
FAKULTET ORGANIZACIJE I INFORMATIKE
VARAŽDIN

Repository / Repozitorij:

Faculty of Organization and Informatics - Digital Repository

zir.nsk.hr

DIGITALNI AKADEMSKI ARHIVI I REPOZITORIJI

SVEUČILIŠTE U ZAGREBU

FAKULTET ORGANIZACIJE I INFORMATIKE

V A R A Ž D I N

Magdalena Đud

IMPLEMENTACIJA APLIKACIJE ZA POTREBE DJEČJEG VRTIĆA

DIPLOMSKI RAD

Varaždin, 2023.

SVEUČILIŠTE U ZAGREBU

FAKULTET ORGANIZACIJE I INFORMATIKE

V A R A Ž D I N

Magdalena Đud

JMBAG: 0016135660

Studij: Informacijsko i programsko inženjerstvo

IMPLEMENTACIJA APLIKACIJE ZA POTREBE DJEČJEG VRTIĆA

DIPLOMSKI RAD

Mentor:

Prof. dr. sc. Kornelije Rabuzin

Varaždin, svibanj 2023.

Magdalena Đud

Izjava o izvornosti

Izjavljujem da je moj diplomski rad izvorni rezultat mojeg rada te da se u izradi istoga nisam koristio drugim izvorima osim onima koji su u njemu navedeni. Za izradu rada su korištene etički prikladne i prihvatljive metode i tehnike rada.

Autorica potvrdila prihvaćanjem odredbi u sustavu FOI-radovi

Sažetak

Cilj ovog rada implementirati je dio ERP sustava za dječje vrtiće u obliku *web* aplikacije vezane uz ulogu ravnatelja i odgojitelja. Međutim, najprije će se objasniti osnovni pojmovi vezani uz ERP sustave općenito i ERP sustave prilagođene dječjim vrtićima. Svi pojmovi biti će dodatno popraćeni kratkim opisom primjera postojećih ERP sustava za dječje vrtiće te nešto detaljnijom analizom preuzete demo verzije jednog. Pošto je implementirana aplikacija rađena u .NET C#, a baza podataka u sustavu MS SQL Server, oni će biti kratko objašnjeni. Također, biti će prikazana njihova postava potrebna prije same implementacije aplikacije. Glavni, praktični, dio rada sadrži opis aplikacijske domene, ERA dijagrame vezane uz pojedine funkcionalnosti aplikacije, prikazanu bazu podataka rađenu prema ERA dijagramima, opis korisničkog sučelja i njegovog korištenja te dokumentaciju programskog koda same aplikacije.

Ključne riječi: ERP sustav; *web* aplikacija; dječji vrtić; baza podataka; MS SQL Server; MS SQL Server Management Studio; .NET C#; MS Visual Studio;

Sadržaj

Uvod 1
Metode i tehnike rada 2
Što je ERP sustav? 3
1.1. ERP sustav kao *web* aplikacija 4
1.2. Popularni ERP sustavi 4
1.3. ERP sustavi u dječjim vrtićima 5
1.3.1. Primjeri ERP sustava za dječje vrtiće 6
1.3.2. Analiza Next ERP Preschool sustava 7
Razvoj ERP sustava 9
1.4. MS SQL Server 9
1.5. ASP.NET Core Web App (MVC) 10
Praktični rad 14
1.6. Opis aplikacijske domene 14
1.7. Izrada ERA modela 15
1.7.1. ERA model *Korisnici aplikacije* 15
1.7.2. ERA model *Šifrarnici* 16
1.7.3. ERA model *Djeca* 18
1.7.4. ERA model *Sanitarni nadzor* 20
1.7.5. ERA model *Nazočnost djece* 21
1.7.6. ERA model *Sukobi među djecom* 22
1.7.7. ERA model *Sukobi među roditeljima* 23
1.7.8. ERA model *Ozljede djece* 24
1.7.9. ERA model *Antropometrijska mjerenja djece* 25
1.7.10. ERA model *Alergija djece* 26
1.7.11. ERA model *Individualni sastanci* 26
1.7.12. ERA model *Dokumenti djece* 27
1.7.13. ERA model *Program stručnog usavršavanja odgojitelja* 28
1.8. Izrada baze podataka 30
1.8.1. Dio baze podataka *Korisnici aplikacije* 30
1.8.2. Dio baze podataka *Šifrarnici* 31
1.8.3. Dio baze podataka *Djeca* 32
1.8.4. Dio baze podataka *Sanitarni nadzor* 34
1.8.5. Dio baze podataka *Nazočnost djece* 35
1.8.6. Dio baze podataka *Sukobi među djecom* 37
1.8.7. Dio baze podataka *Sukobi među roditeljima* 38
1.8.8. Dio baze podataka *Ozljede djece* 39
1.8.9. Dio baze podataka *Antropometrijska mjerenja djece* 41

1.8.10. Dio baze podataka *Alergije djece* 42
1.8.11. Dio baze podataka *Individualni sastanci* 43
1.8.12. Dio baze podataka *Dokumenti djece* 45
1.8.13. Dio baze podataka *Program stručnog usavršavanja odgojitelja* 46
1.9. Korisničko sučelje i korištenje aplikacije 48
1.9.1. Dio aplikacije za korisničku ulogu *Admin* 50
1.9.1.1. Upravljanje korisnicima aplikacije 50
1.9.1.2. Upravljanje djecom 51
1.9.1.3. Upravljanje šifrarnicima 53
1.9.2. Dio aplikacije za korisničku ulogu *Ravnatelj* 56
1.9.2.1. Matična knjiga djece 56
1.9.2.2. Upravljanje skupinama djece 58
1.9.2.3. Evidencija Sanitarnog nadzora 58
1.9.3. Dio aplikacije za korisničku ulogu *Odgojitelj* 60
1.9.3.1. Imenik djece 61
1.9.3.2. Evidencija nazočnosti djece 62
1.9.3.3. Evidencija sukoba 64
1.9.3.4. Evidencija ozljeda djece 68
1.9.3.5. Zdravstveni kartoni djece 71
1.9.3.6. Evidencija antropometrijskog mjerenja djece 72
1.9.3.7. Alergije djece 73
1.9.3.8. Evidencija individualnih sastanaka s roditeljima 75
1.9.3.9. Program stručnog usavršavanja odgojitelja 78
1.9.3.10. Brzi pregled djece 80
1.10. Izrada aplikacije 85
1.10.1. Prijava i autorizacija korisnika 87
1.10.2. Tablice 89
1.10.3. Filteri 92
1.10.4. Forme 95
1.10.4.1. Dodavanje 95
1.10.4.2. Uređivanje 97
1.10.4.3. Brisanje 98
1.10.4.4. Forme unutar tablice 99
1.10.5. Generiranje PDF-a 102
Zaključak 105
Popis literature 106
Popis slika 108
Popis programskih kodova 112

Uvod

Potreba za digitalizacijom u poslovanju sve više i više raste u svim sektorima rada, a tako i kod dječjih vrtića koji predstavljaju središte zanimanja ovog diplomskog rada. Samom digitalizacijom nastoji se automatizirati te olakšati administrativni i drugi rad zaposlenika. Konkretno gledano kod dječjih vrtića, cilj je omogućiti odgojiteljima više vremena u provođenju s djecom i smanjiti vrijeme rada nad administracijom i pripremom materijala.

Pojam ERP (engl. *Enterprise Resource Planning*) sustava veže se uz digitalizaciju te, osim automatizacije rada, nudi i pristup podacima na jednom mjestu. Također, označava jedinstvo svih funkcionalnosti poslovanja i time olakšava te ubrzava rad. Zbog toga je potražnja za ERP sustavima veća nego ikad. U ovom će se radu obraditi problematika razvoja optimalnog ERP sustava za određenu vrstu poslovanja, u ovom slučaju dječjeg vrtića. Biti će analizirani općeniti zahtjevi dječjih vrtića za funkcionalnostima ERP sustava koje bi zadovoljili njihove potrebe. Kako bi se dodatno ušlo u problematiku ERP sustava za dječje vrtiće, raditi će se analiza jednog već postojećeg ERP sustava za dječje vrtiće dostupnog na stranom tržištu.

Motivacija za pisanje rada na ovu temu nepostojanje je jedinstvenog ERP sustava za dječje vrtiće na hrvatskom tržištu. Tijekom stručne prakse u tvrtki Dubal d.o.o. jedan dječji vrtić u Varaždinskoj županiji zatražio je baš to, sve potrebne funkcionalnosti na jednom mjestu. Uz pomoć njihovih zahtjeva razvijen je i dio aplikacije realizirane u ovom diplomskom radu. Osim što se razvojem ove aplikacije digitaliziraju i automatiziraju već postojeće radnje unutar dječjih vrtića, prate se i zakonske odredbe te uzimaju u obzir moguće buduće potrebe.

Metode i tehnike rada

Za izradu ovog diplomskog rada korištene su razne metode i tehnike rada. Za njihov lakši opis, rad se može podijeliti na dva dijela: obrada problematike same teme rada i izrada vlastite aplikacije.

Za prvi dio koristile su se tehnike pronalaska literature (izvora) uz pomoć kojih je obrađen teorijski aspekt rada. Obrađena teorija popraćena je analizom jednog postojećeg ERP sustava za dječje vrtiće.

Za potrebe izrade ERA dijagrama aplikacije korišten je *online* alat draw.io, baza podataka izrađena je u Microsoft SQL Server-u korištenjem Microsoft SQL Server Management Studia, dok je aplikacija razvijena u ASP.NET Core Web App (MVC) korištenjem Microsoft Visual Studia.

Alat draw.io aplikacija je otvorenog izvornog koda za izradu dijagrama krajnjeg korisnika. Pristupa mu se *online* putem preglednika te omogućuje izradu dijagrama raznih vrsta, poput ERA dijagrama (engl. *Entity Relationship Diagram*), slijednog dijagrama (engl. *Sequence Diagram*), dijagrama toka (engl. *Flowchart Diagram*) i mnogih drugih [1].

Microsoft SQL Server sustav namijenjen je upravljanju relacijskim bazama podataka (RDBMS) koji podržava širok raspon obrade transakcija, poslovne inteligencije i analitičkih aplikacija u korporativnim IT okruženjima. Izgrađen je na temelju SQL-a, standardiziranog programskog jezika koji administratori baza podataka i drugi IT stručnjaci koriste za upravljanje bazama podataka i postavljanje upita [2]. SQL Server Management Studio (SSMS) integrirano je okruženje za upravljanje bilo kojom SQL infrastrukturom. SSMS pruža jedan sveobuhvatni uslužni program koji kombinira široku grupu grafičkih alata s uređivačima SQL skripti [3].

Microsoft Visual Studio integrirano je razvojno okruženje koje omogućuje razvoj, pisanje i uređivanje koda. Dopušta korištenje gotovo bilo kojeg jezika kodiranja, između ostalog i C# koji je korišten za potrebe ovog diplomskog rada [4]. Točnije, korišten je ASP.NET Core MVC. On je bogati okvir za izradu *web* aplikacija i API-ja pomoću uzorka dizajna *Model-View-Controller* [5].

Što je ERP sustav?

Planiranje resursa poduzeća (engl. *Enterprise Resource Planning* - ERP) platforma je koju tvrtke koriste za upravljanje i integraciju bitnih dijelova svog poslovanja. Mnoge ERP softverske aplikacije ključne su za tvrtke jer im pomažu implementirati planiranje resursa integracijom svih procesa unutar jednog sustava [6].

Povijesno gledano, ERP sustave prvenstveno su koristile proizvodne i distribucijske tvrtke, budući da su doticali aspekte u operacijama opskrbnog lanca. Ova su se rješenja razvila iz različitih aplikacija opskrbnog lanca kako bi odgovorila na šire računovodstvene i financijske zahtjeve svih poduzeća, zajedno s funkcionalnostima specifičnim za određenu struku [7]. Slika 1. prikazuje module ERP sustava kao što su upravljanje ljudskim resursima, planiranje, upravljanje skladištem, financijama, *reporting* i sl.

Slika 1. Moduli ERP sustava

(Izvor: [7])

1.1. ERP sustav kao *web* aplikacija

Prilikom planiranja izrade ERP sustava, između ostalog, potrebno je odabrati vrstu aplikacije (mobilna, *desktop*, *web*). U nastavku će biti objašnjene prednosti i nedostaci odabira *web* aplikacije koji je i izabran za potrebe ovog diplomskog rada.

Prednosti *web* aplikacija:

- Nema preuzimanja i instalacije (laki pristup aplikaciji)
- Kompatibilnost s više platformi
- Samoažurirajuća rješenja
- Manje opterećenje sustava

Nedostaci *web* aplikacija:

- Potreba za internetom
- Ponekad sporija izvedba
- Sigurnosni problemi [8]

Može se zaključiti da ERP kao *web* aplikacija ima i negativne i pozitivne strane te da se odabir, gotovo pa isključivo, zasniva na potrebama i željama korisnika. Kao jedan od većih razloga sve češćeg odabira *web* aplikacija naspram *desktop* aplikacija, jest bolja dostupnost sustava s bilo kojeg uređaja u bilo koje vrijeme [9].

Na konkretnom primjeru ovog diplomskog rada, odabir *web* aplikacije čini se kao najbolja opcija zbog opisa posla samih odgojitelja. Naime, odgojitelji osim što rade s djecom, bilježe i sve što se s njima događa tijekom boravka u vrtiću. U više slučajeva odgojitelj mora čim lakše i brže pristupiti sustavu, ali i imati pristup nakon radnog vremena kad to situacija zahtijeva.

1.2. Popularni ERP sustavi

Danas nema tko nije čuo ili bio u doticaju s nekim ERP sustavom. ERP sustava ima nebrojivo mnogo, ali su se ipak neki probili više od ostalih te dobili titule „najboljih“ i „najpopularnijih“. U nastavku će biti izdvojeni i opisani top pet najboljih poslovnih ERP sustava, a to su SAP, Oracle, Workday, Kadulja i Infor [10].

SAP ERP je softver za planiranje resursa poduzeća koji je razvila tvrtka SAP SE. ERP ili softver za planiranje resursa poduzeća modularni je softver izrađen za integraciju glavnih funkcija osnovnih poslovnih procesa organizacije u objedinjeni sustav. SAP-ovi različiti ERP proizvodi omogućuju njegovim klijentima vođenje poslovnih procesa, uključujući računovodstvo, prodaju, proizvodnju, ljudske resurse i financije, u integriranom okruženju, s podacima iz svakog modula pohranjenim u središnjoj bazi podataka. ECC ostaje SAP-ov

vodeći ERP i temelj proizvoda sljedeće generacije, S/4HANA [11]. Cijene SAP ERP sustava kreću se od 56 do 220 dolara mjesečno po korisniku.

Oracleov ERP softver nudi rješenje za gotovo svaku industriju i tvrtku bilo koje veličine. Poduzeća koja implementiraju ERP iskuse bolji pristup točnim analitikama, povezivanje različitih sustava, bolje upravljanje rastom poslovanja, poboljšano zadovoljstvo korisnika i smanjenje troškova poslovanja. Oracle ERP Cloud, NetSuite i Oracle JD Edwards EnterpriseOne vodeći su Oracle ERP proizvodi na današnjem tržištu [12]. Cijene Oracle ERP sustava kreću se od 80 do 175 dolara mjesečno po korisniku.

Workday ERP služi upravljanju resursima poduzeća koji se koristi za poslovne procese i transformaciju podataka u cijelom poduzeću. Većina kupaca Workdaya su srednje i velike tvrtke i organizacije. Proizvodi ERP sustava Workday uključuju *Planiranje poduzeća*, *Financijsko upravljanje*, *Upravljanje potrošnjom*, *Analitika i izvješćivanje*, *Upravljanje ljudskim kapitalom* (HCM), *Upravljanje talentima*, *Obračun plaća i upravljanje radnom snagom*, *Automatizacija profesionalnih usluga* (PSA), *Student* i *Proširenja platforme i proizvoda* (PPE) [13]. Cijene Workday ERP sustava kreću se od 99 dolara mjesečno pa na više po korisniku.

Kadulja 200 cjelovito je integrirano ERP rješenje za upravljanje poslovanjem za potpunu kontrolu poslovanja. Omogućuje upravljanje projektima za kontrolu planiranja s financijskim i komercijalnim upravljanjem, mogućnost postprodajnog upravljanja i skladišta, integraciju s Microsoft 365, vezu s bankama, internetsku trgovinu te *online* dostupnost s pristupom s bilo kojeg mjesta i s bilo kojeg uređaja [14]. Cijene Kadulja ERP sustava kreću se od 433,50 do 1207,50 dolara godišnje.

Infor ERP softver je za upravljanje računovodstvom i financijama, lancem opskrbe i za upravljanje poslovnim procesima u mnogim funkcionalnim područjima za cijelu tvrtku s više korisnika u oblaku. Infor ERP proizvodi temeljeni na oblaku uključuju Infor CloudSuite Financije (i upravljanje opskrbom), Infor LN, Infor M3, Infor CloudSuite Industrial (SyteLine) i Infor Distribution SX.e. Infor poslovni softver izgrađen je na platformi oblaka Amazon Web Services (AWS), ali nudi i lokalni softver [15]. Cijene Infor ERP sustava kreću se od 150 do 200 dolara mjesečno po korisniku.

1.3. ERP sustavi u dječjim vrtićima

Kako bi jedan ERP sustav bio dostatan za potrebe dječjih vrtića potrebno je obratiti pažnju na nekoliko izdvojenih značajki [16]:

a) *online* ili *offline* dostupnost:

Kao što je u prethodnom poglavlju spomenuto, postoje dvije opcije dostupnosti aplikacije, *online* i *offline*. *Offline* model ERP sustava je koristan ako vrtić nema 24-

satni pristup internetu, dok je *online* model koristan jer korisnicima omogućuje pristup alatu s bilo kojeg mjesta i u bilo koje vrijeme.

b) **tehnološke mogućnosti:**
   Aplikacija bi trebala zadovoljavati potrebe dječjeg vrtića, pratiti zakonski određene funkcionalnosti i prilagoditi se željama samog klijenta. Konkretno, dio *web* aplikacije ERP sustava za vrtiće, razvijen u sklopu ovog diplomskog rada, rađen je prema potrebama i zahtjevima jednog privatnog vrtića u Varaždinskoj županiji te, između ostalog, zatražene funkcionalnosti prate i zakonski određene radnje te obrasce propisane za dječje vrtiće. Funkcionalnosti će biti detaljnije opisane u praktičnom dijelu rada.
c) **jednostavnost korištenja:**
   Aplikacija bi trebala biti jednostavna i intuitivna za korištenje te bi se sve radnje, za koje je to moguće, trebale automatizirati i na taj način olakšati posao zaposlenicima dječjih vrtića.
d) **skalabilnost:**
   Aplikacija bi trebala biti skalabilna, tj. imati prostora za nadogradnju.

1.3.1. Primjeri ERP sustava za dječje vrtiće

Sustav upravljanja dječjim vrtićima nezaobilazan je alat za stvaranje idealnog obrazovnog okruženja za djecu. Na stranom tržištu postoje mnogi različiti sustavi upravljanja, a neki od njih biti će kratko opisani u nastavku [17].

Mona eLMS Sustav upravljanja obrazovanjem u dječjim vrtićima

Mona eLMS je aplikacija čija je najveća prednost njegova sposobnost da skrati komunikacijski jaz između dječjih vrtića i roditelja. Osim toga, softver također omogućuje stranama da međusobno kontinuirano ažuriraju informacije. Neke druge značajke Mona eLMS aplikacije su: elektroničke kontakt knjige roditelja, izravna razmjena poruka između roditelja i odgojitelja, upravljanje nastavnim planom i programom, upravljanje planom aktivnosti, upravljanje zdravljem djece itd.

KidsOnline softverski sustav za upravljanje predškolskim odgojem

KidsOnline softver aplikacija je za upravljanje dječjim vrtićima specijalizirana za financijski aspekt. Također, pomaže vrtićima i roditeljima da lakše komuniciraju jedni s drugima te pomaže odgojiteljima u vođenju aktivnosti djece.

PMS Sustav upravljanja vrtićem

PMS softver optimizira upravljanje učenjem i igrom djece zahvaljujući sveobuhvatno razvijenom sustavu. Osim toga, aplikacija također podržava vrtiće u izradi hranjivih obroka za djecu.

Softver za upravljanje predškolskim odgojem BimBon

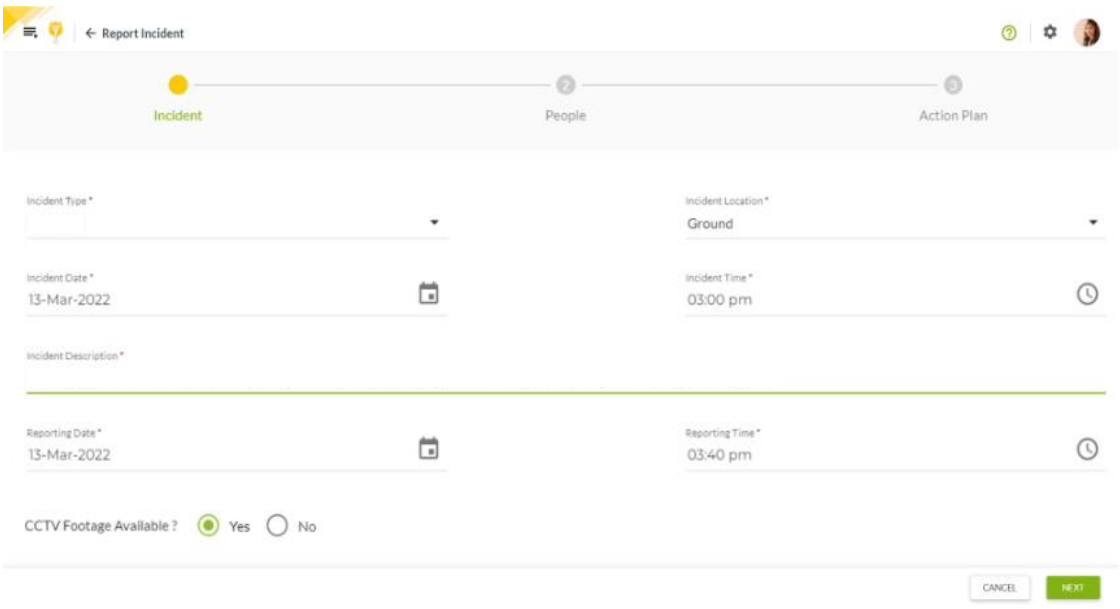
Softver za upravljanje dječjim vrtićem BimBom integriran je s modernim tehnološkim sustavom koji pojednostavljuje upravljanje aktivnostima učenja i igre djece. Softver ne samo da pomaže korisnicima u upravljanju dječjim obrocima i rasporedom, već također može pomoći u financijskom upravljanju te interakciji s odgojiteljima i roditeljima.

1.3.2. Analiza Next ERP Preschool sustava

Next ERP Preschool sustav nudi preuzimanje besplatne demo verzije koja je preuzeta za potrebe ovog istraživačkog teorijskog dijela diplomskog rada [18]. Aplikacija Next Preschool ERP je platforma za upravljanje predškolskim obrazovanjem temeljena na oblaku koja ima funkcionalnosti izvješćivanja o incidentima, evidencije dopuštenja roditelja i podrške za hibridnu učionicu, detaljnije opisanim u nastavku. Neke od ostalih funkcionalnosti aplikacije su: evidencija djece, upravljanje programom prehrane, upravljanje dokumentima, upravljanje nastavnim planom i programom i sl.

Funkcionalnost izvješćivanja o incidentima prikazana je na slici 2. Sastoji se od tri koraka:

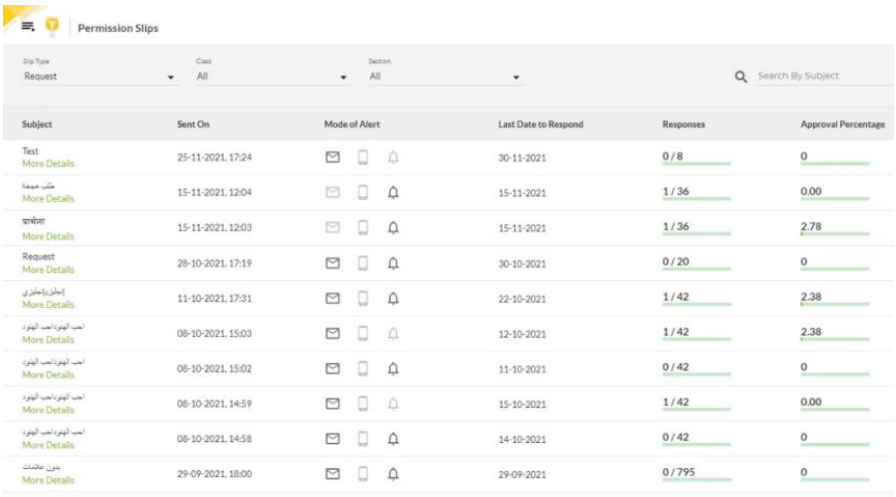
1.) *Incident* - ispunjavanje osnovnih podataka o problemu (tip problema, datum, vrijeme, mjesto, opis, datum prijave problema i sl.)
2.) *People* - ispunjavanje podataka o djeci i svjedocima incidenta
3.) *Action Plan* - upis posljedica incidenta i plana njegovog saniranja



Slika 2. Funkcionalnost izvješćivanja o incidentima - Next ERP Preschool (Izvor: [18])

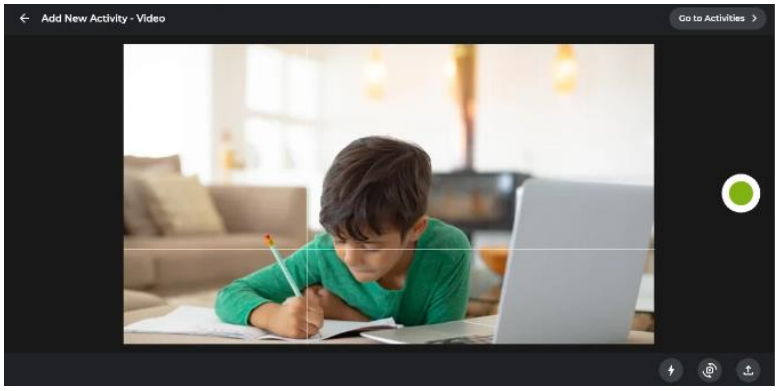
Dio funkcionalnosti evidencije dopuštenja roditelja s prikazom tablice za njezino nadgledanje prikazan je na slici 3. Tablica za nadgledanje stanja dopuštenja sastoji se od šest atributa:

1.) *Subject* - naziv dopuštenja i poveznica na ekran s više detalja
2.) *Sent On* - datum i vrijeme slanja zahtjeva za dopuštenje
3.) *Mode of Alert* - način slanja zahtjeva za dopuštenjem (e-mail, sms, obavijest na aplikaciji)
4.) *Last Date to Respond* - rok za odgovor na zahtjev za dopuštenjem
5.) *Responses* - broj odgovorenih zahtjeva za dopuštenjem
6.) *Approval Percentage* - postotak potvrdnih odgovora na zahtjev za dopuštenjem



Slika 3. Funkcionalnost evidencije dopuštenja roditelja - Next ERP Preschool (Izvor: [18])

Dio funkcionalnosti s podrškom za hibridnu učionicu prikazan je na slici 4. Spomenuta funkcionalnost omogućuje dodavanje aktivnosti, videa, slika i drugih materijala za potrebe hibridnog sudjelovanja djeteta.



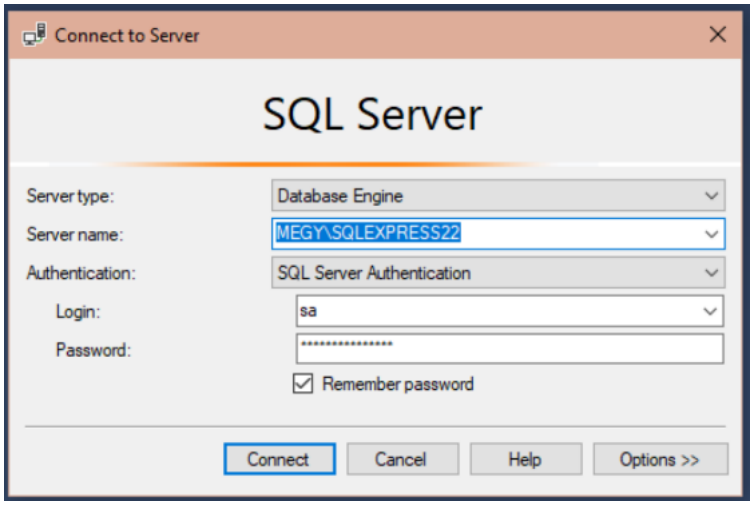
Slika 4. Funkcionalnost podrške za hibridnu učionicu - Next ERP Preschool (Izvor: [18])

1.4. MS SQL Server

MS SQL Server je sustav za upravljanje relacijskim bazama podataka (RDBMS) koji je razvio i plasirao Microsoft. Kao poslužitelj baze podataka, primarna funkcija MS SQL Servera je pohranjivanje i dohvaćanje podataka koje koriste druge aplikacije [19].

MS SQL Server radi isključivo na Windows okruženju više od 20 godina. Microsoft ga je 2016. učinio dostupnim na Linuxu. MS SQL Server 2017 postao je općenito dostupan u listopadu 2016. i radio je na Windowsima i Linuxu. Za upravljanje podacima, MS SQL Server uključuje MS SQL Server Integration Services (SSIS), MS SQL Server Data Quality Services i MS SQL Server Master Data Services. Za razvoj baza podataka, MS SQL Server nudi alate MS SQL Server Data, a za upravljanje, implementaciju i nadzor baza podataka MS SQL Server ima MS SQL Server Management Studio (SSMS) [20].

SSMS korišten je u svrhe ovog diplomskog rada, a instaliran je prema [21]. Zatim je kreiran server vidljiv na slici 5. Postoje dva načina prijave u server, SQL Windows Autentification i SQL Server Authentification. Na slici je prikazan način prijave na server uz pomoć *User ID*-a i lozinke, tj. SQL Server Authentification. Nakon konekcije na server, otvorena je novu baza (*Database*) naziva DV (*Databases* -> desni klik miša -> *New Database…*). Kreiranje tablica, *View*-a i funkcija biti će objašnjeno u praktičnom dijelu rada.



Slika 5. Povezivanje na MS SQL Server

(Izvor: vlastita izrada)

1.5. ASP.NET Core Web App (MVC)

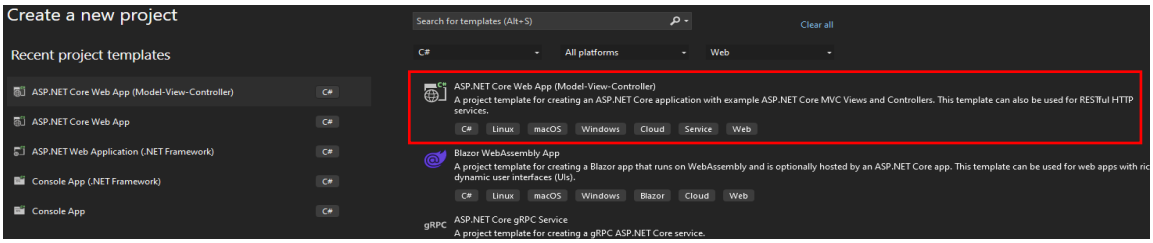
ASP je kratica za *Microsoft Active Server Pages*, što je Microsoftov prvi skriptni stroj na strani poslužitelja koji je omogućio dinamički generirane *web* stranice. Na visokoj razini, ASP je jedna od nekoliko tehnologija za razvoj *web* aplikacija *Internet Information Service* (IIS) [22]. Korištenjem .Net *Framework*-a mogu se razviti razne vrste aplikacija poput [23]:

a) *web* aplikacija - pokreću se unutar *web* poslužitelja kako bi ispunili zahtjeve korisnika putem HTTP-a
b) *web* usluge - funkcionalnosti *web* poziva dostupnih kroz industrijske standarde kao što su SOAP, XML i HTTP
c) Windows aplikacija - aplikacije za stolna računala koje se mogu koristiti za dnevne zadatke
d) Windows usluge - dugotrajne izvršne aplikacije koje se izvode kao pozadinski procesi u sustavu, ne ometaju druge aplikacije koje se izvode na istom računalu
e) konzolna aplikacija - lagani programi koji se mogu pokrenuti unutar DOS naredbenog retka sustava Windows
f) igre, mobilne aplikacije i dr.

Za potrebe ovog diplomskog rada izabrana je *web* aplikacija s MVC obrascem. *Model-View-Controller* (MVC) arhitektonski je obrazac koji dijeli aplikaciju u tri glavne skupine komponenti: modeli, *view*-evi i kontroleri. Načelo iza razvoja MVC-a je odvajanje zadaća. Ovo načelo podrazumijeva da softver treba klasificirati na temelju vrste zadatka koji obavlja. Razdvajanje osigurava da je poslovni model jednostavan za testiranje i da se može mijenjati bez previše uplitanja u detalje implementacije niske razine. Korištenjem ovog MVC obrasca, korisnički zahtjevi se usmjeravaju do kontrolera, koji je zadužen za interakciju s modelom za provođenje radnji korisnika i primanje rezultata upita [5].

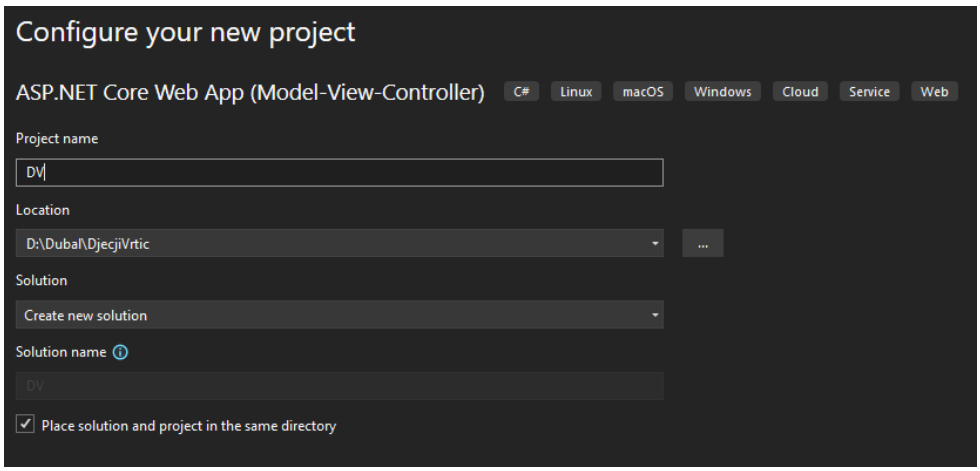
Za otvaranje i postavljanje novog projekta (engl. *Solution*) u MS Visual Studiu potrebno je pratiti sljedeće korake [24]:

1. *Create a new project* -> izabrati okruženje (u ovom slučaju: ASP.NET Core Web App (*Model-View-Controller*) -> *Next*



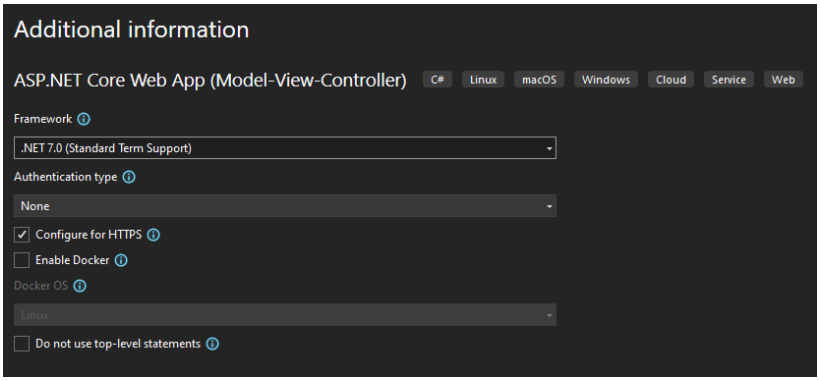
Slika 6. Create a new project
(Izvor: vlastita izrada)

2. *Configure your new project* (upisati ime i lokaciju projekta) -> *Next*



Slika 7. Configure a new project
(Izvor: vlastita izrada)

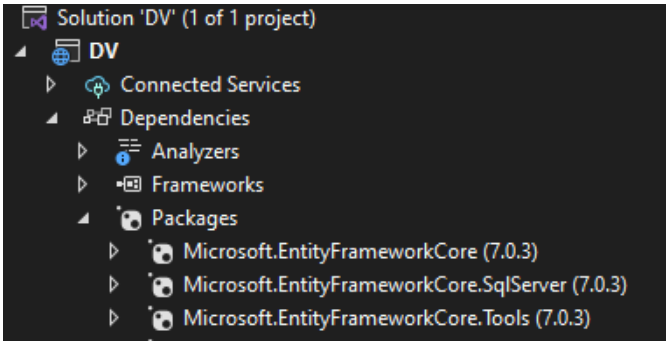
3. *Additional information* (izabrati zadnju verziju *Framework*-a) -> *Create*



Slika 8. Additional information
(Izvor: vlastita izrada)

4. Instalirati pakete (engl. *Packages*) potrebne za razvoj aplikacije (za potrebe ovog diplomskog rada potrebno je bilo instalirati, osim navedenih, još neke pakete koji će detaljnije biti objašnjeni u praktičnom dijelu rada):
    - Microsoft.EntityFrameworkCore (verzija koja se poklapa s verzijom projekta)
    - Microsoft.EntityFrameworkCore.SqlServer (verzija koja se poklapa s verzijom projekta)

- Microsoft.EntityFrameworkCore.Tools (verzija koja se poklapa s verzijom projekta)



Slika 9. *Packages*

(Izvor: vlastita izrada)

Za ostvarenje veze između projekta (engl. *Solution*) u MS Visual Studiu i baze u MS SQL Serveru potrebno je pratiti sljedeće korake [25]:

1. U `appsettings.json` dodati sljedeće linije programskog koda koje sadrže *string* za konekciju s bazom podataka:

```
"AllowedHosts": "*",
  "ConnectionStrings": {
    "DefaultConnection": "Data
Source=MEGY\\SQLEXPRESS22;Initial Catalog=DV;Integrated
Security=True;"
  }
```

Programski kod 1: *Appsettings.json* - Povezivanje s bazom podataka

(Izvor: vlastita izrada)

2. U `Program.cs` dodati sljedeće linije programskog koda koje predstavljaju postavu konteksta za kreiranje upita nad bazom podataka i tako omogućavaju njezino upravljanje:

```
builder.Services.AddDbContext<DvContext>(options =>
options.UseSqlServer(
builder.Configuration.GetConnectionString("DefaultConnection")
    ));
```

Programski kod 2: *Program.cs* - Povezivanje s bazom podataka

(Izvor: vlastita izrada)

3. U projekt dodati mapu *Models* u koju će se migrirati tablice iz baze podataka u obliku modela i kreirati kontekst spomenut u prethodnom koraku (kasnije će se dodati i još neki modeli objašnjeni u praktičnom dijelu rada).

4. U *Package Manage Console* upisati sljedeću komandu za migraciju tablica iz baze podataka u projekt i stvaranje konteksta [26]:

```
Scaffold-DbContext "Server=MEGY\SQLEXPRESS22;Database=DV;User
ID=sa;password=*****;TrustServerCertificate=true"
Microsoft.EntityFrameworkCore.SqlServer -OutputDir Models -
Force
```

Programski kod 3: Migracija baze podataka

(Izvor: vlastita izrada)

Primjer dijela modela Alergije:

```
…
public int Idalergija { get; set; }
public DateTime? AlergijaDatum { get; set; }
public string? AlergijaOpis { get; set; }
…
```

Programski kod 4: Model *Alergije*

(Izvor: vlastita izrada)

Praktični rad

1.6. Opis aplikacijske domene

Cilj praktičnog dijela rada bila je izrada ERP sustava u obliku *web* aplikacije za dječje vrtiće s naglaskom na potrebe korisničkih uloga *Ravnatelj* i *Odgojitelj*. Osim spomenutih uloga, realizirana je i uloga *Admin* koja predstavlja administratora aplikacije. Aplikacija je rađena u suradnji s tvrtkom Dubal d.o.o. za potrebe jednog privatnog vrtića u Varaždinskoj županiji.

Funkcionalnosti dodijeljene korisničkoj ulozi *Admin* su sljedeće:

- Sve funkcionalnosti ostalih korisničkih uloga
- Upravljanje korisnicima (dodavanje korisnika i uređivanje podataka korisnika, dodjeljivanje uloga korisnicima)
- Upravljanje djecom (dodavanje djece i uređivanje podataka djece)
- Upravljanje šifrarnicima (dodavanje i uređivanje *Skupina*, *Pedagoških godina*, *Vrsta dokumenata*, *Nacionalnosti*, *Gradova* i *Neradnih dana*)
- Uređivanje starih evidencija nazočnosti djece

Funkcionalnosti dodijeljene korisničkoj ulozi *Ravnatelj* su sljedeće:

- Pregled i ispis Matične knjige djece
- Upravljanje skupinama djece (dodjela skupina djece *Odgojiteljima*)
- Evidencija sanitarnog nadzora (dodavanje, pregled, uređivanje, ispis)

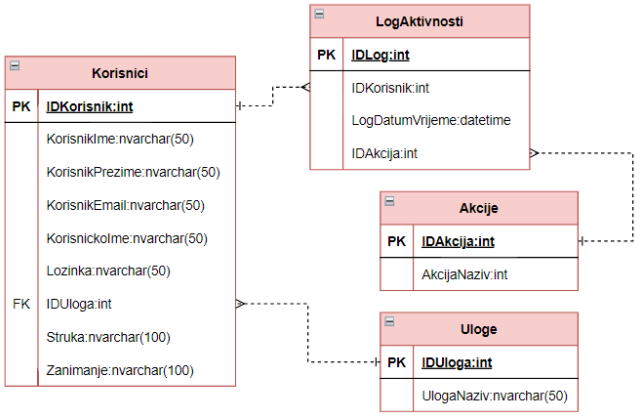
Funkcionalnosti dodijeljene korisničkoj ulozi *Odgojitelj* su sljedeće:

- Pregled i ispis Imenika djece
- Evidencija nazočnosti djece (dodavanje dnevne evidencije nazočnosti djece po skupini, pregledavanje starih evidencija nazočnosti djece te ispis mjesečnog pregleda evidencija nazočnosti djece po skupini)
- Evidencija sukoba djece i sukoba roditelja (dodavanje, uređivanje i ispis)
- Evidencija ozljeda djece (dodavanje, uređivanje i ispis)
- Zdravstveni kartoni djece (ispis)
- Evidencija antropometrijskih mjerenja djece (dodavanje, uređivanje i ispis)
- Evidencija alergija djece (dodavanje, uređivanje i ispis)
- Evidencija individualnih sastanaka s roditeljima (dodavanje, uređivanje i ispis)
- Program stručnog usavršavanja odgojitelja (dodavanje, uređivanje i ispis)
- Brzi pregled djece

1.7. Izrada ERA modela

1.7.1. ERA model *Korisnici aplikacije*

ERA model dijela baze podataka vezan uz korisnike aplikacije, njihove uloge i akcije izvršene u aplikaciji prikazan je na slici 10.



Slika 10. ERA model *Korisnici aplikacije*
(Izvor: vlastita izrada)

ERA modela *Korisnici aplikacije* sastoji se od 4 tablice: *Uloge*, *Korisnici*, *Akcije* i *LogAktivnosti*.

Tablica *Uloge* sadrži atribute: *IDUloga* (primarni ključ: **PK***;* tip podatka: *int*) i *UlogaNaziv* (tip podatka: *nvarchar(50)*). Ona predstavlja šifrarnik mogućih uloga aplikacije (npr. *Admin*, *Odgojitelj*, *Ravnatelj* i sl.).

Tablica *Korisnici* sadrži atribute: *IDKorisnici* (primarni ključ: **PK***;* tip podatka: *int*), *KorisnikIme* (tip podatka: *nvarchar(50)*), *KorisnikPrezime* (tip podatka: *nvarchar(50)*), *KorisnikEmail* (tip podatka: *nvarchar(50)*), *KorisnickoIme* (tip podatka: *nvarchar(50)*), *Lozinka* (tip podatka: *nvarchar(50)*), *IDUloga* (vanjski ključ na tablicu *Uloge*: FK; tip podatka: *int*), *Struka* (tip podatka: *nvarchar(100)*) i *Zanimanje* (tip podatka: *nvarchar(100)*). Ona služi za spremanje podataka korisnika aplikacije (npr. ime, prezime, lozinka i sl.) te daje bitnu informaciju o ulozi pojedinog korisnika na temelju koje korisnik ima pristup određenim funkcionalnostima *web* aplikacije.

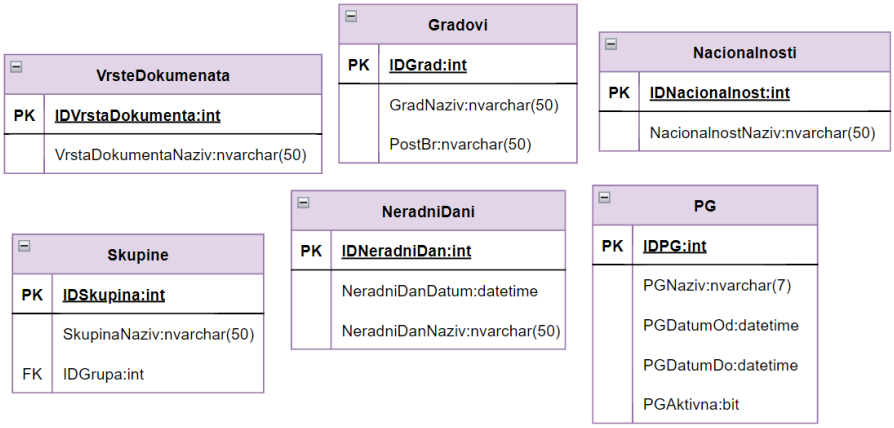
Tablica *Akcije* sadrži atribute: *IDAkcija* (primarni ključ: **PK***;* tip podatka: *int*) i *AkcijaNaziv* (tip podatka: *nvarchar(50)*). Ona predstavlja šifrarnik mogućih akcija koje korisnik može izvršiti u aplikaciji (npr. *Prijava*).

Tablica *LogAktivnosti* sadrži atribute: *IDLog* (primarni ključ: **PK***;* tip podatka: *int*), *IDKorisnik* (vanjski ključ na tablicu *Korisnici*: FK; tip podatka: *int*), *LogDatumVrijeme* (tip podatka: *datetime*) i *IDAkcija* (vanjski ključ na tablicu *Akcije*: FK; tip podatka: *int*). Ona je treća

tablica u vezi više-više između tablica *Korisnici* i *Akcije*. Služi za spremanje podataka o izvršenoj akciji korisnika aplikacije u neko određeno vrijeme (npr. korisnik *Ana Anić*; akcija *Prijava*).

1.7.2. ERA model *Šifrarnici*

ERA model dijela baze podataka vezan uz šifrarnike prikazan je na slici 11.



Slika 11. ERA model *Šifrarnici*

(Izvor: vlastita izrada)

ERA modela *Šifrarnici* sastoji se od 6 tablica: *VrsteDokumenata*, *Gradovi*, *Nacionalnosti, Skupine, NeradniDani* i *PG*.

Tablica *VrsteDokumenata* sadrži atribute: *IDVrstaDokumenta* (primarni ključ: **PK***;* tip podatka: *int*) i *VrstaDokumentaNaziv* (tip podataka: *nvarchar(50)*). Ona predstavlja šifrarnik vrsti dokumenata s njihovim nazivima (npr. *Ispričnica*, *Ugovor* i sl.).

Tablica *Gradovi* sadrži atribute: *IDGrad* (primarni ključ: **PK***;* tip podatka: *int*), *GradNaziv* (tip podatka: *nvarchar(50)*) i *PostBr* (tip podatka: *nvarchar(50)*). Ona predstavlja šifrarnik mogućih gradova/mjesta stanovanja djeteta, roditelja i dr. (npr. *Varaždin*, *Sračinec* i sl.).

Tablica *Nacionalnosti* sadrži atribute: *IDNacionalnost* (primarni ključ: **PK***;* tip podatka: *int*) i *NacionalnostNaziv* (tip podatka: *nvarchar(50)*). Ona predstavlja šifrarnik mogućih nacionalnosti djeteta, roditelja i dr. (npr. *Hrvatsko*, *Romsko* i sl.).

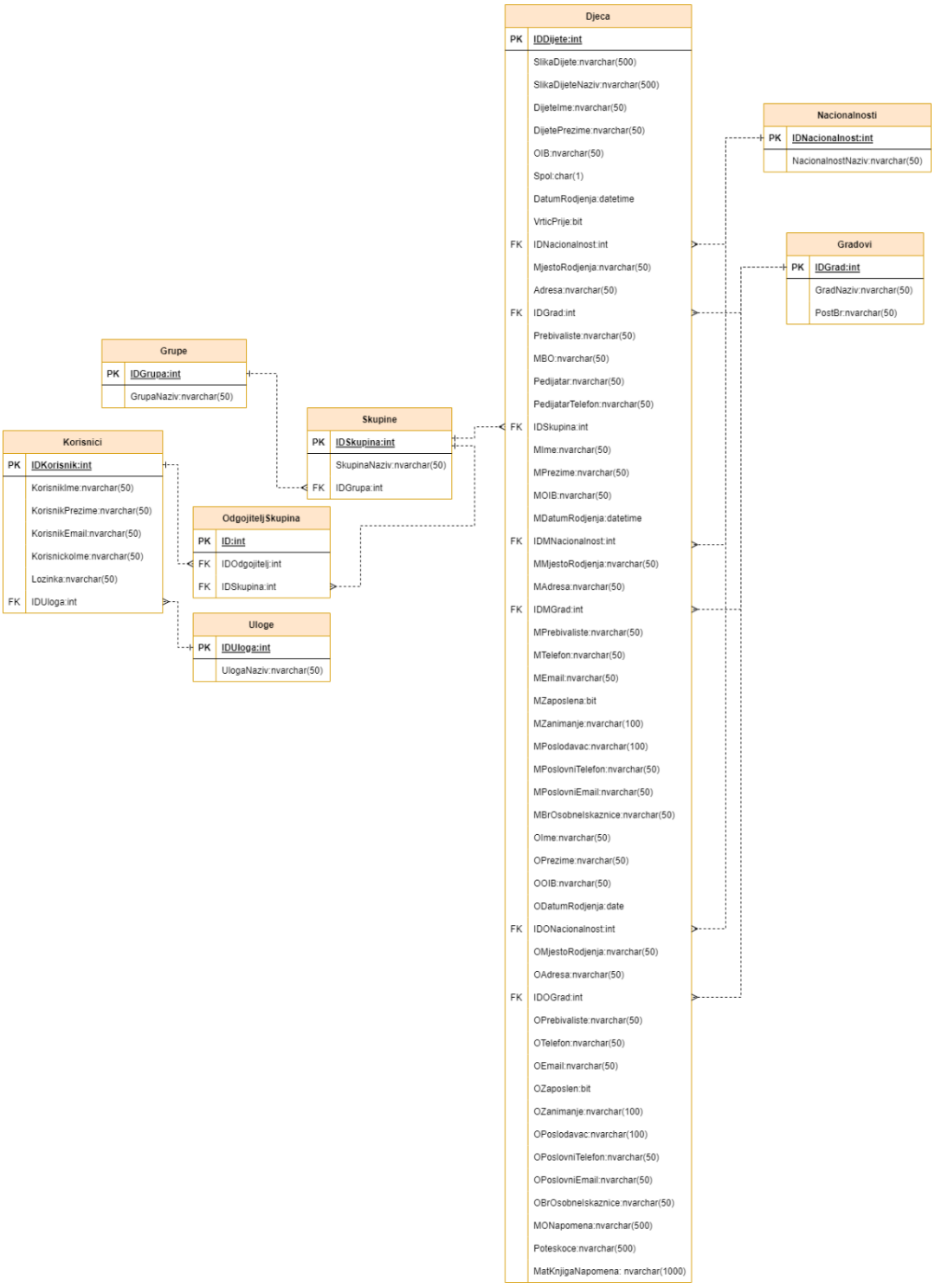
Tablica *Skupine* sadrži atribute: *IDSkupina* (primarni ključ: **PK***;* tip podatka: *int*), *SkupinaNaziv* (tip podatka: *nvarchar(50)*) i *IDGrupa* (vanjski ključ na tablicu *Grupe*: FK, tip podatka: *int*). Ona predstavlja šifrarnik mogućih odgojno-obrazovnih skupina djece u vrtiću (npr. *Bubamare*, *Ribice* i sl.).

Tablica *NeradniDani* sadrži atribute: *IDNeradniDan* (primarni ključ: **PK***;* tip podatka: *int*), *NeradniDanDatum* (tip podataka: *datetime*) i *NeradniDanNaziv* (tip podatka: *nvarchar(50)*). Ona predstavlja šifrarnik mogućih neradnih dana kroz godinu (npr. *01.05.2023. Praznik rada*).

Tablica *PG* sadrži atribute: *IDPG* (primarni ključ: **PK***;* tip podatka: *int*), *PGNaziv* (tip podataka: *nvarchar(7)*), *PGDatumOd* (tip podataka: *datetime*), *PGDatumDo* (tip podataka: *datetime*) i *PGAktivna* (tip podataka: *bit*). Ona predstavlja šifrarnik pedagoških godina s njihovim nazivima, datumima početka i završetka pedagoške godine te informacijom je li pojedina pedagoška godina trenutno aktivna (npr. pedagoška godina *2022/23*; počinje datuma *01.09.2022.;* završava datuma *31.08.2023.*; trenutno je aktivna *1-true*).

1.7.3. ERA model *Djeca*

ERA model dijela baze podataka vezan uz djecu i odgojno-obrazovne skupine djece s grupama kojima pripadaju te s njima dodijeljenim odgojiteljima prikazan je na slici 12.



Slika 12. ERA model *Djeca*

(Izvor: vlastita izrada)

ERA modela *Djeca* sastoji se od 7 tablica: *Nacionalnosti*, *Gradovi*, *Grupe*, *Skupine, Djeca, Korisnici* i *OdgojiteljSkupina*.

Tablica *Nacionalnosti* opisana je u poglavlju 5.2.2. ERA model *Šifrarnici*.

Tablica *Gradovi* opisana je u poglavlju 5.2.2. ERA model *Šifrarnici*.

Tablica *Skupine* opisana je u poglavlju 5.2.2. ERA model *Šifrarnici*.

Tablica *Korisnici* opisana je u poglavlju 5.2.1. ERA model *Korisnici aplikacije*.

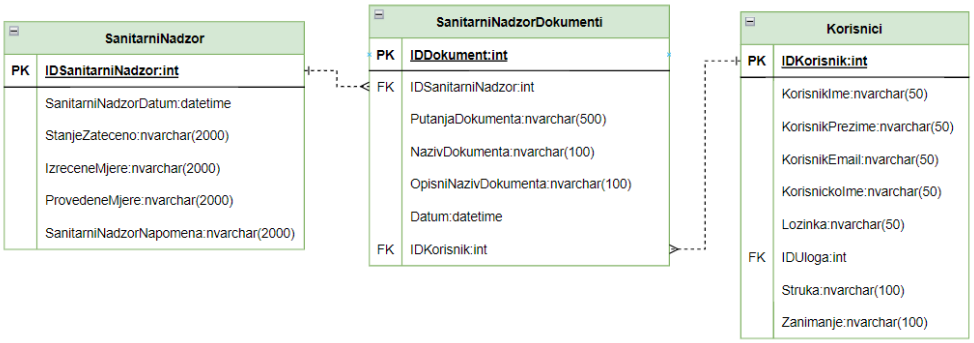
Tablica *OdgojiteljSkupina* sadrži atribute: *ID* (primarni ključ: **PK***;* tip podatka: *int*), *IDOdgojitelj* (vanjski ključ na tablicu *Korisnici*: FK; tip podatka: *int*) i *IDSkupina* (vanjski ključ na tablicu *Skupine*: FK; tip podatka: *int*). Ona je treća tablica u vezi više-više između tablica *Korisnici* i *Skupine*. Služi za spajanje određenog odgojitelja s njegovim skupinama djece koje vodi (npr. korisnik *Ana Anić* je odgojitelj skupine *Bubamare*).

Tablica *Djeca* sadrži atribute: *IDDijete* (primarni ključ: **PK***;* tip podatka: *int*), *SlikaDijete* (tip podatka: *nvarchar(500)*), *SlikaDijeteNaziv* (tip podatka: *nvarchar(500)*), *DijeteIme* (tip podatka: *nvarchar(50)*), *DijetePrezime* (tip podatka: *nvarchar(50)*), *OIB* (tip podatka: *nvarchar(50)*), *Spol* (tip podatka: *char(1)*), *DatumRodjenja* (tip podatka: *datetime*), *VrticPrije* (tip podatka: *bit*), *IDNacionalnost* (vanjski ključ na tablicu *Nacionalnosti*: FK; tip podatka: *int*), *MjestoRodjenja* (tip podatka: *nvarchar(50)*), *Adresa* (tip podatka: *nvarchar(50)*), *IDGrad* (vanjski ključ na tablicu *Gradovi*: FK; tip podatka: *int*), *Prebivaliste* (tip podatka: *nvarchar(50)*), *MBO* (tip podatka: *nvarchar(50)*), *Pedijatar* (tip podatka: *nvarchar(50)*), *PedijatarTelefon* (tip podatka: *nvarchar(50)*), *IDSkupina* (vanjski ključ na tablicu *Skupine*: FK; tip podatka: *int*), *Napomena* (tip podatka: *nvarchar(500)*), *Mime* (tip podatka: *nvarchar(50)*), *MPrezime* (tip podatka: *nvarchar(50)*), *MOIB* (tip podatka: *nvarchar(50)*), *MDatumRodjenja* (tip podatka: *date*), *IDMNacionalnost* (vanjski ključ na tablicu *Nacionalnosti*: FK; tip podatka: *int*), *MMjestoRodjenja* (tip podatka: *nvarchar(50)*), *MAdresa* (tip podatka: *nvarchar(50)*), *IDMGrad* (vanjski ključ na tablicu *Gradovi*: FK; tip podatka: *int*), *MPrebivaliste* (tip podatka: *nvarchar(50)*), *MTelefon* (tip podatka: *nvarchar(50)*), *MEmail* (tip podatka: *nvarchar(50)*), *MZaposlena* (tip podatka: *bit*), *MZanimanje* (tip podatka: *nvarchar(100)*), *MPoslodavac* (tip podatka: *nvarchar(100)*), *MPoslovniTelefon* (tip podatka: *nvarchar(50)*), *MPoslovniEmail* (tip podatka: *nvarchar(50)*), *MBrOsobneIskaznice* (tip podatka: *nvarchar(50)*), *Oime* (tip podatka: *nvarchar(50)*), *OPrezime* (tip podatka: *nvarchar(50)*), *OOIB* (tip podatka: *nvarchar(50)*), *ODatumRodjenja* (tip podatka: *date*), *IDONacionalnost* (vanjski ključ na tablicu *Nacionalnosti*: FK; tip podatka: *int*), *OMjestoRodjenja* (tip podatka: *nvarchar(50)*), *OAdresa* (tip podatka: *nvarchar(50)*), *IDOGrad* (vanjski ključ na tablicu *Gradovi*: FK, tip podatka: *int*), *OPrebivaliste* (tip podatka: *nvarchar(50)*), *OTelefon* (tip podatka: *nvarchar(50)*), *OEmail* (tip podatka: *nvarchar(50)*), *OZaposlen* (tip podatka: *bit*), *OZanimanje* (tip podatka: *nvarchar(100)*), *OPoslodavac* (tip podatka: *nvarchar(100)*), *OPoslovniTelefon* (tip podatka: *nvarchar(50)*), *OPoslovniEmail* (tip podatka*: nvarchar(50)*), *OBrOsobneIskaznice* (tip podatka: *nvarchar(50)*),

MONapomena (tip podatka: *nvarchar(500)*), *Poteskoce* (tip podatka: *nvarchar(500)*), *MatKnjigaNapomena* (tip podatka: *nvarchar(1000)*). Ona predstavlja glavni šifrarnik djece koja su pohađala ili pohađaju vrtić sa svim njihovim osobnim podacima, podacima roditelja, zdravstvenim podacima i sl.

1.7.4. ERA model *Sanitarni nadzor*

ERA model dijela baze podataka vezan uz sanitarni nadzor prikazan je na slici 13.



Slika 13. ERA model *Sanitarni nadzor*

(Izvor: vlastita izrada)

ERA model *Sanitarni nadzor* sastoji se od 3 tablice: *Korisnici, SanitarniNadzor* i *SanitarniNadzorDokumenti*.

Tablica *Korisnici* opisana je u poglavlju 5.2.1. ERA model *Korisnici aplikacije*.

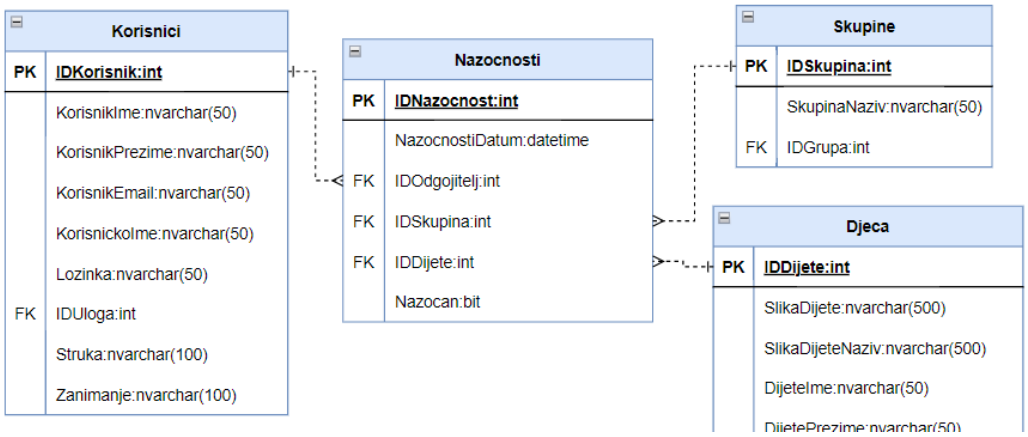
Tablica *SanitarniNadzor* sadrži atribute: *IDSanitarniNadzor* (primarni ključ: **PK***;* tip podatka: *int*), *SanitarniNadzorDatum* (tip podataka: *datetime*), *StanjeZateceno* (tip podataka: *nvarchar(2000)*), *IzreceneMjere* (tip podataka: *nvarchar(2000)*), *ProvedeneMjere* (tip podataka: *nvarchar(2000)*) i *SanitarniNadzorNapomena* (tip podataka: *nvarchar(2000)*). Ona služi spremanju podataka o provedenom sanitarnom nadzoru (npr. datuma *22.02.2023.*; zatečeno stanje *Usklađenost s propisima o sigurnosti hrane*; izrečene mjere *Izrada plana čišćenja i dezinfekcije opreme*; provedene mjere *Spremljeno u registar kuhinje*; napomena *Predati plan do 22.03.2023.*).

Tablica *SanitarniNadzorDokumenti* sadrži atribute: *IDDokument* (primarni ključ: **PK***;* tip podatka: *int*), *IDSanitarniNadzor* (vanjski ključ na tablicu *SanitarniNadzor*: FK; tip podatka: *int*), *PutanjaDokumenta* (tip podatka: *nvarchar(500)*), *NazivDokumenta* (tip podatka: *nvarchar(100)*), *OpisniNazivDokumenta* (tip podatka: *nvarchar(100)*), *Datum* (tip podatka: *datetime*) i *IDKorisnik* (vanjski ključ na tablicu *Korisnici*: FK; tip podatka: *int*). Ona služi za spremanje dokumenata povezanih uz određeni sanitarni nadzor sa svim informacijama vezanim uz pojedini dokument poput putanje do samog dokumenta, njegovog naziva i opisnog

naziva, datuma objave i korisnika koji je objavio dokument (npr. dokument *planCiscenja.pdf*; opisnoga naziva *PlanCiscenja2023*; nalazi se na putanji */dokumenti/sanitarniNadzor/2023-02-23/ planCiscenja.pdf*; pripada sanitarnom nadzoru *23.02.2023.*; objavila ga je ravnateljica *Nina Ninić*; datuma *23.02.2023.*).

1.7.5. ERA model *Nazočnost djece*

ERA model dijela baze podataka vezan uz evidenciju nazočnosti djece prikazan je na slici 14.



Slika 14. ERA model *Nazočnost djece*

(Izvor: vlastita izrada)

ERA model *Nazočnost djece* sastoji se od 4 tablice: *Korisnici*, *Skupine*, *Djeca* i *Nazocnosti*.

Tablica *Korisnici* opisana je u poglavlju 5.2.1. ERA model *Korisnici aplikacije*.

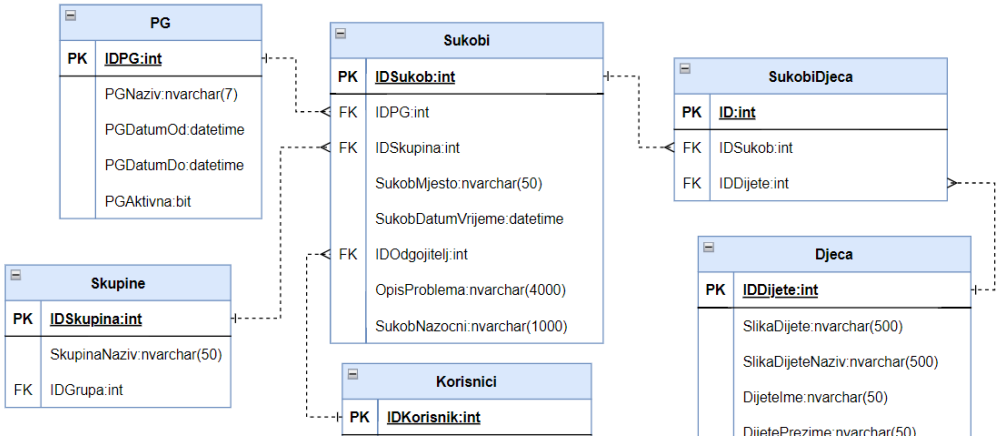
Tablica *Skupine* opisana je u poglavlju 5.2.3. ERA model *Djeca*.

Tablica *Djeca* opisana je u poglavlju 5.2.3. ERA model *Djeca*.

Tablica *Nazocnosti* sadrži atribute: *IDNazocnost* (primarni ključ: **PK**; tip podatka: *int*), *NazocnostiDatum* (tip podatka: *datetime*), *IDOdgojitelj* (vanjski ključ na tablicu *Korisnici*: FK; tip podatka: *int*), *IDSkupina* (vanjski ključ na tablicu *Skupine*: FK; tip podatka: *int*), *IDDijete* (vanjski ključ na tablicu *Djeca*: FK; tip podatka: *int*) i *Nazocan* (tip podatka: *bit*). Ona služi za spremanje podataka o nazočnosti djeteta u danu (npr. dijete *Pero Perić;* datuma *20.02.2023.*; nazočan *1-true*).

1.7.6. ERA model *Sukobi među djecom*

ERA model dijela baze podataka vezan uz evidenciju sukoba među djecom prikazan je na slici 15.



Slika 15. ERA model *Sukobi među djecom*

(Izvor: vlastita izrada)

ERA model *Sukobi među djecom* sastoji se od 6 tablice: *Skupine, PG, Korisnici*, *Sukobi*, *Djeca* i *SukobiDjeca*.

Tablica *Skupine* opisana je u poglavlju 5.2.3. ERA model *Djeca*.

Tablica *PG* opisana je u poglavlju 5.2.2. ERA model *Šifrarnici*.

Tablica *Korisnici* opisana je u poglavlju 5.2.1. ERA model *Korisnici aplikacije*.

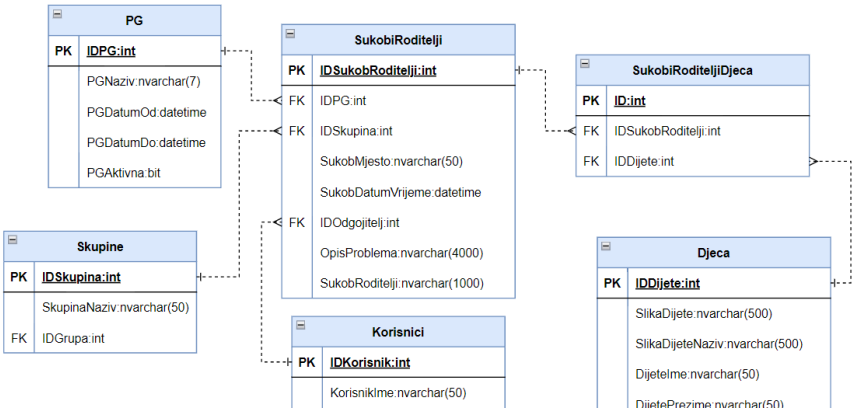
Tablica *Sukobi* sadrži atribute: *IDSukob* (primarni ključ: **PK***;* tip podatka: *int*), *IDPG* (vanjski ključ na tablicu *PG*: FK; tip podatka: *int*), *IDSkupina* (vanjski ključ na tablicu *Skupine*: FK; tip podatka: *int*), *SukobMjesto* (tip podatka: *nvarchar(50)*), *SukobDatumVrijeme* (tip podatka: *datetime*), *IDOdgojitelj* (vanjski ključ na tablicu *Korisnici*: FK; tip podatka: *int*), *OpisProblema* (tip podatka: *nvarchar(4000)*) i *SukobNazocni* (tip podataka: *nvarchar(1000)*). Ona služi za spremanje opisnih podataka o sukobima, tj. kad i gdje se dogodio sukob, u kojoj skupini, koji odgojitelj je zabilježio sukob, tko je sve bio nazočan te opis samog sukoba (npr. pedagoške godine *2022/23*; u skupini *Bubamare*; datuma i vremena *22.03.2023. 10:00*; na mjestu *hodnik vrtića*; nazočne su bile *Lara Larić i Sara Sarić*; s opisom problema da se *Dogodila svađa među djecom oko igračke*; zabilježio odgojitelj *Ana Anić*).

Tablica *Djeca* opisana je u poglavlju 5.2.3. ERA model *Djeca*.

Tablica *SukobiDjeca* sadrži atribute: *ID* (primarni ključ: **PK***;* tip podatka: *int*), *IDSukob* (vanjski ključ na tablicu *Sukobi*: FK; tip podatka: *int*) i *IDDijete* (vanjski ključ na tablicu *Djeca*: FK; tip podatka: *int*). Ona je treća tablica u vezi više-više između tablica *Sukobi* i *Djeca*. Služi za spajanje određenog djeteta sa sukobom u kojem je sudjelovalo pomoću vanjskih ključeva.

1.7.7. ERA model *Sukobi među roditeljima*

ERA model dijela baze podataka vezan uz evidenciju sukoba među roditeljima prikazan je na slici 16.



Slika 16. ERA model Sukobi među roditeljima
(Izvor: vlastita izrada)

ERA model *Sukobi među roditeljima* sastoji se od 6 tablica: *Skupine, PG, Korisnici*, *SukobiRoditelji*, *Djeca* i *SukobiRoditeljiDjeca*.

Tablica *Skupine* opisana je u poglavlju 5.2.3. ERA model *Djeca*.

Tablica *PG* opisana je u poglavlju 5.2.6. ERA model *Sukobi među djecom*.

Tablica *Korisnici* opisana je u poglavlju 5.2.1. ERA model *Korisnici aplikacije*.

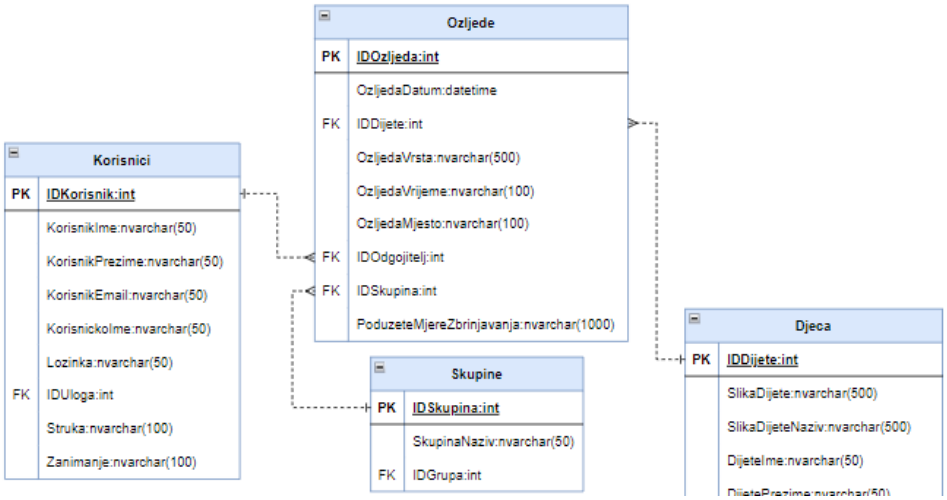
Tablica *SukobiRoditelji* sadrži atribute: *IDSukobRoditelji* (primarni ključ: **PK***;* tip podatka: *int*), *IDPG* (vanjski ključ na tablicu *PG*: FK; tip podatka: *int*), *IDSkupina* (vanjski ključ na tablicu *Skupine*: FK; tip podatka: *int*), *SukobMjesto* (tip podatka: *nvarchar(50)*), *SukobDatumVrijeme* (tip podatka: *datetime*), *IDOdgojitelj* (vanjski ključ na tablicu *Korisnici*: FK; tip podatka: *int*), *OpisProblema* (tip podatka: *nvarchar(4000)*) i *SukobRoditelji* (tip podatka: *nvarchar(1000)*). Ona služi za spremanje opisnih podataka o sukobima, tj. kad i gdje se dogodio sukob, u kojoj skupini, koji odgojitelj je zabilježio sukob, opis samog sukoba i popis njegovih sudionika (npr. pedagoške godine *2022/23*; u skupini *Bubamare*; datuma i vremena *22.03.2023. 10:00*; na mjestu *Hodnik vrtića*; s opisom problema da se *Dogodila svađa među roditeljima*; zabilježio odgojitelj *Ana Anić*; sudionici su bili *Majka Petra Perića i majka Tina Tinića*).

Tablica *Djeca* opisana je u poglavlju 5.2.3. ERA model *Djeca*.

Tablica *SukobiRoditeljiDjeca* sadrži atribute: *ID* (primarni ključ: **PK***;* tip podatka: *int*), *IDSukobRoditelji* (vanjski ključ na tablicu *SukobiRoditelji*: FK; tip podatka: *int*) i *IDDijete* (vanjski ključ na tablicu *Djeca*: FK; tip podatka: *int*). Ona je treća tablica u vezi više-više između tablica *SukobiRoditelji* i *Djeca*. Služi za spajanje određenog djeteta sa sukobom u kojem je njegov roditelj sudjelovao pomoću vanjskih ključeva.

1.7.8. ERA model *Ozljede djece*

ERA model dijela baze podataka vezan uz evidenciju ozljeda djece prikazan je na slici 17.



Slika 17. ERA model *Ozljede djece*

(Izvor: vlastita izrada)

ERA model *Ozljede djece* sastoji se od 4 tablice: *Skupine, Korisnici*, *Djeca* i *Ozljede*.

Tablica *Skupine* opisana je u poglavlju 5.2.3. ERA model *Djeca*.

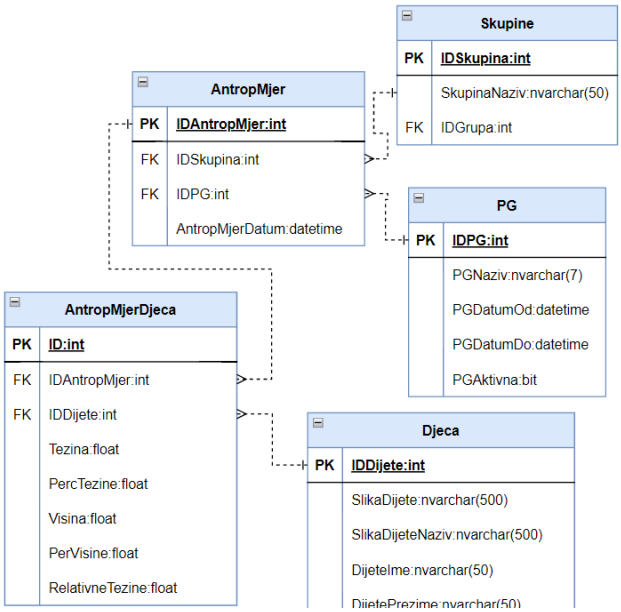
Tablica *Korisnici* opisana je u poglavlju 5.2.1. ERA model *Korisnici aplikacije*.

Tablica *Djeca* opisana je u poglavlju 5.2.3. ERA model *Djeca*.

Tablica *Ozljede* sadrži atribute: *IDOzljeda* (primarni ključ: **PK***;* tip podatka: *int*), *OzljedaDatum* (tip podatka: *datetime*), *IDDijete* (vanjski ključ na tablicu *Djeca*: FK; tip podatka: *int*), *OzljedaVrsta* (tip podataka: *nvarchar(100)*), *OzljedaVrijeme* (tip podataka: *nvarchar(100)*), *OzljedaMjesto* (tip podataka: *nvarchar(100)*), *IDOdgojitelj* (vanjski ključ na tablicu *Korisnici*: FK; tip podatka: *int*), *IDSkupina* (vanjski ključ na tablicu *Skupine*: FK; tip podatka: *int*) i *PoduzeteMjereZbrinjavanja* (tip podatka: *nvarchar(1000)*). Ona služi za spremanje podataka o ozljedi djeteta (npr. dijete *Pero Perić;* datuma *20.02.2023.*; vrsta ozljede *Pad na glavu*; vrijeme ozljede *Oko 13h*; mjesto ozljede *Dvorište vrtića*; poduzete mjere zbrinjavanja *Stavljen oblog*).

1.7.9. ERA model *Antropometrijska mjerenja djece*

ERA model dijela baze podataka vezan uz evidenciju sukoba među djecom prikazan je na slici 18.



Slika 18. ERA model Antropometrijska mjerenja djece
(Izvor: vlastita izrada)

ERA model *Antropometrijska mjerenja djece* sastoji se od 5 tablica: *Skupine, PG, AntropMjer, Djeca* i *AntropMjerDjeca*.

Tablica *Skupine* opisana je u poglavlju 5.2.3. ERA model *Djeca*.

Tablica *PG* opisana je u poglavlju 5.2.6. ERA model *Sukobi među djecom*.

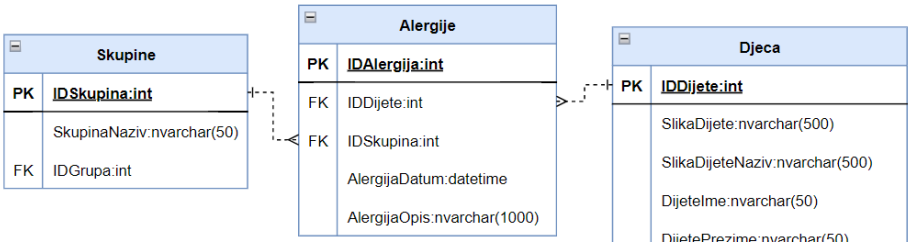
Tablica *AntropMjer* sadrži atribute: *IDAntropMjer* (primarni ključ: **PK***;* tip podatka: *int*), *IDSkupina* (vanjski ključ na tablicu *Skupine*: FK; tip podatka: *int*), *IDPG* (vanjski ključ na tablicu *PG*: FK; tip podatka: *int*) i *AntropMjerDatum* (tip podataka: *datetime*). Ona služi za spremanje opisnih podataka o antropometrijskim mjerenjima, tj. u kojoj je skupini koje pedagoške godine i kojeg datuma obavljeno antropometrijsko mjerenje (npr. pedagoške godine *2022/23*; u skupini *Bubamare*; datuma *22.03.2023.*).

Tablica *Djeca* opisana je u poglavlju 5.2.3. ERA model *Djeca*.

Tablica *AntropMjerDjeca* sadrži atribute: *ID* (primarni ključ: **PK***;* tip podatka: *int*), *IDAntropMjer* (vanjski ključ na tablicu *AntropMjer*: FK; tip podatka: *int*), *IDDijete* (vanjski ključ na tablicu *Djeca*: FK; tip podatka: *int*), *Tezina* (tip podatka: *float*), *PercTezine* (tip podatka: *float*), *Visina* (tip podatka: *float*), *PerVisine* (tip podatka: *float*), *RelativneTezine* (tip podatka: *float*). Ona je treća tablica u vezi više-više između tablica *AntropMjer* i *Djeca*. Služi za spajanje određenog djeteta s provedenim antropometrijskim mjerenjem i za spremanje podataka o izmjerenim mjerama za to dijete (njegova težina, visina i sl.).

1.7.10. ERA model *Alergija djece*

ERA model dijela baze podataka vezan uz evidenciju sukoba među djecom prikazan je na slici 19.



Slika 19. ERA model *Alergija djece*

(Izvor: vlastita izrada)

ERA model *Alergija djece* sastoji se od 3 tablice: *Skupine, Djeca* i *Alergije*.

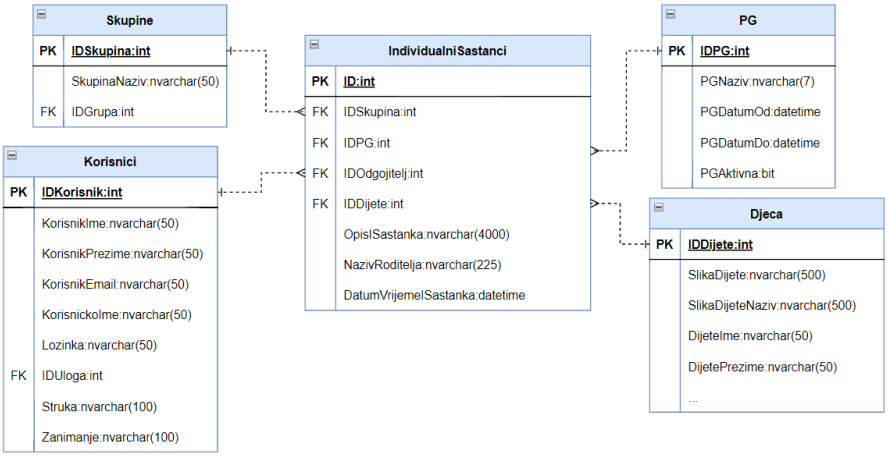
Tablica *Skupine* opisana je u poglavlju 5.2.3. ERA model *Djeca*.

Tablica *Djeca* opisana je u poglavlju 5.2.3. ERA model *Djeca*.

Tablica *Alergije* sadrži atribute: *IDAlergija* (primarni ključ: **PK***;* tip podatka: *int*), *IDDijete* (vanjski ključ na tablicu *Djeca*: FK; tip podatka: *int*), *IDSkupina* (vanjski ključ na tablicu *Skupine*: FK, tip podatka: *int*), *AlergijaDatum* (tip podatka: *datetime*), *AlergijaOpis* (tip podatka: *nvarchar(1000)*). Ona služi za spremanje podataka o alergijama djece (npr. dijete *Pero Perić*; iz skupine *Bubamare*; od datuma *20.02.2023.*; ima dijagnosticiranu *Alergiju na pelud*).

1.7.11. ERA model *Individualni sastanci*

ERA model dijela baze podataka vezan uz evidenciju individualnih sastanaka s roditeljima djece prikazan je na slici 20.



Slika 20. ERA model Individualni sastanci

(Izvor: vlastita izrada)

ERA model *Individualni sastanci* sastoji se od 5 tablica: *Skupine, PG, Korisnici, Djeca* i *IndividualniSastanci*.

Tablica *Skupine* opisana je u poglavlju 5.2.3. ERA model *Djeca*.

Tablica *PG* opisana je u poglavlju 5.2.6. ERA model *Sukobi među djecom*.

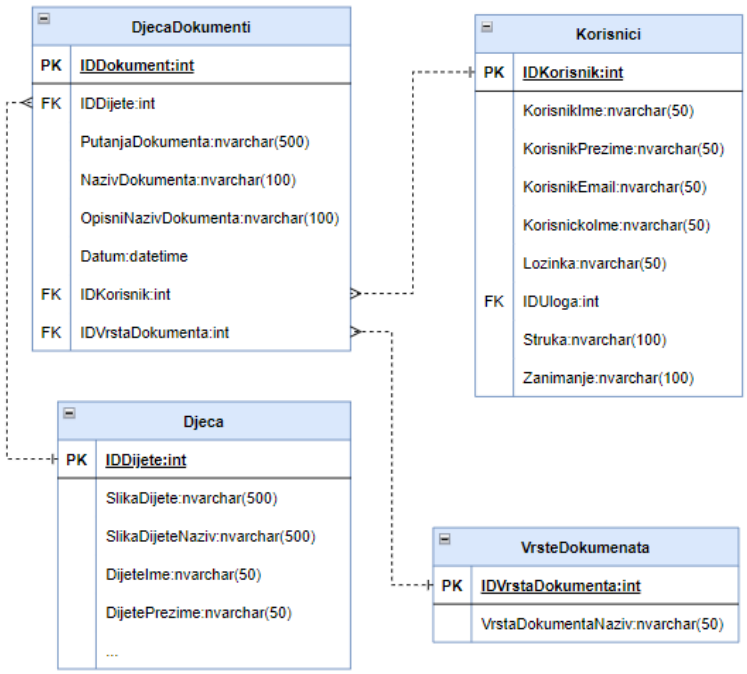
Tablica *Korisnici* opisana je u poglavlju 5.2.1. ERA model *Korisnici aplikacije*.

Tablica *Djeca* opisana je u poglavlju 5.2.3. ERA model *Djeca*.

Tablica *IndividualniSastanci* sadrži atribute: *ID* (primarni ključ: **PK***;* tip podatka: *int*), *IDSkupina* (vanjski ključ na tablicu *Skupine*: FK; tip podatka: *int*), *IDPG* (vanjski ključ na tablicu *PG*: FK; tip podatka: *int*), *IDOdgojitelj* (vanjski ključ na tablicu *Korisnici*: FK; tip podatka: *int*), *IDDijete* (vanjski ključ na tablicu *Djeca*: FK; tip podatka: *int*), *OpisISastanka* (tip podatka: *nvarchar(4000)*), *NazivRoditelja* (tip podatka: *nvarchar(225)*) i *DatumVrijemeISastanka* (tip podataka: *datetime*). Ona služi za spremanje podataka o individualnim sastancima, tj. u kojoj skupini, koje pedagoške godine, kojeg datuma i u koliko sati je obavljen individualni sastanak s kojim roditeljem, za koje dijete te koji odgojitelj je obavio sastanak i zapisao njegov opis (npr. pedagoške godine *2022/23*; u skupini *Bubamare*; za dijete *Pero Perić*; datuma i vremena *22.03.2023. 16:00*; s roditeljem *Maja Majić*; odgojitelj *Ana Anić*; obavljen *Individualni sastanak o djetetovom napretku kroz mjesec*).

1.7.12. ERA model *Dokumenti djece*

ERA model dijela baze podataka vezan uz dokumente djece prikazan je na slici 21.



Slika 21. ERA model *Dokumenti djece*

(Izvor: vlastita izrada)

ERA model *Dokumenti djece* sastoji se od 4 tablice: *Korisnici, VrsteDokumenata, Djeca* i *DjecaDokumenti*.

Tablica *Korisnici* opisana je u poglavlju 5.2.1. ERA model *Korisnici aplikacije*.

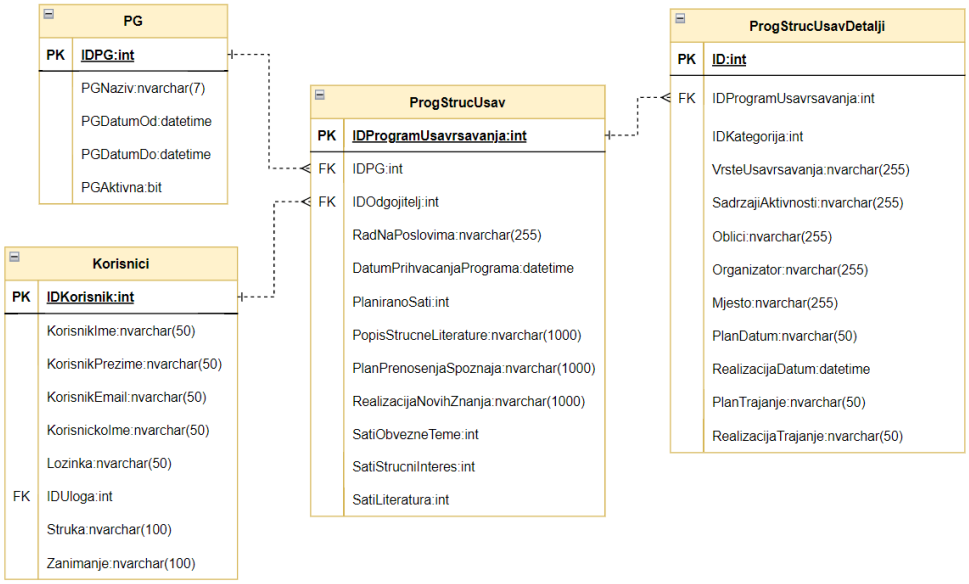
Tablica *VrsteDokumenata* opisana je u poglavlju 5.2.2. ERA model *Šifrarnici*.

Tablica *Djeca* opisana je u poglavlju 5.2.3. ERA model *Djeca*.

Tablica *DjecaDokumenti* sadrži atribute: *IDDokument* (primarni ključ: **PK***;* tip podatka: *int*), *IDDijete* (vanjski ključ na tablicu *Djeca*: FK; tip podatka: *int*), *PutanjaDokumenta* (tip podatka: *nvarchar(500)*), *NazivDokumenta* (tip podatka: *nvarchar(100)*), *OpisniNazivDokumenta* (tip podatka: *nvarchar(100)*), *Datum* (tip podatka: *datetime*), *IDKorisnik* (vanjski ključ na tablicu *Korisnici*: FK; tip podatka: *int*) i *IDVrstaDokumenta* (vanjski ključ na tablicu *VrsteDokumenata*: FK; tip podatka: *int*). Ona služi za spremanje dokumenata povezanih uz određeno dijete sa svim informacijama vezanim uz određeni dokument poput putanje do samog dokumenta, njegovog naziva i opisnog naziva, datuma objave i korisnika koji je objavio dokument te kojoj on vrsti dokumenta pripada (npr. dokument *ugovor.pdf*; opisnoga naziva *Ugovor Pero Perić 2022*; nalazi se na putanji */dokumenti/djeca/11662143178/ugovor.pdf*; pripada djetetu *Pero Perić*; objavio ga je odgojitelj *Iva Ivić*; datuma *20.02.2023.*).

1.7.13. ERA model *Program stručnog usavršavanja odgojitelja*

ERA model dijela baze podataka vezan uz program stručnog usavršavanja odgojitelja prikazan je na slici 22.



Slika 22. ERA model Program stručnog usavršavanja odgojitelja

(Izvor: vlastita izrada)

ERA model *Program stručnog usavršavanja odgojitelja* sastoji se od 4 tablice: *PG, Korisnici, ProgStrucUsav* i *ProgStrucUsavDetalji*.

Tablica *PG* opisana je u poglavlju 5.2.2. ERA model *Šifrarnici*.

Tablica *Korisnici* opisana je u poglavlju 5.2.1. ERA model *Korisnici aplikacije*.

Tablica *ProgStrucUsav* sadrži atribute: *IDProgramUsavrsavanja* (primarni ključ: **PK***;* tip podatka: *int*), *IDPG* (vanjski ključ na tablicu *PG*: FK; tip podatka: *int*), *IDOdgojitelj* (vanjski ključ na tablicu *Korisnici*: FK, tip podatka: *int*), *RadNaPoslovima* (tip podataka: *nvarchar(255)*), *DatumPrihvacanjaPrograma* (tip podataka: *datetime*), *PlaniranoSati* (tip podataka: *int*), *PopisStrucneLiterature* (tip podataka: *nvarchar(1000)*), *PlanPrenosenjaSpoznaja* (tip podataka: *nvarchar(1000)*), *RealizacijaNovihZnanja* (tip podataka: *nvarchar(1000)*), *SatiObvezneTeme* (tip podataka: *int*), *SatiStrucniInteres* (tip podataka: *int*) i *SatiLiteratura* (tip podataka: *int*). Ona služi za spremanje opisnih podataka o knjizi programa stručnog usavršavanja odgojitelja pojedine pedagoške godine.

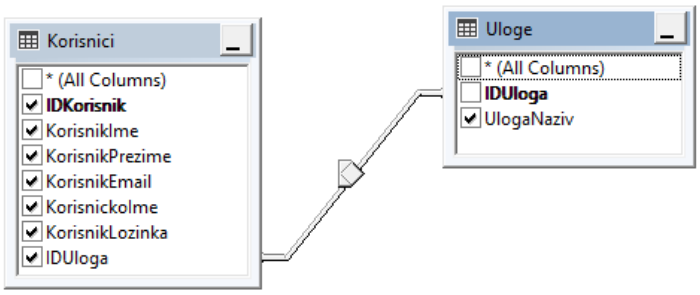
Tablica *ProgStrucUsavDetalji* sadrži atribute: *ID* (primarni ključ: **PK***;* tip podatka: *int*), *IDProgramUsavrsavanja* (vanjski ključ na tablicu *ProgStrucUsav*: FK; tip podatka: *int*), *IDKategorija* (tip podatka: *int*), *VrsteUsavrsavanja* (tip podatka: *nvarchar(255)*), *SadrzajAktivnosti* (tip podataka: *nvarchar(255)), Oblici* (tip podataka: *nvarchar(255)*), *Organizator* (tip podataka: *nvarchar(255)*), *Mjesto* (tip podataka: *nvarchar(255)*), *PlanDatum* (tip podataka: *nvarchar(50)*), *RealizacijaDatum* (tip podataka: *datetime), PlanTrajanje* (tip podataka: *nvarchar(50))* i *RealizacijaTrajanje* (tip podataka: *nvarchar(50))*. Ona služi za spremanje podataka o pojedinom programu stručnog usavršavanja odgojitelja unutar knjige programa stručnog usavršavanja odgojitelja pojedine pedagoške godine.

1.8. Izrada baze podataka

Baza podataka za potrebe *web* aplikacije ERP sustava za dječje vrtiće izrađena je u MS SQL Server Management Studiu prema prethodno izrađenim ERA dijagramima (objašnjeno u poglavlju 5.2. Izrada ERA modela). Osim kreiranja tablica, izrađeni su i *View*-evi za lakše dohvaćanje i prikaz podataka u samoj aplikaciji. Unutar *View*-a korištene su i neke funkcije poput *ISNULL*, *DATEDIFF* i sl. U nastavku će biti objašnjeni dijelovi baze podataka organizirani po izrađenim *View*-ima

1.8.1. Dio baze podataka *Korisnici aplikacije*

Dizajnerski prikaz *View*-a dijela baze podataka vezan uz korisnike aplikacije izrađen u MS SQL Server Management Studiu prikazan je na slici 23.



Slika 23. KorisniciView

(Izvor: vlastita izrada)

KorisniciView osim što uključuje sve atribute tablica *Korisnici* i *Uloge* te vezu između njih, uključuje i dodatne atribute pod nazivima *ImePrezimeKorisnik* i *PrezimeImeKorisnik*. Oni su izgrađeni spajanjem atributa *KorisnikIme* i *KorisnikPrezime* korištenjem *ISNULL* funkcije koja osigurava prikaz podatka iako koji od atributa ima *null* vrijednost. Cijela skripta *KorisniciView* vidljiva je u nastavku, a primjer ispisa skripte prikazan je na slici 24.

```
SELECT ISNULL(dbo.Korisnici.KorisnikIme,N'') + N' ' +
ISNULL(dbo.Korisnici.KorisnikPrezime,N'') AS ImePrezimeKorisnik,
ISNULL(dbo.Korisnici.KorisnikPrezime, N'') + N' ' +
ISNULL(dbo.Korisnici.KorisnikIme, N'') AS PrezimeImeKorisnik,
dbo.Korisnici.IDKorisnik, dbo.Korisnici.KorisnikIme,
dbo.Korisnici.KorisnikPrezime, dbo.Korisnici.KorisnikEmail,
dbo.Korisnici.KorisnickoIme, dbo.Korisnici.KorisnikLozinka,
dbo.Korisnici.IDUloga, dbo.Uloge.UlogaNaziv FROM dbo.Uloge RIGHT
OUTER JOIN dbo.Korisnici ON dbo.Uloge.IDUloga = dbo.Korisnici.IDUloga
```

Programski kod 5. SQL skripta *KorisniciView*

(Izvor: vlastita izrada)

ImePrezimeKo...	PrezimeImeKo...	IDKorisnik	KorisnikIme	KorisnikPrezime	KorisnikEmail	KorisnickoIme	KorisnikLozinka	IDUloga	UlogaNaziv
Iva Ivić	Ivić Iva	6	Iva	Ivić	iivic@vrtic.hr	iivic@vrtic.hr	1234	3	Odgojitelj
Admin Admin	Admin Admin	7	Admin	Admin	admin@vrtic.hr	admin@vrtic.hr	1234	1	Admin
Ana Anić	Anić Ana	13	Ana	Anić	aanic@vrtic.hr	aanic@vrtic.hr	1234	3	Odgojitelj
Zara Zarić	Zarić Zara	14	Zara	Zarić	zzaric@vrtic.hr	zzaric@email.hr	1234	3	Odgojitelj
Maja Majić	Majić Maja	15	Maja	Majić	mmajic@vrtic.hr	mmajic@vrtic.hr	1234	4	Kuhinja
Nina Ninić	Ninić Nina	16	Nina	Ninić	nninic@vrtic.hr	nninic@vrtic.hr	1234	2	Ravnatelj

Slika 24. Ispis SQL skripte *KorisniciView*

(Izvor: vlastita izrada)

Za potrebe izdvajanja samo onih korisnika koji nose ulogu *Odgojitelj* (u ovom slučaju pod *IDUloga* = *3*), napravljena je skripta u nastavku, a njezin ispis vidljiv je na slici 25.

```
SELECT IDKorisnik, KorisnikIme, KorisnikEmail, KorisnickoIme,
KorisnikLozinka, IDUloga, ISNULL(KorisnikIme, N'') + N' ' +
ISNULL(KorisnikPrezime, N'') AS OdgojiteljImePrezime FROM
dbo.Korisnici WHERE (IDUloga = 3)
```

Programski kod 6. SQL skripta *Odgojitelji*

(Izvor: vlastita izrada)

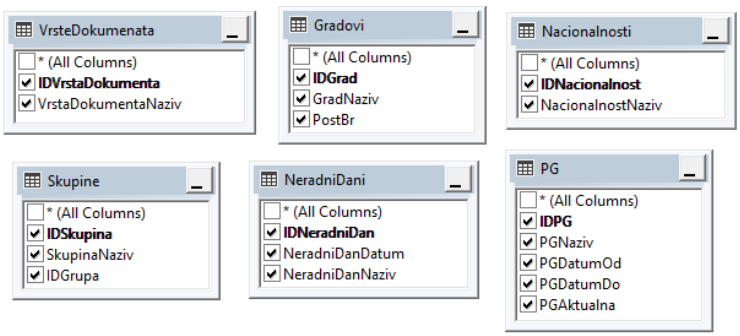
IDKorisnik	KorisnikIme	KorisnikEmail	Korisnickolme	KorisnikLozinka	IDUloga	OdgojiteljIme...
6	Iva	iivic@vrtic.hr	iivic@vrtic.hr	1234	3	Iva Ivić
13	Ana	aanic@vrtic.hr	aanic@vrtic.hr	1234	3	Ana Anić
14	Zara	zzaric@vrtic.hr	zzaric@email.hr	1234	3	Zara Zarić
18	Tara	ttaric@vrtic.hr	ttaric	12345	3	Tara Tarić

Slika 25. Ispis SQL skripte *Odgojitelji*

(Izvor: vlastita izrada)

1.8.2. Dio baze podataka *Šifrarnici*

Dio baze podataka vezan uz šifrarnike nema *View*, već sadrži odvojene tablice koje se direktno pozivaju iz aplikacije. Tablice *VrsteDokumenata*, *Gradovi*, *Nacionalnosti*, *Skupine*, *NeradniDani* i *PG* izrađene u MS SQL Server Management Studiu prikazane su na slici 26.

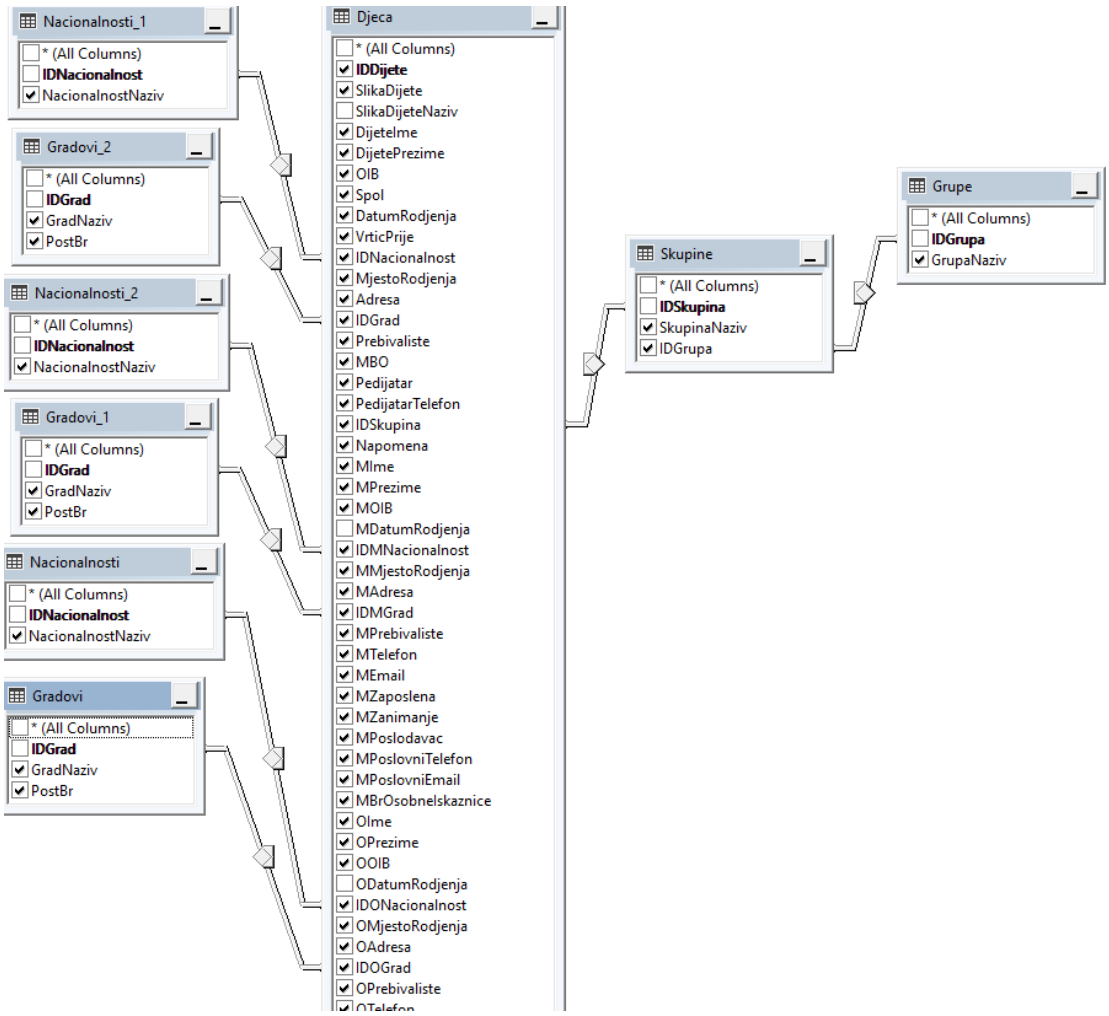


Slika 26. SifrarniciView

(Izvor: vlastita izrada)

1.8.3. Dio baze podataka *Djeca*

Dizajnerski prikaz *View*-a dijela baze podataka vezan uz djecu aplikacije izrađen u MS SQL Server Management Studiu prikazan je na slici 27.



Slika 27. *DjecaView*

(Izvor: vlastita izrada)

DjecaView osim što uključuje sve atribute tablica *Djeca, Skupine, Grupe, Gradovi* i *Nacionalnosti* te pripadajuće veze između njih, uključuje i dodatne atribute pod nazivima *ImePrezime, PrezimeIme, MImePrezime, OImePrezime, ImenaRoditelja* i *Dob*. Atributi *ImePrezime* i *PrezimeIme* izgrađeni su spajanjem atributa *DijeteIme* i *DijetePrezime* korištenjem *ISNULL* funkcije. Atribut *MImePrezime* izgrađen je spajanjem atributa *MIme* i *MPrezime* korištenjem *ISNULL* funkcije te predstavlja ime i prezime majke. Atribut *OImePrezime* izgrađen je spajanjem atributa *OIme* i *OPrezime* korištenjem *ISNULL* funkcije te predstavlja ime i prezime oca. Atribut *ImenaRoditelja* izgrađen je spajanjem atributa *Mime*, *MPrezime, OIme* i *OPrezime* korištenjem *ISNULL* funkcije te predstavlja spoj imena i

prezimena obiju roditelja. Atribut *Dob* izrađen je uporabom funkcije *DATEDIFF* koja računa dob djeteta prema njegovom datumu rođenja i trenutnom danu. Cijela skripta *DjecaView* vidljiva je u nastavku, a primjer dijela ispisa skripte prikazan je na slici 28.

```
SELECT dbo.Djeca.IDDijete, dbo.Djeca.DijeteIme,
dbo.Djeca.DijetePrezime, dbo.Djeca.OIB, dbo.Djeca.Spol,
dbo.Djeca.VrticPrije, dbo.Djeca.IDNacionalnost,
dbo.Djeca.MjestoRodjenja, dbo.Djeca.Adresa, dbo.Djeca.IDGrad,
dbo.Djeca.Prebivaliste, dbo.Djeca.MBO, dbo.Djeca.Pedijatar,
dbo.Djeca.PedijatarTelefon, dbo.Djeca.IDSkupina, dbo.Djeca.Napomena,
dbo.Djeca.MIme, dbo.Djeca.MPrezime, dbo.Djeca.MOIB,
dbo.Djeca.IDMNacionalnost, dbo.Djeca.MMjestoRodjenja,
dbo.Djeca.MAdresa, dbo.Djeca.IDMGrad, dbo.Djeca.MPrebivaliste,
dbo.Djeca.MTelefon, dbo.Djeca.MEmail, dbo.Djeca.MZaposlena,
dbo.Djeca.MZanimanje, dbo.Djeca.MPoslodavac,
dbo.Djeca.MPoslovniTelefon, dbo.Djeca.MPoslovniEmail,
dbo.Djeca.MBrOsobneIskaznice, dbo.Djeca.OIme, dbo.Djeca.OPrezime,
dbo.Djeca.OOIB, Gradovi_2.GradNaziv, Gradovi_2.PostBr,
Nacionalnosti_1.NacionalnostNaziv, dbo.Grupe.GrupaNaziv,
dbo.Skupine.SkupinaNaziv, dbo.Skupine.IDGrupa,
Nacionalnosti_2.NacionalnostNaziv AS MNacionalnostNaziv,
Gradovi_1.GradNaziv AS MGradNaziv, Gradovi_1.PostBr AS MPostBr,
dbo.Djeca.IDONacionalnost, dbo.Nacionalnosti.NacionalnostNaziv AS
ONacionalnostNaziv, dbo.Djeca.IDOGrad, dbo.Gradovi.GradNaziv AS
OGradNaziv, dbo.Gradovi.PostBr AS OPostBr, dbo.Djeca.OMjestoRodjenja,
dbo.Djeca.OAdresa, dbo.Djeca.OPrebivaliste, dbo.Djeca.OTelefon,
dbo.Djeca.OEmail, dbo.Djeca.OZaposlen, dbo.Djeca.OZanimanje,
dbo.Djeca.OPoslodavac, dbo.Djeca.OPoslovniTelefon,
dbo.Djeca.OPoslovniEmail, dbo.Djeca.OBrOsobneIskaznice,
dbo.Djeca.MONapomena, dbo.Djeca.Poteskoce,
dbo.FormatDD_MM_YYYY(dbo.Djeca.DatumRodjenja) AS DatumRodjenja,
dbo.FormatDD_MM_YYYY(dbo.Djeca.ODatumRodjenja) AS ODatumRodjenja,
dbo.FormatDD_MM_YYYY(dbo.Djeca.MDatumRodjenja) AS MDatumRodjenja,
ISNULL(dbo.Djeca.DijeteIme, N'') + N' ' +
ISNULL(dbo.Djeca.DijetePrezime, N'') AS ImePrezime,
ISNULL(dbo.Djeca.DijetePrezime, N'') + N' ' +
ISNULL(dbo.Djeca.DijeteIme, N'') AS PrezimeIme,
dbo.Djeca.DatumRodjenja AS DatumRodjenjaDijete,
dbo.Djeca.SlikaDijete, ISNULL(dbo.Djeca.MIme, N'') + N' ' +
ISNULL(dbo.Djeca.MPrezime, N'') AS MImePrezime,
ISNULL(dbo.Djeca.OIme, N'') + N' ' + ISNULL(dbo.Djeca.OPrezime, N'')
AS OImePrezime, CASE WHEN ISNULL(dbo.Djeca.MIme, N'') + N' ' +
ISNULL(dbo.Djeca.MPrezime, N'') <> ' ' AND ISNULL(dbo.Djeca.OIme,
N'') + N' ' + ISNULL(dbo.Djeca.OPrezime, N'') <> ' ' THEN
ISNULL(dbo.Djeca.MIme, N'') + N' ' + ISNULL(dbo.Djeca.MPrezime, N'')
+ ' i ' + ISNULL(dbo.Djeca.OIme, N'') + N' ' +
ISNULL(dbo.Djeca.OPrezime, N'') ELSE '' END AS ImenaRoditelja,
DATEDIFF(month, dbo.Djeca.DatumRodjenja, GETDATE()) AS Dob,
dbo.Djeca.MatKnjigaNapomena
FROM dbo.Nacionalnosti AS Nacionalnosti_2 RIGHT OUTER JOIN
dbo.Gradovi RIGHT OUTER JOIN dbo.Djeca ON dbo.Gradovi.IDGrad =
dbo.Djeca.IDOGrad LEFT OUTER JOIN dbo.Nacionalnosti ON
dbo.Djeca.IDONacionalnost = dbo.Nacionalnosti.IDNacionalnost LEFT
OUTER JOIN dbo.Gradovi AS Gradovi_1 ON dbo.Djeca.IDMGrad =
Gradovi_1.IDGrad ON Nacionalnosti_2.IDNacionalnost =
dbo.Djeca.IDMNacionalnost LEFT OUTER JOIN dbo.Grupe RIGHT OUTER JOIN
dbo.Skupine ON dbo.Grupe.IDGrupa = dbo.Skupine.IDGrupa ON
dbo.Djeca.IDSkupina = dbo.Skupine.IDSkupina LEFT OUTER JOIN
```

```
dbo.Nacionalnosti AS Nacionalnosti_1 ON dbo.Djeca.IDNacionalnost =
Nacionalnosti_1.IDNacionalnost LEFT OUTER JOIN dbo.Gradovi AS
Gradovi_2 ON dbo.Djeca.IDGrad = Gradovi_2.IDGrad
```

Programski kod 7. SQL skripta *DjecaView*

(Izvor: vlastita izrada)

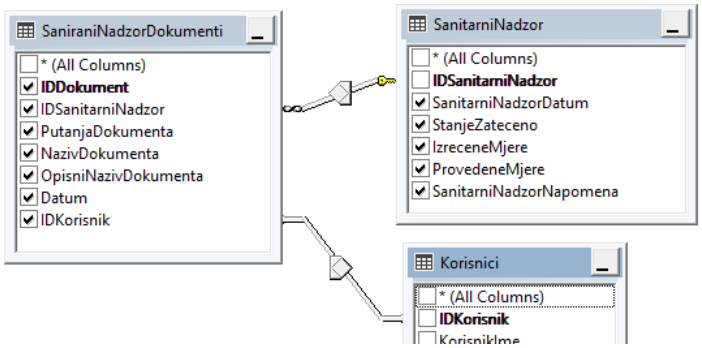
IDDijete	Dijetelme	DijetePrezime	OIB	Spol	VrticPrije	IDNacionalnost	MjestoRodjenja	Adresa	IDGrad	Prebivaliste	MBO	Pedijatar	PedijatarTelefon	IDSkupin
1	Karlo	Ribić	76097716319	M	False	7	Varaždin	Selnik 3	1	Selnik 3	264764690	Bojana Ričko	781 082	5
2	Jan	Šimek	94229317682	M	False	1	Varaždin	C. Tužno 4	2	C. Tužno 4	264447460	Dubravka Cikač...	741 336	5
3	Antonio	Peharda	57613495697	M	False	1	Varaždin	Bikovec 56	3	Bikovec 56		Bojana Ričko	781 082	5
4	Katarina	Šimunić	15856799071	Ž	False	1	Varaždin	Biljevec 49	4	Biljevec 49	2701016325002	Bojana Ričko	781 082	4
5	Lucija	Marčec	21508892093	Ž	False	1	Varaždin	Korenjak 15	5	Korenjak 15	264706034	Mirjana Kolarek...	393 540	5
6	Korana	Hanžek	54353496654	Ž	False	1	Varaždin	Selnik 115	6	Selnik 115	264597745	Bojana Ričko	781 082	5
7	Ana	Vrbanec	50668016591	Ž	False	1	Varaždin	Selnik 22	7	Selnik 22	0510014325059	Anela Daraboš	321 300	7
8	Alen	Vrbanec	03259099643	M	False	1	Varaždin	Selnik 22	8	Selnik 22		Anela Daraboš	321 300	7
9	Viktor	Kračun	47894349604	M	False	1	Varaždin	Greda 172	9	Greda 172	263551084	Bojana Ričko	781 082	7

Slika 28. Ispis SQL skripte *DjecaView*

(Izvor: vlastita izrada)

1.8.4. Dio baze podataka *Sanitarni nadzor*

Dizajnerski prikaz *View*-a dijela baze podataka vezan uz sanitarni nadzor izrađen u MS SQL Server Management Studiu prikazan je na slici 29.



Slika 29. SanitarniNadzorView

(Izvor: vlastita izrada)

SanitarniNadzorView osim što uključuje atribute tablica *SanitarniNadzor, Korisnici* i *SanitarniNadzorDokumenti* te pripadajuće veze između njih, uključuje i dodatni atribut pod nazivom *KorisnikImePrezime.* Izgrađen je spajanjem atributa *KorisnikIme* i *KorisnikPrezime* korištenjem *ISNULL* funkcije. Cijela skripta *SanitarniNadzorView* vidljiva je u nastavku, a primjer dijela ispisa skripte prikazan je na slici 30.

```
SELECT dbo.SaniraniNadzorDokumenti.IDDokument,
dbo.SaniraniNadzorDokumenti.IDSanitarniNadzor,
dbo.SaniraniNadzorDokumenti.PutanjaDokumenta,
dbo.SaniraniNadzorDokumenti.NazivDokumenta,
dbo.SaniraniNadzorDokumenti.OpisniNazivDokumenta,
dbo.SaniraniNadzorDokumenti.Datum,
dbo.SaniraniNadzorDokumenti.IDKorisnik,
```

```
dbo.SanitarniNadzor.StanjeZateceno,
dbo.SanitarniNadzor.SanitarniNadzorDatum,
dbo.SanitarniNadzor.IzreceneMjere,
dbo.SanitarniNadzor.ProvedeneMjere,
dbo.SanitarniNadzor.SanitarniNadzorNapomena,
ISNULL(dbo.Korisnici.KorisnikIme, N'') + N' ' +
ISNULL(dbo.Korisnici.KorisnikPrezime, N'') AS KorisnikImePrezime
FROM dbo.SaniraniNadzorDokumenti LEFT OUTER JOIN dbo.Korisnici ON
dbo.SaniraniNadzorDokumenti.IDKorisnik = dbo.Korisnici.IDKorisnik
RIGHT OUTER JOIN dbo.SanitarniNadzor ON
dbo.SaniraniNadzorDokumenti.IDSanitarniNadzor =
dbo.SanitarniNadzor.IDSanitarniNadzor
```

Programski kod 8. SQL skripta *SanitarniNadzorView*

(Izvor: vlastita izrada)

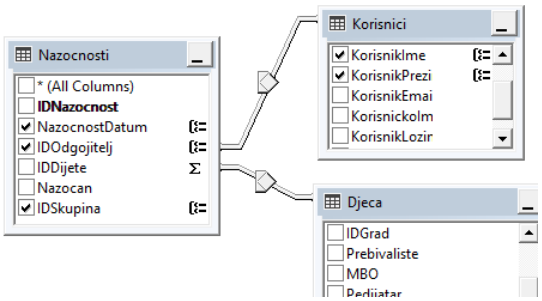
IDDokument	IDSanitarniNa...	PutanjaDoku...	NazivDokume...	OpisniNazivDo...	Datum	IDKorisnik	StanjeZateceno	SanitarniNadz...	IzreceneMjere	ProvedeneMjere	SanitarniNadz...	KorisnikImePr...
29	3	dokumenti/san...	20230510T1104_...	Sanitarni nadzo...	2023-05-10 11:0...	7	Lorem Ipsum is...	2023-05-08 00:0...	Lorem Ipsum is...	Lorem Ipsum is...	Lorem Ipsum is...	Admin Admin

Slika 30. Ispis SQL skripte *SanitarniNadzorView*

(Izvor: vlastita izrada)

1.8.5. Dio baze podataka *Nazočnost djece*

Dizajnerski prikaz *View*-a dijela baze podataka vezan uz evidenciju nazočnosti djece izrađen u MS SQL Server Management Studiu prikazan je na slici 31.



Slika 31. NazocnostiView

(Izvor: vlastita izrada)

NazocnostiView osim što uključuje atribute tablice *Nazocnosti* te pripadajuće veze između nje i tablica *Korisnici* i *Djeca*, uključuje i dodatne atribute pod nazivima *BrNazocneDjece, BrojNenaz* i *UkBrojDjece.* Atributi *BrNazocneDjece* i *BrojNenaz* realizirani su postavljanjem *CASE* uvjeta nad atributom *Nazocan*. Atribut *UkBrojDjece* realiziran je brojanjem (funkcija *COUNT*) atributa *IDDijete*. Cijela skripta *NazocnostiView* vidljiva je u nastavku, a primjer dijela ispisa skripte prikazan je na slici 32.

```
SELECT dbo.Nazocnosti.NazocnostDatum, dbo.Nazocnosti.IDOdgojitelj,
dbo.Korisnici.KorisnikIme, dbo.Korisnici.KorisnikPrezime, SUM(CASE
WHEN Nazocan = 1 THEN 1 ELSE 0 END) AS BrNazocneDjece, SUM(CASE WHEN
Nazocan = 0 THEN 1 ELSE 0 END) AS BrojNenaz,
COUNT(dbo.Nazocnosti.IDDijete) AS UkBrojDjece,
dbo.Nazocnosti.IDSkupina
FROM dbo.Nazocnosti LEFT OUTER JOIN dbo.Djeca ON
dbo.Nazocnosti.IDDijete = dbo.Djeca.IDDijete LEFT OUTER JOIN
dbo.Korisnici ON dbo.Nazocnosti.IDOdgojitelj =
dbo.Korisnici.IDKorisnik
GROUP BY dbo.Nazocnosti.NazocnostDatum, dbo.Nazocnosti.IDOdgojitelj,
dbo.Korisnici.KorisnikIme, dbo.Korisnici.KorisnikPrezime,
dbo.Nazocnosti.IDSkupina
```

Programski kod 9. SQL skripta *NazocnostiView*

(Izvor: vlastita izrada)

NazocnostDat...	IDOdgojitelj	KorisnikIme	KorisnikPrezime	BrNazocneDjece	BrojNenaz	UkBrojDjece	IDSkupina
2023-04-10 00:0...	7	Admin	Admin	1	0	1	5
2023-04-11 00:0...	7	Admin	Admin	1	3	4	1
2023-04-11 00:0...	7	Admin	Admin	1	19	20	3
2023-04-11 00:0...	7	Admin	Admin	7	4	11	5
2023-04-12 00:0...	7	Admin	Admin	4	0	4	1
2023-04-12 00:0...	7	Admin	Admin	12	8	20	3
2023-04-13 00:0...	7	Admin	Admin	3	1	4	1
2023-04-13 00:0...	7	Admin	Admin	5	15	20	3

Slika 32. Ispis SQL skripte *NazocnostiView*

(Izvor: vlastita izrada)

Za potrebe izdvajanja zadnjih evidencija nazočnosti po skupinama, napravljena je skripta *NazocnostiZadnjiDatum* prikazana u nastavku, a njezin ispis vidljiv je na slici 33.

```
SELECT MAX(dbo.Nazocnosti.NazocnostDatum) AS ZadnjiDatum,
dbo.Djeca.IDSkupina
FROM dbo.Nazocnosti LEFT OUTER JOIN dbo.Djeca ON
dbo.Nazocnosti.IDDijete = dbo.Djeca.IDDijete
GROUP BY dbo.Djeca.IDSkupina
```

Programski kod 10. SQL skripta *NazocnostiZadnjiDatum*

(Izvor: vlastita izrada)

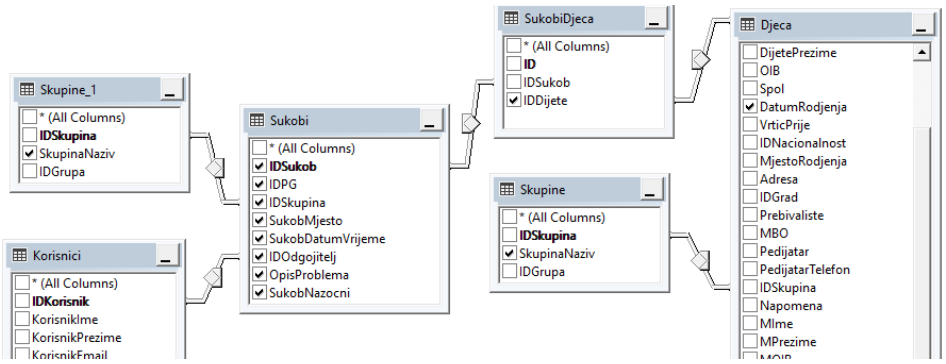
ZadnjiDatum	IDSkupina
2023-05-10 00:00:00.000	1
2023-05-05 00:00:00.000	3
2023-04-13 00:00:00.000	5

Slika 33. Ispis SQL skripte *NazocnostiZadnjiDatum*

(Izvor: vlastita izrada)

1.8.6. Dio baze podataka *Sukobi među djecom*

Dizajnerski prikaz *View*-a dijela baze podataka vezan uz evidenciju sukoba među djecom izrađen u MS SQL Server Management Studiu prikazan je na slici 34.



Slika 34. *SukobiDjecaView*

(Izvor: vlastita izrada)

SukobiDjecaView osim što uključuje atribute tablica *Djeca, Skupine, SukobiDjeca, Sukobi* i *Korisnici* te pripadajuće veze između njih, uključuje i dodatne atribute pod nazivima *DijeteImePrezime, MImePrezime, OImePrezime* i *KorisnikImePrezime*. Spomenuti atributi izgrađeni su spajanjem atributa imena i prezimena djeteta, majke, oca i odgojitelja korištenjem *ISNULL* funkcije. Cijela skripta *SukobiDjecaView* vidljiva je u nastavku, a primjer dijela ispisa skripte prikazan je na slici 35.

```
SELECT dbo.Sukobi.IDSukob, dbo.Sukobi.IDPG, dbo.Sukobi.IDSkupina,
dbo.Sukobi.SukobMjesto, dbo.Sukobi.SukobDatumVrijeme,
dbo.Sukobi.IDOdgojitelj, dbo.Sukobi.OpisProblema,
dbo.SukobiDjeca.IDDijete, ISNULL(dbo.Djeca.DijeteIme, N'') + N' ' +
ISNULL(dbo.Djeca.DijetePrezime, N'') AS DijeteImePrezime,
ISNULL(dbo.Djeca.MIme, N'') + N' ' + ISNULL(dbo.Djeca.MPrezime, N'')
AS MImePrezime, Skupine_1.SkupinaNaziv, ISNULL(dbo.Djeca.OIme, N'') +
N' ' + ISNULL(dbo.Djeca.OPrezime, N'') AS OImePrezime,
ISNULL(dbo.Korisnici.KorisnikIme, N'') + N' ' +
ISNULL(dbo.Korisnici.KorisnikPrezime, N'') AS KorisnikImePrezime,
dbo.Djeca.DatumRodjenja, dbo.Skupine.SkupinaNaziv AS SkupinaDjeteta,
dbo.Sukobi.SukobNazocni
FROM dbo.SukobiDjeca LEFT OUTER JOIN dbo.Skupine RIGHT OUTER JOIN
dbo.Djeca ON dbo.Skupine.IDSkupina = dbo.Djeca.IDSkupina LEFT OUTER
JOIN
ON dbo.SukobiDjeca.IDDijete = dbo.Djeca.IDDijete RIGHT OUTER JOIN
dbo.Sukobi LEFT OUTER JOIN dbo.Skupine AS Skupine_1 ON
dbo.Sukobi.IDSkupina = Skupine_1.IDSkupina LEFT OUTER JOIN
dbo.Korisnici ON dbo.Sukobi.IDOdgojitelj = dbo.Korisnici.IDKorisnik
ON dbo.SukobiDjeca.IDSukob = dbo.Sukobi.IDSukob
```

Programski kod 11. SQL skripta *SukobiDjecaView*

(Izvor: vlastita izrada)

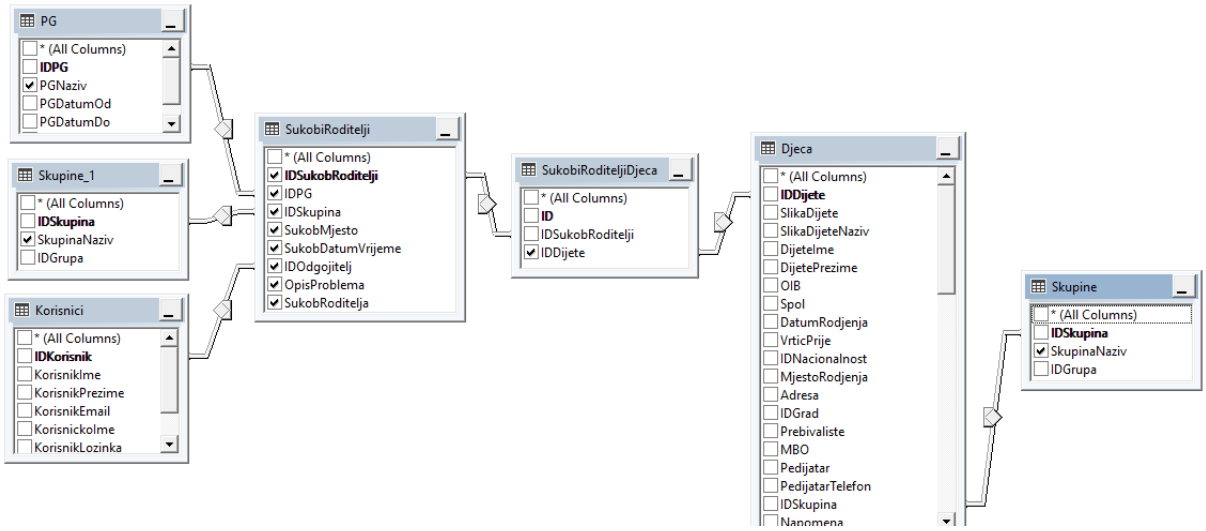
IDSukob	IDPG	IDSkupina	SukobMjesto	SukobDatumV...	IDOdgojitelj	OpisProblema	MAdresa	IDDijete	MTelefon	OAdresa	OTelefon	DijetelmePrezi...	Ml
3	1	3	NULL	2023-04-25 12:5...	7	NULL	NULL	NULL	NULL	NULL	NULL		
7	1	1	NULL	2023-04-25 14:0...	7	NULL	NULL	50	NULL	NULL	NULL	Lucija Kumrić	
7	1	1	NULL	2023-04-25 14:0...	7	NULL	NULL	34	NULL	NULL	NULL	Martin Banec	
7	1	1	NULL	2023-04-25 14:0...	7	NULL	NULL	66	NULL	NULL	NULL	Anja Novak	
7	1	1	NULL	2023-04-25 14:0...	7	NULL	NULL	29	NULL	NULL	NULL	Jan Možanić	
16	1	1	NULL	2023-05-04 00:0...	7	NULL	NULL	50	NULL	NULL	NULL	Lucija Kumrić	
16	1	1	NULL	2023-05-04 00:0...	7	NULL	NULL	43	NULL	NULL	NULL	Elena Kutnjak	

Slika 35. Ispis SQL skripte *SukobiDjecaView*

(Izvor: vlastita izrada)

1.8.7. Dio baze podataka *Sukobi među roditeljima*

Dizajnerski prikaz *View*-a dijela baze podataka vezan uz evidenciju sukoba među roditeljima izrađen u MS SQL Server Management Studiu prikazan je na slici 36.



Slika 36. *SukobiRoditeljiView*

(Izvor: vlastita izrada)

SukobiRoditeljiView osim što uključuje atribute tablica *Djeca, Skupine, PG, SukobiRoditeljiDjeca, SukobiRoditelji* i *Korisnici* te pripadajuće veze između njih, uključuje i dodatne atribute pod nazivima *DijeteImePrezime* i *KorisnikImePrezime*. Spomenuti atributi izgrađeni su spajanjem atributa imena i prezimena djeteta te odgojitelja korištenjem *ISNULL* funkcije. Cijela skripta *SukobiRoditeljiView* vidljiva je u nastavku, a primjer dijela ispisa skripte prikazan je na slici 37.

```
SELECT dbo.SukobiRoditelji.IDPG,
dbo.SukobiRoditelji.IDSukobRoditelji, dbo.SukobiRoditelji.IDSkupina,
dbo.SukobiRoditelji.SukobMjesto,
dbo.SukobiRoditelji.SukobDatumVrijeme,
dbo.SukobiRoditelji.IDOdgojitelj, dbo.SukobiRoditelji.OpisProblema,
dbo.SukobiRoditelji.SukobRoditelja, Skupine_1.SkupinaNaziv,
dbo.PG.PGNaziv, dbo.SukobiRoditeljiDjeca.IDDijete,
ISNULL(dbo.Djeca.DijeteIme, N'') + N' ' +
```

```
ISNULL(dbo.Djeca.DijetePrezime, N'') AS DijeteImePrezime,
ISNULL(dbo.Korisnici.KorisnikIme, N'') + N' ' +
ISNULL(dbo.Korisnici.KorisnikPrezime, N'') AS KorisnikImePrezime,
dbo.Skupine.SkupinaNaziv AS SkupinaDjeteta
FROM dbo.Djeca LEFT OUTER JOIN dbo.Skupine ON dbo.Djeca.IDSkupina =
dbo.Skupine.IDSkupina RIGHT OUTER JOIN dbo.PG RIGHT OUTER JOIN
dbo.SukobiRoditelji ON dbo.PG.IDPG = dbo.SukobiRoditelji.IDPG LEFT
OUTER JOIN dbo.Korisnici ON dbo.SukobiRoditelji.IDOdgojitelj =
dbo.Korisnici.IDKorisnik LEFT OUTER JOIN dbo.SukobiRoditeljiDjeca ON
dbo.SukobiRoditelji.IDSukobRoditelji =
dbo.SukobiRoditeljiDjeca.IDSukobRoditelji ON dbo.Djeca.IDDijete =
dbo.SukobiRoditeljiDjeca.IDDijete LEFT OUTER JOIN dbo.Skupine AS
Skupine_1 ON dbo.SukobiRoditelji.IDSkupina = Skupine_1.IDSkupina
```

Programski kod 12. SQL skripta *SukobiRoditeljiView*

(Izvor: vlastita izrada)

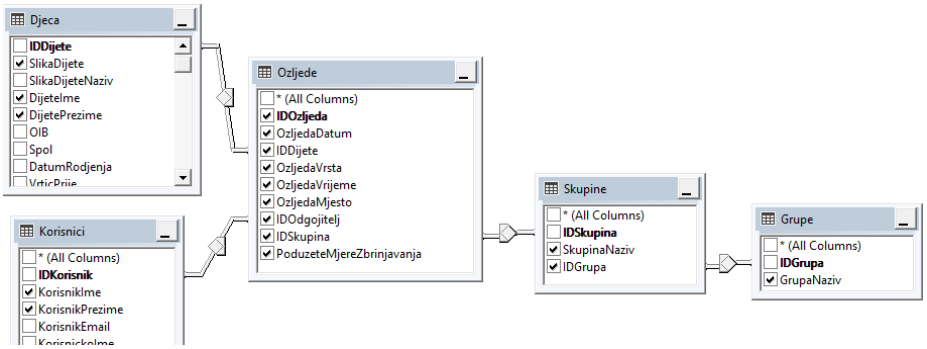
IDPG	IDSukobRoditelji	IDSkupina	SukobMjesto	SukobDatumV...	IDOdgojitelj	OpisProblema	SukobRoditelja	SkupinaNaziv	PGNaziv	IDDije...	DijetelmePrezi...	KorisnikImePr...	SkupinaDjeteta
1	1	1	Dvorište vrtića	2023-05-09 15:0...	6	Svađa majki	NULL	Bubamare	2022/23	50	Lucija Kumrić	Iva Ivić	Bubamare
1	1	1	Dvorište vrtića	2023-05-09 15:0...	6	Svađa majki	NULL	Bubamare	2022/23	47	Dejan Mašić	Iva Ivić	Bubamare
1	2	1	NULL	2023-05-10 00:0...	7	sdfsf	očevi	Bubamare	2022/23	47	Dejan Mašić	Admin Admin	Bubamare
1	2	1	NULL	2023-05-10 00:0...	7	sdfsf	očevi	Bubamare	2022/23	50	Lucija Kumrić	Admin Admin	Bubamare
1	2	1	NULL	2023-05-10 00:0...	7	sdfsf	očevi	Bubamare	2022/23	43	Elena Kutnjak	Admin Admin	Bubamare

Slika 37. Ispis SQL skripte *SukobiRoditeljiView*

(Izvor: vlastita izrada)

1.8.8. Dio baze podataka *Ozljede djece*

Dizajnerski prikaz *View*-a dijela baze podataka vezan uz evidenciju ozljeda djece izrađen u MS SQL Server Management Studiu prikazan je na slici 38.



Slika 38. *OzljedeView*

(Izvor: vlastita izrada)

OzljedeView osim što uključuje atribute tablica *Djeca, Korisnici, Ozljede, Skupine* i *Grupe* te pripadajuće veze između njih, uključuje i dodatne atribute pod nazivima *ImePrezime, PrezimeIme* i *OdgojiteljImePrezime*. Spomenuti atributi izgrađeni su spajanjem atributa imena i prezimena djeteta te odgojitelja korištenjem *ISNULL* funkcije. Nad atributom *OzljedaDatum* provedena je funkcija *FormatDD_MM_YYYY* koja pretvara datum u format oblika *dd.MM.yyyy*. (npr. *22.03.2023.*) Njezina skripta prikazana je u nastavku.

```
USE [DV] GO SET ANSI_NULLS ON GO SET QUOTED_IDENTIFIER ON GO ALTER
FUNCTION [dbo].[FormatDD_MM_YYYY] ( @Datum as datetime ) RETURNS
varchar(12) AS BEGIN -- Declare the return variable here DECLARE
@Result varchar(12) -- Add the T-SQL statements to compute the return
value here SELECT @Result =
dbo.FormatNule(DATEPART(dd,@Datum),2)+'.'+dbo.FormatNule(DATEPART(mm,
@Datum),2)+'.' +convert(char(4),DATEPART(yyyy,@Datum))+'.' -- Return
the result of the function RETURN @Result END
```

Programski kod 13. SQL skripta funkcije *FormatDD_MM_YYYY*

(Izvor: vlastita izrada)

Cijela skripta *OzljedeView* vidljiva je u nastavku, a primjer dijela ispisa skripte prikazan je na slici 39.

```
SELECT dbo.Ozljede.IDOzljeda, dbo.Ozljede.IDDijete,
dbo.Ozljede.OzljedaVrsta, dbo.Ozljede.OzljedaVrijeme,
dbo.Ozljede.OzljedaMjesto, dbo.Ozljede.IDSkupina,
dbo.Ozljede.PoduzeteMjereZbrinjavanja, dbo.Skupine.SkupinaNaziv,
dbo.Ozljede.IDOdgojitelj, dbo.Korisnici.KorisnikIme,
dbo.Korisnici.KorisnikPrezime, dbo.Grupe.GrupaNaziv,
dbo.Skupine.IDGrupa, dbo.Djeca.DijeteIme, dbo.Djeca.DijetePrezime,
dbo.FormatDD_MM_YYYY(dbo.Ozljede.OzljedaDatum) AS OzljedaDatum,
ISNULL(dbo.Djeca.DijeteIme, N'') + N' ' +
ISNULL(dbo.Djeca.DijetePrezime, N'') AS ImePrezime,
ISNULL(dbo.Djeca.DijetePrezime, N'') + N' ' +
ISNULL(dbo.Djeca.DijeteIme, N'') AS PrezimeIme,
ISNULL(dbo.Korisnici.KorisnikIme, N'') + N' ' +
ISNULL(dbo.Korisnici.KorisnikPrezime, N'') AS OdgojiteljImePrezime,
dbo.Ozljede.OzljedaDatum AS Datum, dbo.Djeca.SlikaDijete
FROM dbo.Djeca RIGHT OUTER JOIN dbo.Ozljede ON dbo.Djeca.IDDijete =
dbo.Ozljede.IDDijete LEFT OUTER JOIN dbo.Skupine LEFT OUTER JOIN
dbo.Grupe ON dbo.Skupine.IDGrupa = dbo.Grupe.IDGrupa ON
dbo.Ozljede.IDSkupina = dbo.Skupine.IDSkupina LEFT OUTER JOIN
dbo.Korisnici ON dbo.Ozljede.IDOdgojitelj = dbo.Korisnici.IDKorisnik
```

Programski kod 14. SQL skripta *OzljedeView*

(Izvor: vlastita izrada)

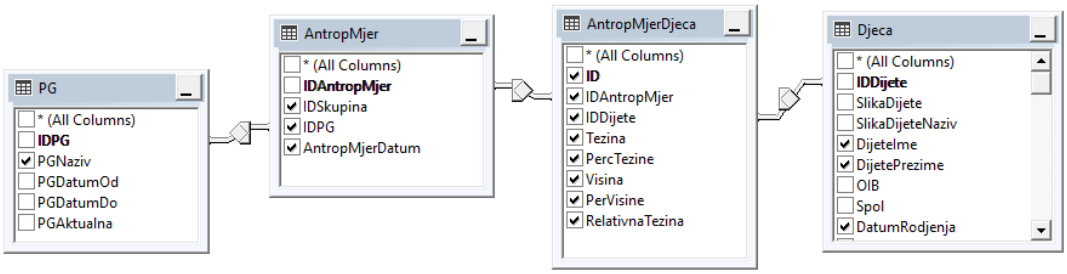
IDOzljeda	IDDijete	OzljedaVrsta	OzljedaVrijeme	OzljedaMjesto	IDSkupina	PoduzeteMjer...	SkupinaNaziv	IDOdgojitelj	KorisnikIme	KorisnikPrezime	GrupaNaziv	IDGrupa	DijeteIme	DijetePre
16	1	Pad na glavu	Iza 14 h	Prostorije vrtića	3	Stavljen oblog	Ježići	13	Ana	Anić	Vrtić	2	Karlo	Ribić
23	37	Porezan prst.	Oko 14 sati.	Blagovaonica v...	7	Sanirana rana.	Slonići	6	Iva	Ivić	Jaslice	1	Teodor	Tomek
39	65	NULL	NULL	NULL	7	NULL	Slonići	7	Admin	Admin	Jaslice	1	Petar	Možanić
40	0	NULL	NULL	NULL	0	NULL	NULL	0	NULL	NULL	NULL	NULL	NULL	NULL
41	0	NULL	NULL	NULL	0	NULL	NULL	0	NULL	NULL	NULL	NULL	NULL	NULL
42	65	Pad na koljeno.	Oko 10 sati.	Dvorište vrtića.	7	NULL	Slonići	7	Admin	Admin	Jaslice	1	Petar	Možanić

Slika 39. Ispis SQL skripte *SukobiRoditeljiView*

(Izvor: vlastita izrada)

1.8.9. Dio baze podataka *Antropometrijska mjerenja djece*

Dizajnerski prikaz *View*-a dijela baze podataka vezan uz antropometrijsko mjerenje djece izrađen u MS SQL Server Management Studiu prikazan je na slici 40.



Slika 40. *AntropMjerView*

(Izvor: vlastita izrada)

AntropMjerView osim što uključuje atribute tablica *PG*, *AntropMjer*, *AntropMjerDjeca* i *Djeca* te pripadajuće veze između njih, uključuje i dodatni atribut pod nazivom *DijeteImePrezime.* Spomenuti atribut izgrađen je spajanjem atributa imena i prezimena djeteta korištenjem *ISNULL* funkcije. Cijela skripta *AntropMjerView* vidljiva je u nastavku, a primjer dijela ispisa skripte prikazan je na slici 41.

```
SELECT dbo.AntropMjer.IDSkupina, dbo.PG.PGNaziv, dbo.AntropMjer.IDPG,
dbo.AntropMjerDjeca.IDAntropMjer, dbo.AntropMjerDjeca.IDDijete,
dbo.AntropMjerDjeca.Tezina, dbo.AntropMjerDjeca.PercTezine,
dbo.AntropMjerDjeca.Visina, dbo.AntropMjerDjeca.PerVisine,
dbo.AntropMjerDjeca.RelativnaTezina, dbo.Djeca.DatumRodjenja,
dbo.Djeca.DijeteIme, dbo.Djeca.DijetePrezime,
ISNULL(dbo.Djeca.DijeteIme, N'') + N' ' +
ISNULL(dbo.Djeca.DijetePrezime, N'') AS DijeteImePrezime,
dbo.AntropMjerDjeca.ID, dbo.AntropMjer.AntropMjerDatum
FROM dbo.AntropMjerDjeca LEFT OUTER JOIN dbo.Djeca ON
dbo.AntropMjerDjeca.IDDijete = dbo.Djeca.IDDijete LEFT OUTER JOIN
dbo.PG RIGHT OUTER JOIN dbo.AntropMjer ON dbo.PG.IDPG =
dbo.AntropMjer.IDPG ON dbo.AntropMjerDjeca.IDAntropMjer =
dbo.AntropMjer.IDAntropMjer
```

Programski kod 15. SQL skripta *AntropMjerView*

(Izvor: vlastita izrada)

IDSkupina	PGNaziv	IDPG	IDAntropMjer	IDDijete	Tezina	PercTezine	Visina	PerVisine	RelativnaTezina	DatumRodjenja	Dijetelme	DijetePrezime	DijetelmePrezi...	ID
1	2022/23	1	1	43	25	0	120	0	0	2014-05-13 00:0...	Elena	Kutnjak	Elena Kutnjak	270
1	2022/23	1	1	47	20	0	0	0	0	2016-08-26 00:0...	Dejan	Mašić	Dejan Mašić	271
1	2022/23	1	1	50	0	0	0	0	0	2014-05-13 00:0...	Lucija	Kumrić	Lucija Kumrić	272
1	2022/23	1	1	66	0	0	0	0	0	2017-09-11 00:0...	Anja	Novak	Anja Novak	273
1	2022/23	1	2	43	15	23	233	232	232	2014-05-13 00:0...	Elena	Kutnjak	Elena Kutnjak	274

Slika 41. Ispis SQL skripte *AntropMjerView*

(Izvor: vlastita izrada)

Za potrebe prikaza broja izmjerene djece naspram ukupnog broja djece pojedine skupine tijekom evidencija antropometrijskih mjerenja, napravljena je skripta *AntropMjerPopisView.* Dodani atributi su *BrUkupneDjece* i *BrIzmjereneDjece*. Atribut *BrIzmjereneDjece* realiziran je postavljanjem *CASE* uvjeta nad atributima *Tezina i Visina* na način da se pomoću *ISNULL* funkcije provjerava imaju li spomenuti atributi vrijednosti. Ako oba atributa imaju vrijednost, pripadajuće dijete se pribraja u broj izmjerene djece. Atribut *BrUkupneDjece* realiziran je brojanjem (funkcija *COUNT*) atributa *IDDijete*. Cijela skripta *AntropMjerPopisView* vidljiva je u nastavku, a primjer ispisa skripte prikazan je na slici 42.

```
SELECT dbo.AntropMjer.IDAntropMjer,
COUNT(dbo.AntropMjerDjeca.IDDijete) AS BrUkupneDjece, SUM(CASE WHEN
ISNULL(Tezina, 0) <> 0 AND isnull(visina, 0) <> 0 THEN 1 ELSE 0 END)
AS BrIzmjereneDjece, dbo.AntropMjer.IDSkupina,
dbo.AntropMjer.AntropMjerDatum, dbo.AntropMjer.IDPG
FROM dbo.AntropMjer LEFT OUTER JOIN dbo.AntropMjerDjeca ON
dbo.AntropMjer.IDAntropMjer = dbo.AntropMjerDjeca.IDAntropMjer
GROUP BY dbo.AntropMjer.IDAntropMjer, dbo.AntropMjer.IDSkupina,
dbo.AntropMjer.AntropMjerDatum, dbo.AntropMjer.IDPG
```

Programski kod 16. SQL skripta *AntropMjerPopisView*

(Izvor: vlastita izrada)

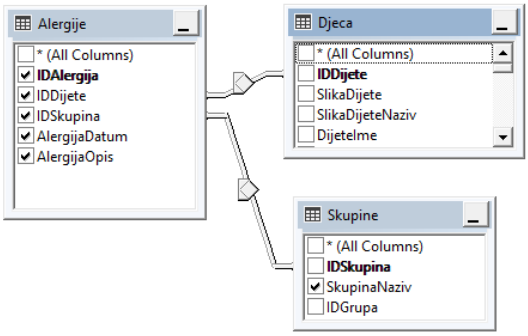
IDAntropMjer	BrUkupneDjece	BrIzmjereneDj...	IDSkupina	AntropMjerDa...	IDPG
1	4	1	1	2023-04-20 00:0...	1
2	4	2	1	2023-04-19 00:0...	1
3	22	1	3	2023-04-21 18:5...	1
5	4	1	1	2023-05-04 15:1...	1
6	4	2	1	2023-05-04 19:3...	1

Slika 42. Ispis SQL skripte *AntropMjerPopisView*

(Izvor: vlastita izrada)

1.8.10. Dio baze podataka *Alergije djece*

Dizajnerski prikaz *View*-a dijela baze podataka vezan uz alergije djece izrađen u MS SQL Server Management Studiu prikazan je na slici 43.



Slika 43. *AlergijeView*

(Izvor: vlastita izrada)

AlergijeView osim što uključuje atribute tablica *Alergije, Skupine* i *Djeca* te pripadajuće veze između njih, uključuje i dodatne atribute pod nazivima *DijeteImePrezime* i *DijetePrezimeIme.* Spomenuti atributi izgrađeni su spajanjem atributa imena i prezimena djeteta korištenjem *ISNULL* funkcije. Cijela skripta *AlergijeView* vidljiva je u nastavku, a primjer ispisa skripte prikazan je na slici 44.

```
SELECT dbo.Alergije.IDAlergija, dbo.Alergije.IDDijete,
dbo.Alergije.IDSkupina, dbo.Alergije.AlergijaDatum,
dbo.Alergije.AlergijaOpis, dbo.Skupine.SkupinaNaziv,
ISNULL(dbo.Djeca.DijeteIme, N'') + N' ' +
ISNULL(dbo.Djeca.DijetePrezime, N'') AS DijeteImePrezime,
ISNULL(dbo.Djeca.DijetePrezime, N'') + N' ' +
ISNULL(dbo.Djeca.DijeteIme, N'') AS DijetePrezimeIme
FROM dbo.Alergije LEFT OUTER JOIN dbo.Skupine ON
dbo.Alergije.IDSkupina = dbo.Skupine.IDSkupina LEFT OUTER JOIN
dbo.Djeca ON dbo.Alergije.IDDijete = dbo.Djeca.IDDijete
```

Programski kod 17. SQL skripta *AlergijeView*

(Izvor: vlastita izrada)

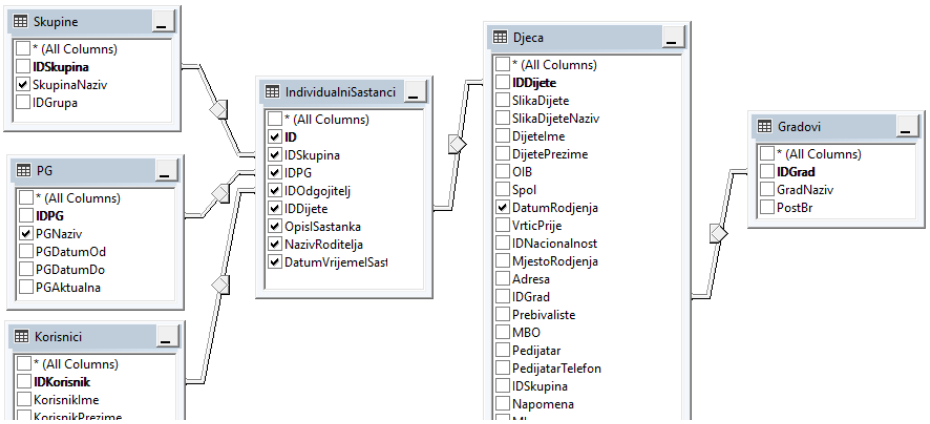
IDAlergija	IDDijete	IDSkupina	AlergijaDatum	AlergijaOpis	SkupinaNaziv	DijetelmePrezi...	DijetePrezimelme
2	47	1	2023-05-09 00:0...	Alergija na jago...	Bubamare	Dejan Mašić	Mašić Dejan
3	28	3	2023-05-09 00:0...	Alergija na praš...	Ježići	Tea Božić	Božić Tea
4	66	1	2023-05-19 00:0...	aha	Bubamare	Anja Novak	Novak Anja
5	52	3	2023-05-10 00:0...	Alergija na oraš...	Ježići	Lana Križnjak-P...	Križnjak-Prešovar Lana
6	50	1	2023-05-10 00:0...	Alergija na oraš...	Bubamare	Lucija Kumrić	Kumrić Lucija

Slika 44. Ispis SQL skripte *AntropMjerPopisView*

(Izvor: vlastita izrada)

1.8.11. Dio baze podataka *Individualni sastanci*

Dizajnerski prikaz *View*-a dijela baze podataka vezan uz individualne sastanke izrađen u MS SQL Server Management Studiu prikazan je na slici 45.



Slika 45. *IndividualniSastanciView*

(Izvor: vlastita izrada)

IndividualniSastanciView osim što uključuje atribute tablica *Skupine, PG, Korisnici, IndividualniSastanci, Djeca* i *Gradovi* te pripadajuće veze između njih, uključuje i dodatne atribute pod nazivima *DijeteImePrezime, DijetePrezimeIme, KorisnikImePrezime, MImePrezime, OImePrezime* i *AdresaGrad.* Atributi *DijeteImePrezime, DijetePrezimeIme, KorisnikImePrezime, MImePrezime, OImePrezime* izgrađeni su spajanjem atributa imena i prezimena djeteta, odgojitelja, majke i oca korištenjem *ISNULL* funkcije. Novoizgrađeni atribut *AdresaGrad* nastao je spajanjem atributa *Adresa* i *Grad* iz tablice *Djeca* također koristeći *ISNULL* funkciju. Cijela skripta *IndividualniSastanciView* vidljiva je u nastavku, a primjer dijela ispisa skripte prikazan je na slici 46.

```
SELECT dbo.IndividualniSastanci.IDSkupina,
dbo.IndividualniSastanci.IDPG, dbo.IndividualniSastanci.IDOdgojitelj,
dbo.IndividualniSastanci.IDDijete,
dbo.IndividualniSastanci.OpisISastanka,
dbo.IndividualniSastanci.NazivRoditelja, dbo.IndividualniSastanci.ID,
dbo.Skupine.SkupinaNaziv, dbo.PG.PGNaziv, ISNULL(dbo.Djeca.DijeteIme,
N'') + N' ' + ISNULL(dbo.Djeca.DijetePrezime, N'') AS
DijeteImePrezime, ISNULL(dbo.Korisnici.KorisnikIme, N'') + N' ' +
ISNULL(dbo.Korisnici.KorisnikPrezime, N'') AS KorisnikImePrezime,
dbo.IndividualniSastanci.DatumVrijemeISastanka,
ISNULL(dbo.Djeca.DijetePrezime, N'') + N' ' +
ISNULL(dbo.Djeca.DijeteIme, N'') AS DijetePrezimeIme,
dbo.Djeca.DatumRodjenja, ISNULL(dbo.Djeca.Adresa, N'') + N' ' +
ISNULL(dbo.Gradovi.GradNaziv, N'') AS AdresaGrad,
ISNULL(dbo.Djeca.MIme, N'') + N' ' + ISNULL(dbo.Djeca.MPrezime, N'')
AS MImePrezime, ISNULL(dbo.Djeca.OIme, N'') + N' ' +
ISNULL(dbo.Djeca.OPrezime, N'') AS OImePrezime
FROM dbo.Gradovi RIGHT OUTER JOIN dbo.Djeca ON dbo.Gradovi.IDGrad =
dbo.Djeca.IDGrad RIGHT OUTER JOIN dbo.IndividualniSastanci LEFT OUTER
JOIN dbo.PG ON dbo.IndividualniSastanci.IDPG = dbo.PG.IDPG LEFT OUTER
JOIN dbo.Skupine ON dbo.IndividualniSastanci.IDSkupina =
dbo.Skupine.IDSkupina ON dbo.Djeca.IDDijete =
dbo.IndividualniSastanci.IDDijete LEFT OUTER JOIN dbo.Korisnici ON
dbo.IndividualniSastanci.IDOdgojitelj = dbo.Korisnici.IDKorisnik
```

Programski kod 18. SQL skripta *IndividualniSastanciView*

(Izvor: vlastita izrada)

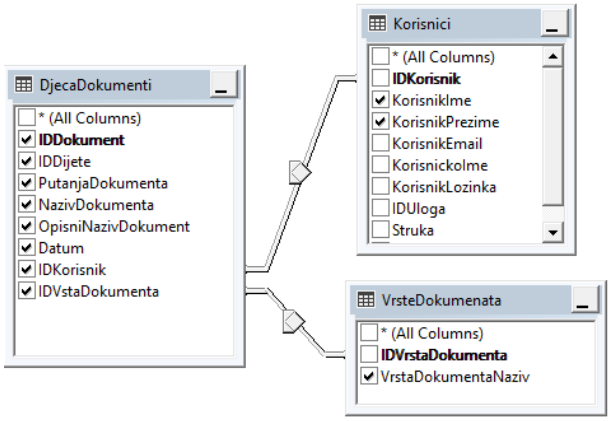
IDSkupina	IDPG	IDOdgoji...	IDDijete	OpisISastanka	NazivRoditelja	ID	SkupinaNaziv	PGNaziv	DijetelmePrezi...	KorisnikImePr...	DatumVrijeme...	DijetePrezimel...	DatumRodjenja	Adresa
5	1	7	3	Dijete je sklon...	Majka	29	Lavići	2022/23	Antonio Peharda	Admin Admin	2023-05-03 00:0...	Peharda Antonio	2015-07-24 00:0...	Bikovec 56 Arža...
5	1	7	59	Dijete se supe...	Otac i majka	36	Lavići	2022/23	Tomo Kolačko	Admin Admin	2023-05-04 10:0...	Kolačko Tomo	2015-05-11 00:0...	V.Nazora 135, L...
1	1	7	66	Razgovor o dj...	Oba roditelja	38	Bubamare	2022/23	Anja Novak	Admin Admin	2023-05-04 15:1...	Novak Anja	2017-09-11 00:0...	Ul.S.Radića 8, G...
1	1	7	47	Dijete odlično...	Majka	40	Bubamare	2022/23	Dejan Mašić	Admin Admin	2023-05-03 13:1...	Mašić Dejan	2016-08-26 00:0...	Jerovec 84 Blinj...

Slika 46. Ispis SQL skripte *IndividualniSastanciView*

(Izvor: vlastita izrada)

1.8.12. Dio baze podataka *Dokumenti djece*

Dizajnerski prikaz *View*-a dijela baze podataka vezan uz dokumente djece izrađen u MS SQL Server Management Studiu prikazan je na slici 47.



Slika 47. *DjecaDokumentiView*

(Izvor: vlastita izrada)

DjecaDokumentiView osim što uključuje atribute tablica *DjecaDokumenti, Djeca* i *Korisnici* te pripadajuće veze između njih, uključuje i dodatni atribut pod nazivom *KorisnikImePrezime* koji je izgrađen spajanjem atributa imena i prezimena korisnika korištenjem *ISNULL* funkcije. Cijela skripta *DjecaDokumentiView* vidljiva je u nastavku, a primjer dijela ispisa skripte prikazan je na slici 48.

```
SELECT dbo.DjecaDokumenti.IDDokument, dbo.DjecaDokumenti.IDDijete,
dbo.DjecaDokumenti.PutanjaDokumenta,
dbo.DjecaDokumenti.NazivDokumenta, dbo.DjecaDokumenti.Datum,
dbo.DjecaDokumenti.IDKorisnik, dbo.Korisnici.KorisnikIme,
dbo.Korisnici.KorisnikPrezime, ISNULL(dbo.Korisnici.KorisnikIme, N'')
+ N' ' + ISNULL(dbo.Korisnici.KorisnikPrezime, N'') AS
ImePrezimeKorisnik, dbo.DjecaDokumenti.OpisniNazivDokumenta,
dbo.DjecaDokumenti.IDVstaDokumenta,
dbo.VrsteDokumenata.VrstaDokumentaNaziv
FROM dbo.DjecaDokumenti LEFT OUTER JOIN dbo.VrsteDokumenata ON
dbo.DjecaDokumenti.IDVstaDokumenta =
dbo.VrsteDokumenata.IDVrstaDokumenta LEFT OUTER JOIN dbo.Korisnici ON
dbo.DjecaDokumenti.IDKorisnik = dbo.Korisnici.IDKorisnik
```

Programski kod 19. SQL skripta *DjecaDokumentiView*

(Izvor: vlastita izrada)

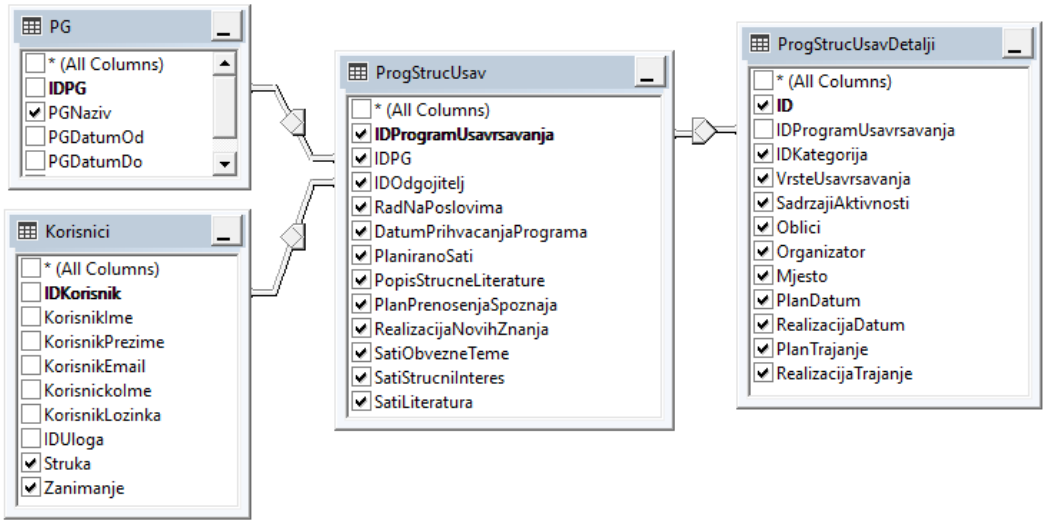
IDDokument	IDDijete	PutanjaDoku...	NazivDokume...	Datum	IDKorisnik	KorisnikIme	KorisnikPrezime	ImePrezimeKo...	OpisniNazivDo...	IDVstaDokume...	VrstaDokumen...
49	50	dokumenti/dje...	20230510T1107_...	2023-05-10 11:0...	7	Admin	Admin	Admin Admin	Nešto	2	Ispričnica
50	50	dokumenti/dje...	20230511T0927_...	2023-05-11 09:2...	7	Admin	Admin	Admin Admin	Nešto	2	Ispričnica

Slika 48. Ispis SQL skripte *DjecaDokumentiView*

(Izvor: vlastita izrada)

1.8.13. Dio baze podataka *Program stručnog usavršavanja odgojitelja*

Dizajnerski prikaz *View*-a dijela baze podataka vezan uz program stručnog usavršavanja odgojitelja izrađen u MS SQL Server Management Studiu prikazan je na slici 49.



Slika 49. *ProgStrucUsavView*

(Izvor: vlastita izrada)

ProgStrucUsavView osim što uključuje atribute tablica *PG, Korisnici, ProgStrucUsav* i *ProgStrucUsavDetalji* te pripadajuće veze između njih, uključuje i dodatni atribut pod nazivom *OdgojiteljImePrezime* koji je izgrađen spajanjem atributa imena i prezimena odgojitelja korištenjem *ISNULL* funkcije. Cijela skripta *ProgStrucUsavView* vidljiva je u nastavku, a primjer dijela ispisa skripte prikazan je na slici 50.

```
SELECT dbo.ProgStrucUsav.IDProgramUsavrsavanja,
dbo.ProgStrucUsav.IDPG, dbo.ProgStrucUsav.IDOdgojitelj,
dbo.ProgStrucUsav.RadNaPoslovima,
dbo.ProgStrucUsav.DatumPrihvacanjaPrograma,
dbo.ProgStrucUsav.PlaniranoSati,
dbo.ProgStrucUsav.PopisStrucneLiterature,
dbo.ProgStrucUsav.PlanPrenosenjaSpoznaja,
dbo.ProgStrucUsav.RealizacijaNovihZnanja,
dbo.ProgStrucUsavDetalji.IDKategorija,
dbo.ProgStrucUsavDetalji.VrsteUsavrsavanja,
dbo.ProgStrucUsavDetalji.SadrzajiAktivnosti,
dbo.ProgStrucUsavDetalji.Oblici,
dbo.ProgStrucUsavDetalji.Organizator,
dbo.ProgStrucUsavDetalji.Mjesto, dbo.ProgStrucUsavDetalji.PlanDatum,
dbo.ProgStrucUsavDetalji.PlanTrajanje,
dbo.ProgStrucUsavDetalji.RealizacijaTrajanje,
dbo.ProgStrucUsavDetalji.RealizacijaDatum,
ISNULL(dbo.Korisnici.KorisnikIme, N'') + N' ' +
ISNULL(dbo.Korisnici.KorisnikPrezime, N'') AS OdgojiteljImePrezime,
```

```
dbo.Korisnici.Struka, dbo.Korisnici.Zanimanje, dbo.PG.PGNaziv,
dbo.ProgStrucUsavDetalji.ID, dbo.ProgStrucUsav.SatiObvezneTeme,
dbo.ProgStrucUsav.SatiStrucniInteres,
dbo.ProgStrucUsav.SatiLiteratura
FROM dbo.ProgStrucUsav LEFT OUTER JOIN dbo.PG ON
dbo.ProgStrucUsav.IDPG = dbo.PG.IDPG LEFT OUTER JOIN dbo.Korisnici ON
dbo.ProgStrucUsav.IDOdgojitelj = dbo.Korisnici.IDKorisnik LEFT OUTER
JOIN dbo.ProgStrucUsavDetalji ON
dbo.ProgStrucUsav.IDProgramUsavrsavanja =
dbo.ProgStrucUsavDetalji.IDProgramUsavrsavanja
```

Programski kod 20. SQL skripta *ProgStrucUsavView*

(Izvor: vlastita izrada)

IDP...	IDPG	IDO...	RadNaPoslovi...	DatumPrihvac...	PlaniranoSati	PopisStrucneLi...	PlanPrenose...	RealizacijaNov...	IDK...	VrsteUsavrsavanja	SadrzajiAktivn...	Oblici	Organizat
1	1	6	Skrbi, njege, od...	2022-08-30 00:0...	50	Literatura 1, lite...	Izrada radion...	Igraonice kroz i...	1	odgojiteljsko vijeće	plan i program	Online edukacija	DV Vrtić
1	1	6	Skrbi, njege, od...	2022-08-30 00:0...	50	Literatura 1, lite...	Izrada radion...	Igraonice kroz i...	2	stručna online edu...	Pedagoška dok...	Online edukacija	Beta eduka
2	2	6	Skrbi, njege, od...	2023-02-20 00:0...	50	Literatura 1, lite...	Izrada radion...	Igraonice kroz i...	1	odgojiteljsko vijeće	plan i program	Online edukacija	DV Vrtić

Slika 50. Ispis SQL skripte *ProgStrucUsavView*

(Izvor: vlastita izrada)

Za potrebe prikaza popisa programa stručnog usavršavanju prema odgojitelju i pedagoškoj godini, napravljena je skripta *ProgStrucUsavPopisView.* Cijela skripta *ProgStrucUsavPopisView* vidljiva je u nastavku, a ispis skripte prikazan je na slici 51.

```
SELECT dbo.ProgStrucUsav.IDProgramUsavrsavanja,
ISNULL(dbo.Korisnici.KorisnikIme, N'') + N' ' +
ISNULL(dbo.Korisnici.KorisnikPrezime, N'') AS OdgojiteljImePrezime,
dbo.Korisnici.Struka, dbo.Korisnici.Zanimanje, dbo.PG.PGNaziv,
dbo.ProgStrucUsav.RadNaPoslovima,
dbo.ProgStrucUsav.DatumPrihvacanjaPrograma, dbo.ProgStrucUsav.IDPG,
dbo.ProgStrucUsav.IDOdgojitelj
FROM dbo.ProgStrucUsav INNER JOIN dbo.PG ON dbo.ProgStrucUsav.IDPG =
dbo.PG.IDPG LEFT OUTER JOIN dbo.Korisnici ON
dbo.ProgStrucUsav.IDOdgojitelj = dbo.Korisnici.IDKorisnik
```

Programski kod 21. SQL skripta *ProgStrucUsavPopisView*

(Izvor: vlastita izrada)

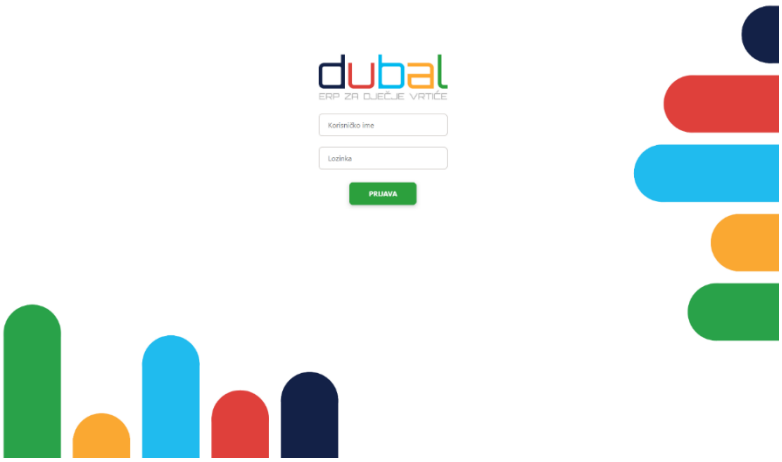
IDProgramUsa...	OdgojiteljIme...	Struka	Zanimanje	PGNaziv	RadNaPoslovi...	DatumPrihvac...	IDPG	IDOdgojitelj
1	Ana Anić	Rani i predškols...	Odgojitelj pred...	2022/23	Skrbi, njege, od...	2022-08-30 00:0...	1	13
2	Iva Ivić	Rani i predškols...	Odgojitelj pred...	2023/24	Skrbi, njege, od...	2023-02-20 00:0...	2	6
3	Iva Ivić	Rani i predškols...	Odgojitelj pred...	2022/23	Odgoj predškol...	2023-05-16 00:0...	1	6

Slika 51. Ispis SQL skripte *ProgStrucUsavView*

(Izvor: vlastita izrada)

1.9. Korisničko sučelje i korištenje aplikacije

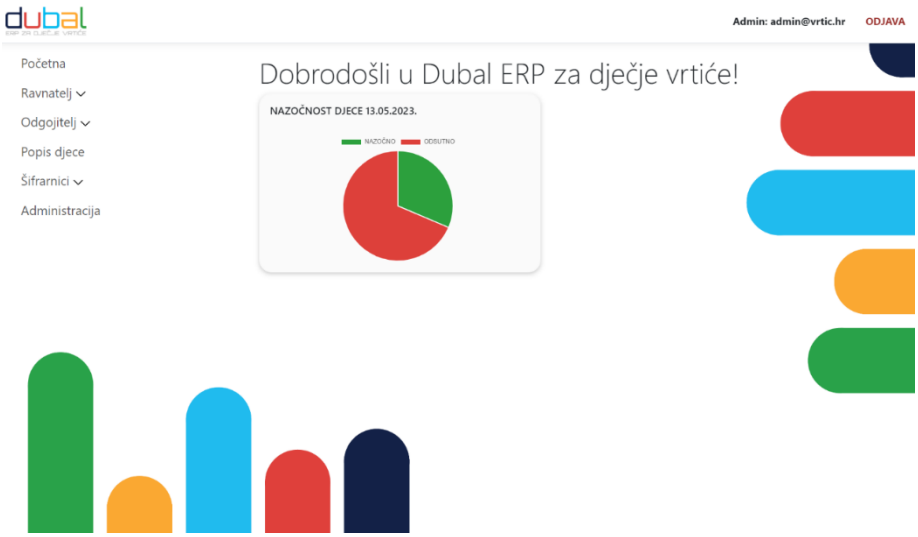
Prilikom pokretanja aplikacije, prikazuje se ekran za prijavu vidljiv na slici 52. Za ulazak u samu aplikaciju potrebno je upisati korisničko ime i lozinku. Ulaz u aplikaciju omogućen je trima različitim korisničkim ulogama: *Admin*, *Ravnatelj* i *Odgojitelj*.



Slika 52. Prijava u aplikaciju

(Izvor: vlastita izrada)

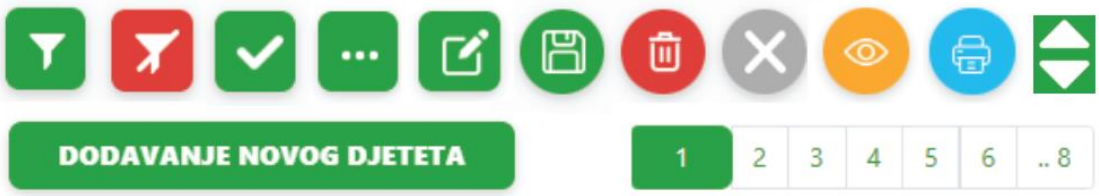
Nakon ulaza u aplikaciju prikazuje se početni ekran s informativnim grafom o broju nazočne i nenazočne djece trenutnog dana. U gornjem desnom kutu nalazi se informacija tko je trenutno prijavljen s kojom korisničkom ulogom te gumb za odjavu. Logo aplikacije u gornjem lijevom kutu otvara i zatvara izbornik koji se razlikuje s obzirom na trenutno prijavljenu korisničku ulogu. Slika 53. prikazuje početni ekran prijavljene administratorske korisničke uloge.



Slika 53. Početna stranica aplikacije

(Izvor: vlastita izrada)

Kroz cijelu aplikaciju protežu se neki isti elementi, poput gumbi, kako bi se zadržala konzistentnost i omogućilo lakše korištenje same aplikacije. Na slici 54. nalaze se svi elementi takvoga tipa. Prvi i drugi gumb označavaju postavljanje i uklanjanje filtera, treći gumb s „kvačicom“ označava odabir pojedinog reda neke tablice, četvrti gumb s „tri točkice“ označava otvaranje detalja pojedinog reda tablice, peti gumb s „olovkom“ označava otvaranje uređivanja reda tablice, šesti gumb s „disketom“ označava spremanje, sedmi gumb s „kantom za smeće“ označava brisanje, osmi gumb s „x-om“ označava zatvaranje forme ili odustajanje od neke radnje, deveti gumb s „okom“ označava pregled/otvaranje nekog dokumenta/slike, deseti gumb s „printerom“ označava kreiranje izvještaja u PDF obliku i njegov mogući ispis. Oznaka sa strelicama, koje pokazuju prema gore i dolje, označava sortiranje stupca tablice silaznim ili uzlaznim redoslijedom. Duguljasti gumb označava dodavanje, a natpisom na njemu jasno je o kakvom se dodavanju radi. Element s označenim brojkama predstavlja tzv. *pager*, tj. pomoćnik za kretanje po tablici s više strana prikaza.



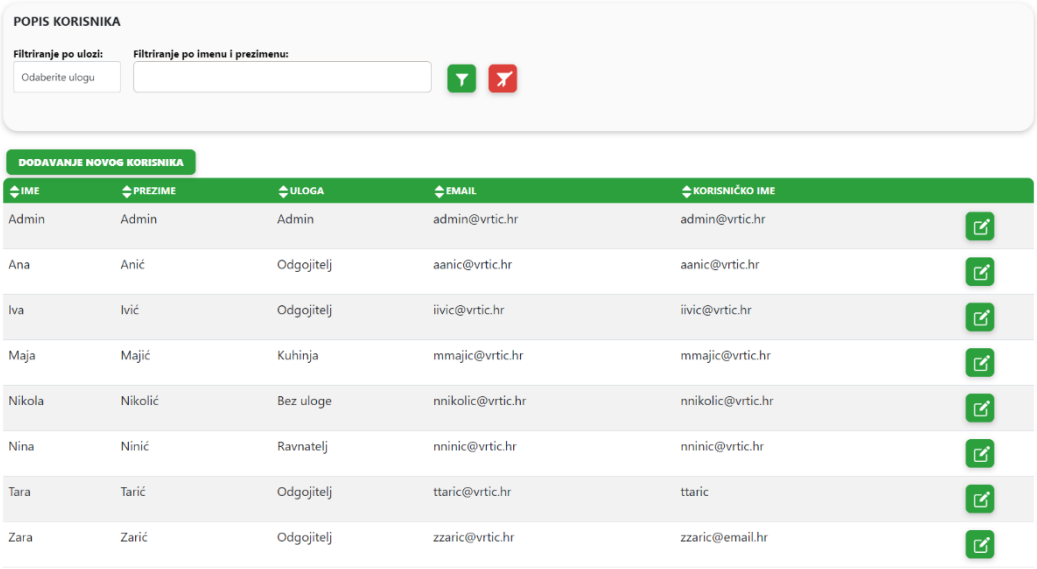
Slika 54. Gumbi
(Izvor: vlastita izrada)

1.9.1. Dio aplikacije za korisničku ulogu *Admin*

1.9.1.1. Upravljanje korisnicima aplikacije

Funkcionalnost upravljanja korisnicima sastoji se od dva različita ekrana: prikaz popisa korisnika (slika 55.) i dodavanje/uređivanje korisnika (slika 56.).

Ekran s popisom korisnika na vrhu ima filter za filtriranje prema korisničkoj ulozi (*select form element*) te prema imenu i/ili prezimenu (*input form element*). Ispod filtera nalazi se gumb za dodavanje novog korisnika te tablica s postojećim korisnicima. Prikazani atributi korisnika mogu se sortirati, a u svakom redu postoji gumb za uređivanje pripadajućeg korisnika.



Slika 55. Popis korisnika

(Izvor: vlastita izrada)

Pritiskom na gumb za dodavanje ili uređivanje pojavljuje se ekran s formom prikazanom na slici 56. Forma zadrži sedam *input* elemenata za upis imena, prezimena, email-a, korisničkog imena, lozinke, struke i zanimanja te jedan *select* element za korisničku ulogu. Forma se sprema pritiskom na gumb za spremanje, a zatvara s pritiskom na gumb za zatvaranje.

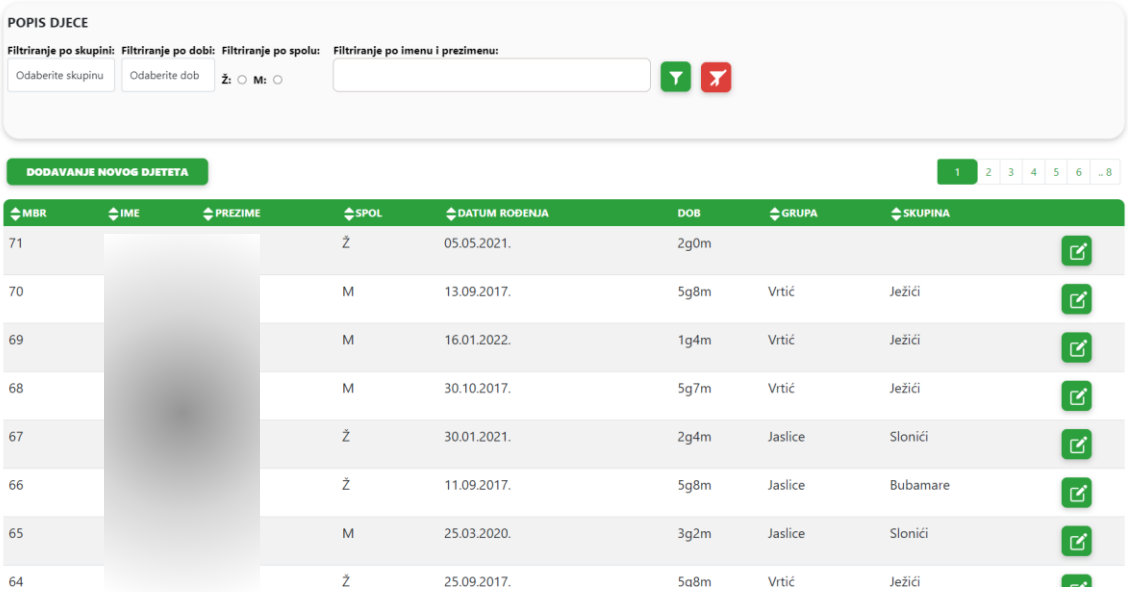
Slika 56. Dodavanje/uređivanje korisnika

(Izvor: vlastita izrada)

1.9.1.2. Upravljanje djecom

Funkcionalnost upravljanja djecom sastoji se od dva različita ekrana: prikaz popisa djece (slika 57.) i dodavanje/uređivanje djeteta (slika 58., slika 59. i slika 60.).

Ekran s popisom djece na vrhu ima filter za filtriranje prema skupini, dobi (*select form elementi*), prema spolu (*radio input form element*) te prema imenu i/ili prezimenu (*input form element*). Ispod filtera nalazi se gumb za dodavanje novog djeteta te tablica s postojećom djecom. Prikazani atributi djeteta mogu se sortirati, a u svakom redu postoji gumb za uređivanje pripadajućeg djeteta.

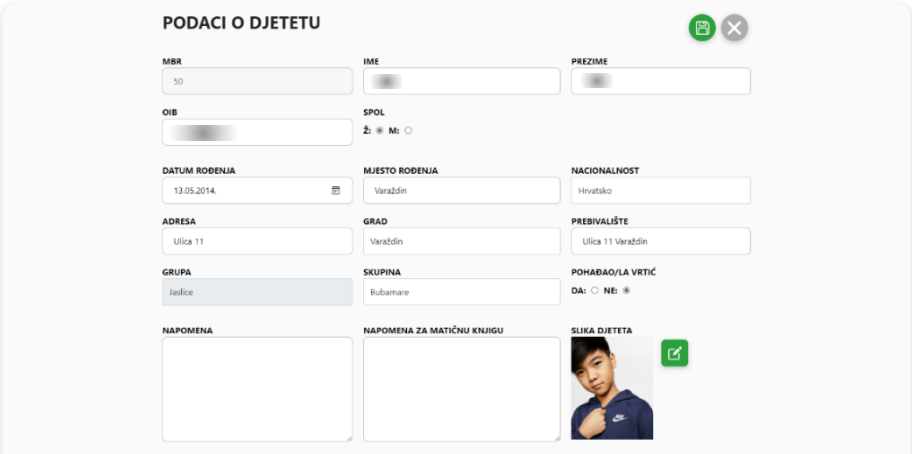


Slika 57. Popis djece

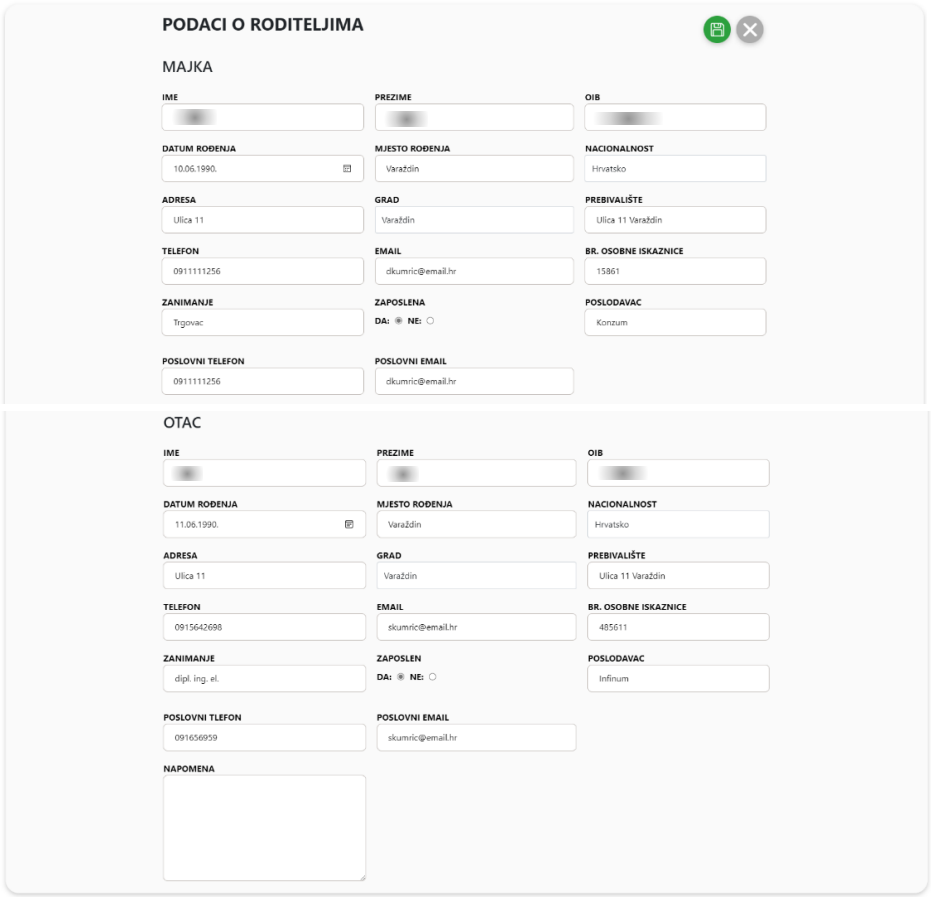
(Izvor: vlastita izrada)

Pritiskom na gumb za dodavanje ili uređivanje pojavljuje se ekran s formom prikazanom na slici 58., slici 59. i slici 60. Forma sadrži tri dijela (*Podaci o djetetu, Podaci o roditeljima* i *Podaci o zdravlju*). Dio *Podaci o djetetu* ima sedam *input* elemenata za upis imena, prezimena, OIB-a, datuma rođenja (*type date*), mjesta rođenja, adrese i prebivališta, tri *select* elementa za nacionalnost, grad i skupinu, dva *radio* elementa za spol i informaciju je li dijete već prije pohađalo vrtić, dva *textarea* elementa za općenitu napomenu i napomenu specijaliziranu za matičnu knjigu, jedan *input* element tipa *file* za unos slike djeteta, dva onemogućena (*disabled*) elementa koji se popunjavaju automatski za matični broj (MBR) i grupu skupine. Slika se može promijeniti klikom na gumb za uređivanje pokraj same slike. Dio *Podaci o roditeljima* ima dva podjednaka dijela za majku i oca koji sadrže 14 *input* elemenata za upis imena, prezimena, OIB-a, datuma rođenja (*type date*), mjesta rođenja, adrese, prebivališta, telefona, email-a, broja osobne iskaznice, zanimanja, poslodavca, poslovnog telefona i poslovnog email-a, dva *select* elementa za nacionalnost i grad, jedan *radio* element za informaciju je li roditelj

zaposlen, jedan *textarea* element za napomenu vezanu uz oba roditelja. Dio *Podaci o zdravlju* ima tri *input* elementa za upis MBO-a, imena i prezimena pedijatra te njegovog telefona, jedan *textarea* element za upis mogućih poteškoća djeteta. Forma se sprema pritiskom na gumb za spremanje, a zatvara s pritiskom na gumb za zatvaranje.



Slika 58. Dodavanje/uređivanje djeteta - *Podaci o djetetu*
(Izvor: vlastita izrada)



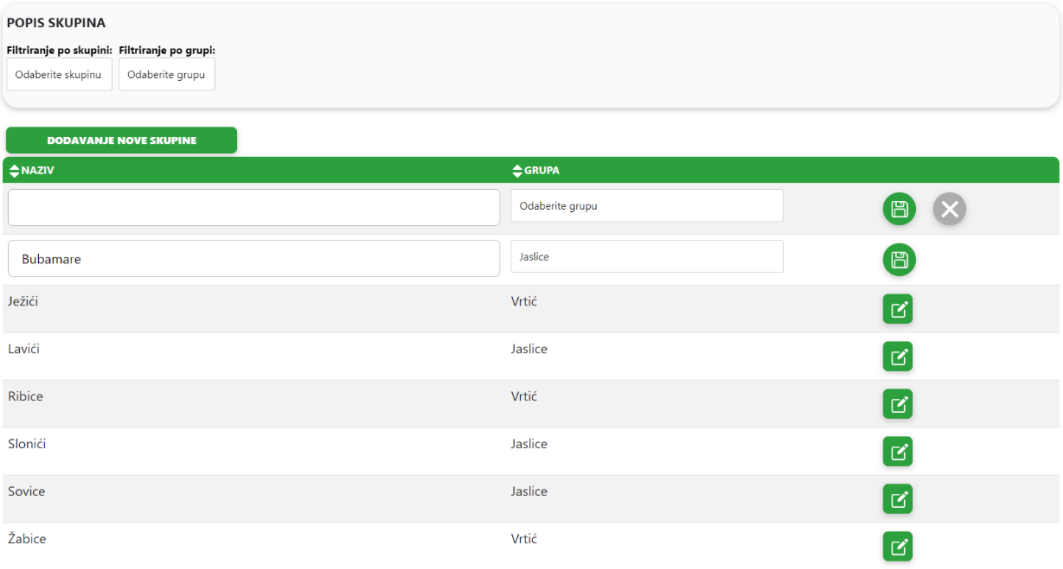
Slika 59. Dodavanje/uređivanje djeteta - *Podaci o roditeljima*
(Izvor: vlastita izrada)

Slika 60. Dodavanje/uređivanje djeteta - *Podaci o zdravlju*

(Izvor: vlastita izrada)

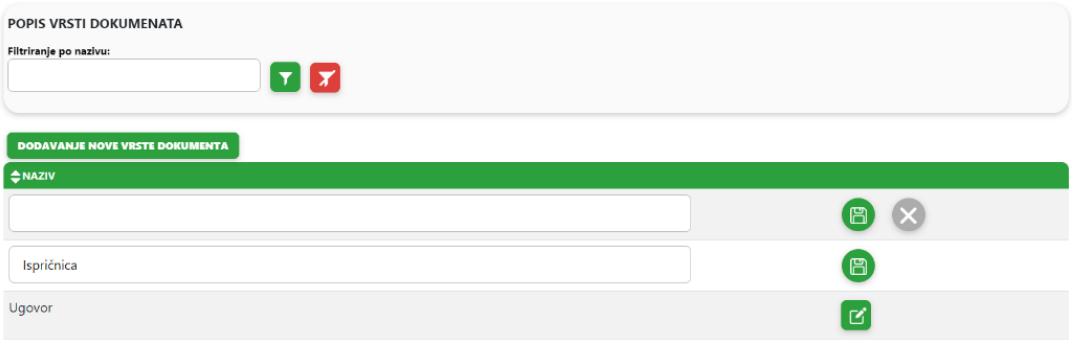
1.9.1.3. Upravljanje šifrarnicima

U izborniku pod opcijom *Šifrarnici* ponuđena su šest šifrarnika *Skupine, Vrste dokumenata*, *Nacionalnosti*, *Neradni dani* i *Gradovi*. Svaki od šifrarnika se sastoji od jednog ekrana na kojem je moguće filtrirati, sortirati i pregledavati popise postojećih stavaka šifrarnika, uređivati ih te dodavati nove. Spomenuti ekrani vidljivi su na slikama 61., 62., 63., 64. i 65. Pošto se radi o samo jednom ili dva atributa pojedine stavke, dodavanje i uređivanje omogućeno je unutar same tablice popisa. Pritiskom na gumb dodavanja, dodaje se redak tablice na njen sami vrh te se omogućuje upis atributa u *input* elemente forme (i *select* element za odabir grupe ili skupine kod *Skupina*). Pritiskom na gumb spremanja, forma se sprema, a pritiskom na gumb za zatvaranje, odustaje se od dodavanja. Isto tako pritiskom na gumb za uređivanje, atributi izabrane stavke se pretvaraju u *editabilne* elemente forme, a sama forma se sprema pritiskom na gumb za spremanje.



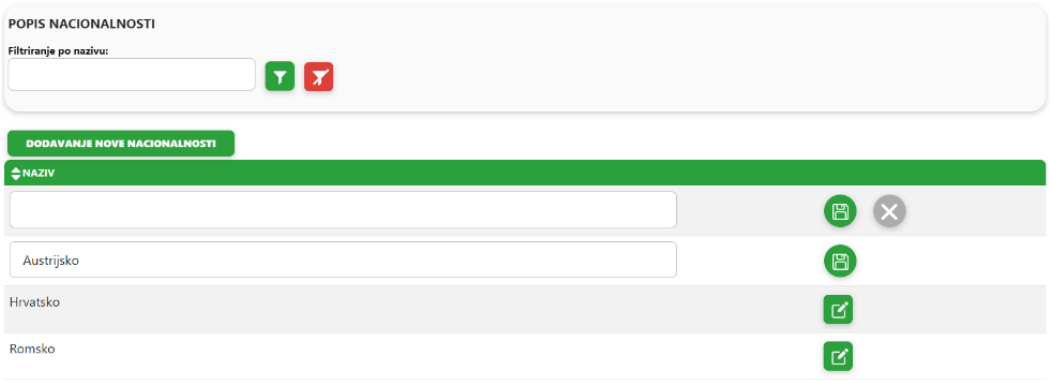
Slika 61. Upravljanje skupinama

(Izvor: vlastita izrada)



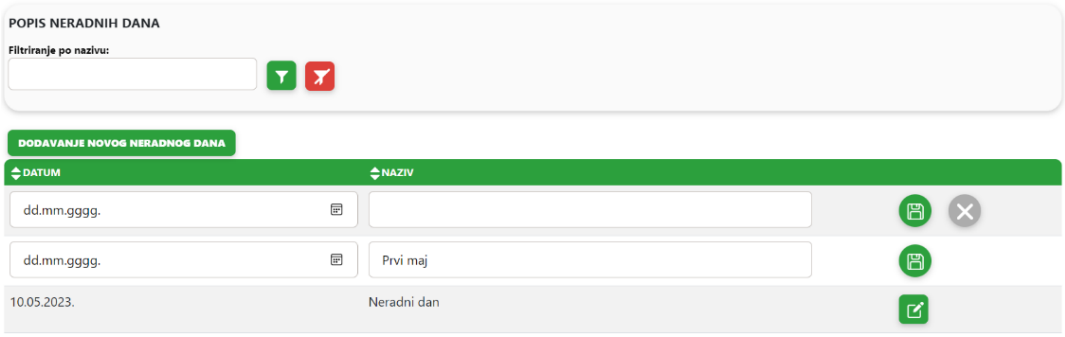
Slika 62. Upravljanje vrstama dokumenata

(Izvor: vlastita izrada)



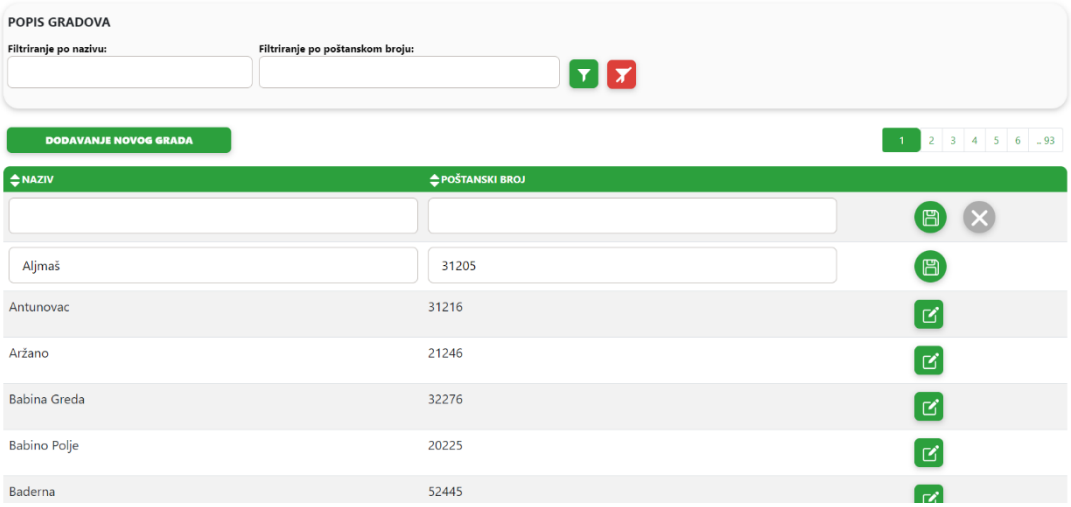
Slika 63. Upravljanje nacionalnostima

(Izvor: vlastita izrada)



Slika 64. Upravljanje neradnim danima

(Izvor: vlastita izrada)



Slika 65. Upravljanje gradovima
(Izvor: vlastita izrada)

Šifrarnik za pedagoške godine također se nalazi unutar opcije *Šifrarnici*, ali funkcionira na nešto drugačiji način od ostalih. Prikaz ekrana popisa pedagoških godina nalazi se na slici 66. Dodavanje nove pedagoške godine moguće je samo ako već ne postoji pedagoška godina za sljedeću godinu. Ako sljedeća godina još nije upisana, pritiskom na gumb za dodavanje dodaje se prva sljedeća pedagoška godina s mogućnošću postavljanja iste na aktualnu. Isto tako, ako se neka od pedagoških godina želi urediti, omogućeno je samo postavljanje iste na aktualnu godinu. U slučaju postavljanja pedagoške godine na aktualnu, do tad aktualna godina gubi taj status. Pritiskom na gumb spremanja, forma se sprema.

Slika 66. Upravljanje pedagoškim godinama
(Izvor: vlastita izrada)

1.9.2. Dio aplikacije za korisničku ulogu *Ravnatelj*

Korisnička uloga *Ravnatelj* nakon prijave u aplikaciju, za razliku od *Admina,* ima nešto manji izbornik prikazan na slici 67.

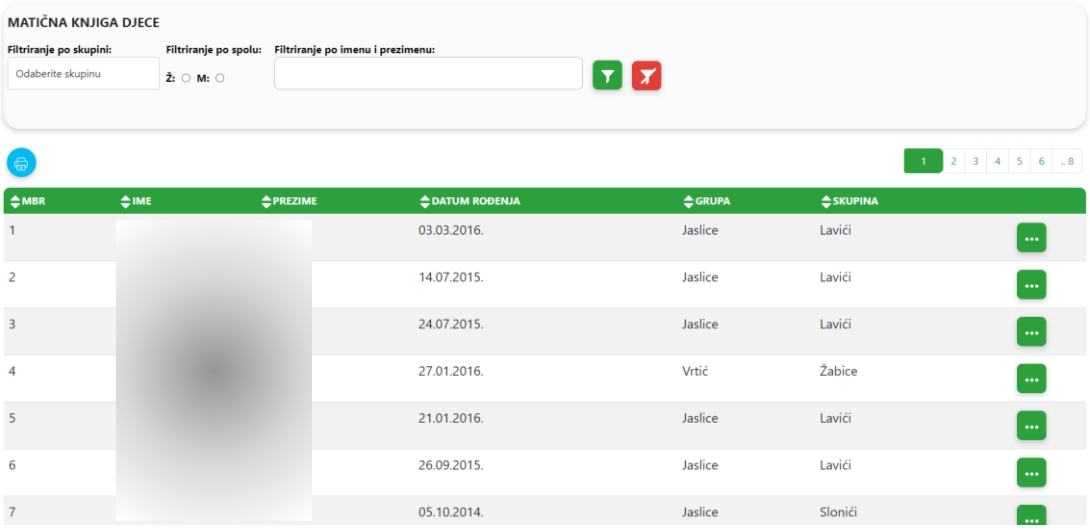
Slika 67. Izbornik za korisničku ulogu *Ravnatelj*

(Izvor: vlastita izrada)

1.9.2.1. Matična knjiga djece

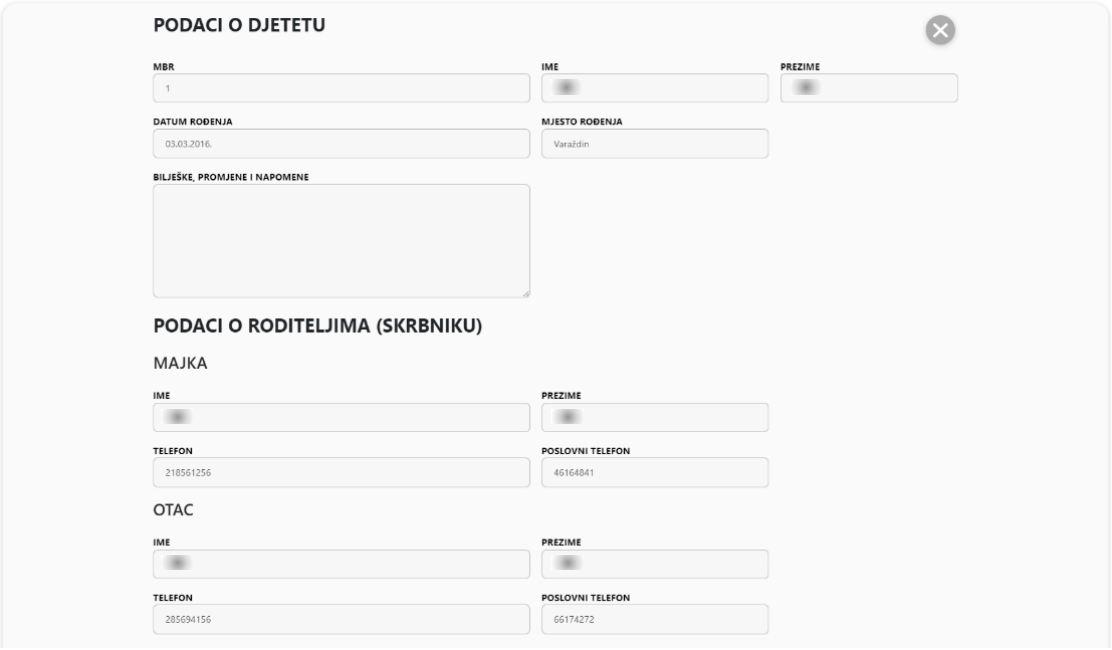
Funkcionalnost matične knjige djece sastoji se od dva različita ekrana: prikaz popisa djece matične knjige (slika 68.) i detaljni pregled podataka u matičnoj knjizi za pojedino dijete (slika 69.).

Ekran popisa djece u matičnoj knjizi na vrhu ima filter za filtriranje prema skupini (*select form element*), spolu (*radio form element*) te prema imenu i/ili prezimenu (*input form element*). Ispod filtera nalazi se gumb za ispis matične knjige vidljiv na slici 70. te tablica s djecom matične knjige. Prikazani atributi djeteta mogu se sortirati, a u svakom redu postoji gumb za detaljan pregled informacija o djetetu.



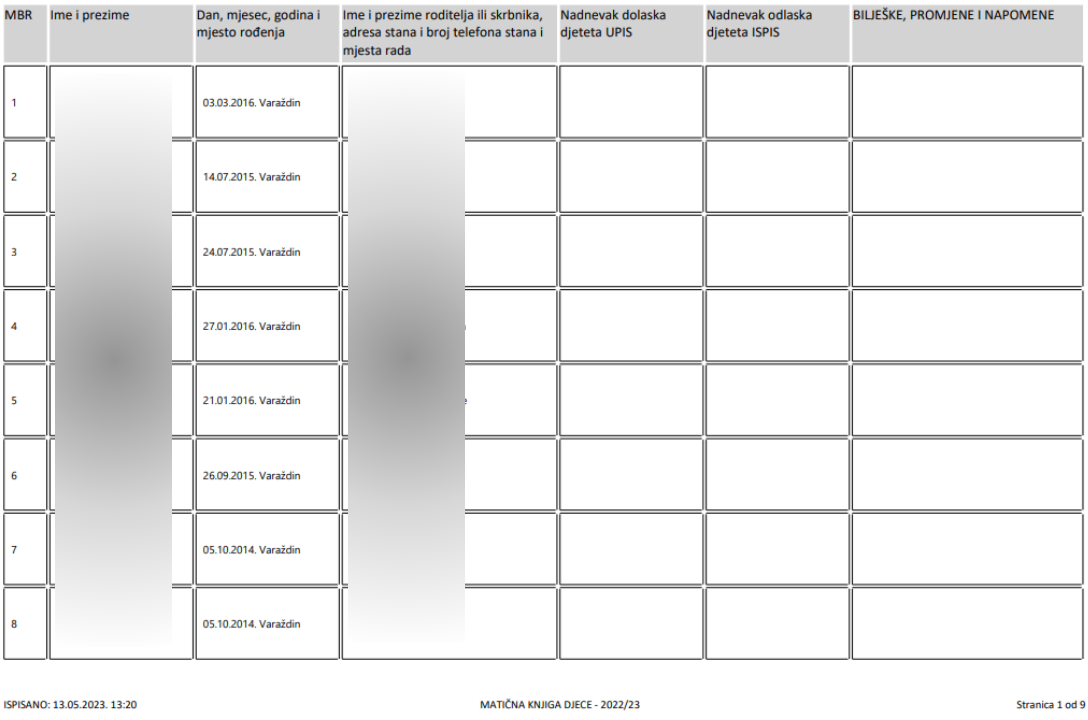
Slika 68. Matična knjiga djece

(Izvor: vlastita izrada)



Slika 69. Detalji matične knjige za dijete

(Izvor: vlastita izrada)



MBR	Ime i prezime	Dan, mjesec, godina i mjesto rođenja	Ime i prezime roditelja ili skrbnika, adresa stana i broj telefona stana i mjesta rada	Nadnevak dolaska djeteta UPIS	Nadnevak odlaska djeteta ISPIS	BILJEŠKE, PROMJENE I NAPOMENE
1		03.03.2016. Varaždin				
2		14.07.2015. Varaždin				
3		24.07.2015. Varaždin				
4		27.01.2016. Varaždin				
5		21.01.2016. Varaždin				
6		26.09.2015. Varaždin				
7		05.10.2014. Varaždin				
8		05.10.2014. Varaždin				

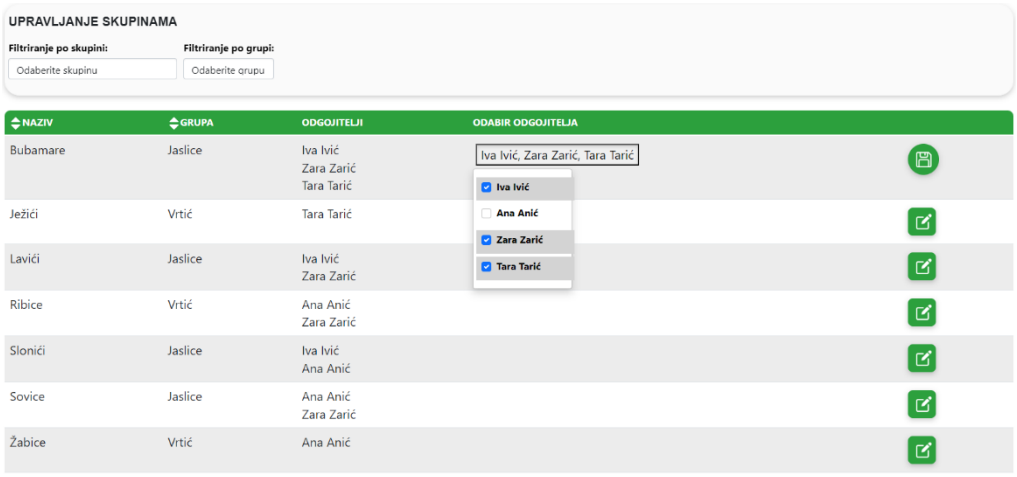
ISPISANO: 13.05.2023. 13:20 MATIČNA KNJIGA DJECE - 2022/23 Stranica 1 od 9

Slika 70. Ispis matične knjige

(Izvor: vlastita izrada)

1.9.2.2. Upravljanje skupinama djece

Funkcionalnost upravljanja skupinama djece sastoji se od jednog ekrana na kojem je moguće filtrirati, sortirati i pregledavati popis skupina s njima dodijeljenim *Odgojiteljima*. Također, omogućeno je i dodavanje ili uklanjanje *Odgojitelja* iz pojedine skupine (*multiselect form element*). Pritiskom na gumb za uređivanje otvara se polje *Odabir odgojitelja* gdje se iz *multiselect* elementa mogu označiti ili odznačiti željeni *Odgojitelji*. Pritiskom na gumb spremanja, forma se sprema. Spomenuti ekran vidljiv je na slici 71.



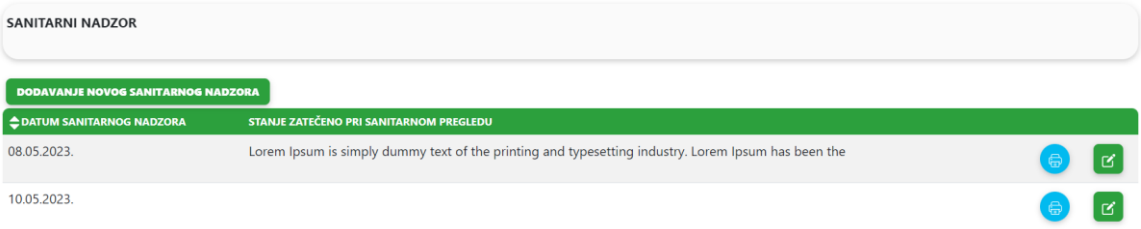
Slika 71. Upravljanje skupinama djece

(Izvor: vlastita izrada)

1.9.2.3. Evidencija Sanitarnog nadzora

Funkcionalnost evidencije *Sanitarnog nadzora* sastoji se od dva različita ekrana: prikaz popisa sanitarnih nadzora (slika 72.) i dodavanje/uređivanje sanitarnog nadzora (slika 74.).

Ekran popisa sanitarnih nadzora ima gumb za dodavanje novog sanitarnog nadzora te tablicu s postojećim sanitarnim nadzorima. Prikazani atribut datuma može se sortirati, a u svakom redu postoje dva gumba za ispis (prikazan na slici 73.) i za uređivanje pripadajućeg sanitarnog nadzora.



Slika 72. Popis sanitarnih nadzora

(Izvor: vlastita izrada)

EVIDENCIJA SANITARNOG NADZORA

Datum:	08.05.2023.
Stanje zatečeno pri sanitarnom pregledu:	Lorem Ipsum is simply dummy text of the printing and typesetting industry. Lorem Ipsum has been the industry's standard dummy text ever since the 1500s, when an unknown printer took a galley of type and scrambled it to make a type specimen book.
Izrečene mjere:	Lorem Ipsum is simply dummy text of the printing and typesetting industry. Lorem Ipsum has been the industry's standard dummy text ever since the 1500s, when an unknown printer
Provedene mjere:	Lorem Ipsum is simply dummy text of the printing and typesetting industry. Lorem Ipsum has been the industry's standard dummy text ever since the 1500s, when an unknown printer took a galley of
Napomena sanitarnog nadzora:	Lorem Ipsum is simply dummy text of the printing and typesetting industry. Lorem Ipsum has been the industry's standard dummy text ever since the 1500s, when an unknown printer took a galley of

Bikovec, 08.05.2023 godine.

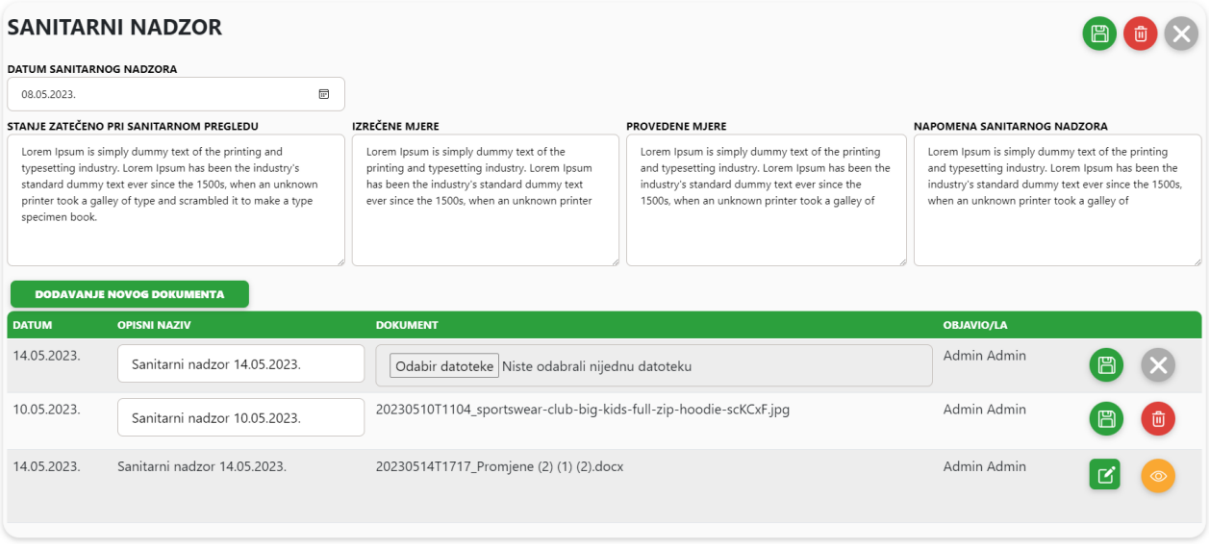
RAVNATELJ:

_____ M.P.

Nina Ninić

Slika 73. Ispis sanitarnog nadzora
(Izvor: vlastita izrada)

Pritiskom na gumb za dodavanje ili uređivanje pojavljuje se ekran s formom prikazanom na slici 74. Forma ima pet *input* elementa za upis datuma sanitarnog nadzora (*type date*), stanja zatečenog pri sanitarnom pregledu, izrečenih mjera, provedenih mjera i napomene sanitarnog nadzora. Forma se sprema pritiskom na gumb za spremanje, briše pritiskom na gumb za brisanje, a zatvara s pritiskom na gumb za zatvaranje. Ispod spomenutih *input* elemenata forme nalazi se gumb za dodavanje mogućih dokumenata vezanih za određeni sanitarni nadzor i tablica već dodanih dokumenata. Na slici 74. prvi red tablice prikazuje redak koji se pojavljuje nakon pritiska na gumb za dodavanje, sadrži datum dodavanja dokumenta (dodaje se automatski prema trenutnom datumu), opisni naziv (*input form element*) te sami dokument (*input form element* tipa *file*). Gumb spremanja sprema dodani dokument, a gumb zatvaranja, poništava dodavanje dokumenta te dodani red nestaje iz tablice. Drugi red tablice prikazuje redak koji se pojavljuje nakon pritiska na gumb za uređivanje postojećeg dokumenta (gumb vidljiv u trećem retku), sadrži datum dodavanja dokumenta i sam dokument koji se ne mogu uređivati te opisni naziv (*input form element*) koji postaje *editabilan*. Gumb spremanja sprema uređeni dokument, a gumb brisanja ga briše te uređivani red nestaje iz tablice. Treći red tablice prikazuje kako izgleda dokument kad je već dodan i nije u fazi uređivanja. Sadrži gumb za uređivanje koji red pretvara u oblik prikazan u drugom redu te gumb za prikaz koji u novom prozoru preglednika otvara konkretni dokument za pregled.

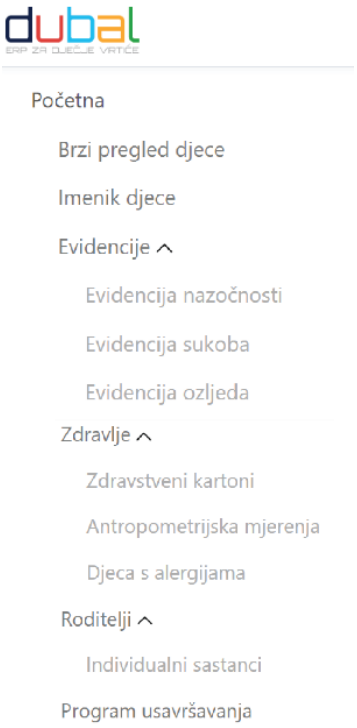


Slika 74. Dodavanje/uređivanje sanitarnih nadzora

(Izvor: vlastita izrada)

1.9.3. Dio aplikacije za korisničku ulogu *Odgojitelj*

Korisnička uloga *Odgojitelj* nakon prijave u aplikaciju ima izbornik prikazan na slici 75.



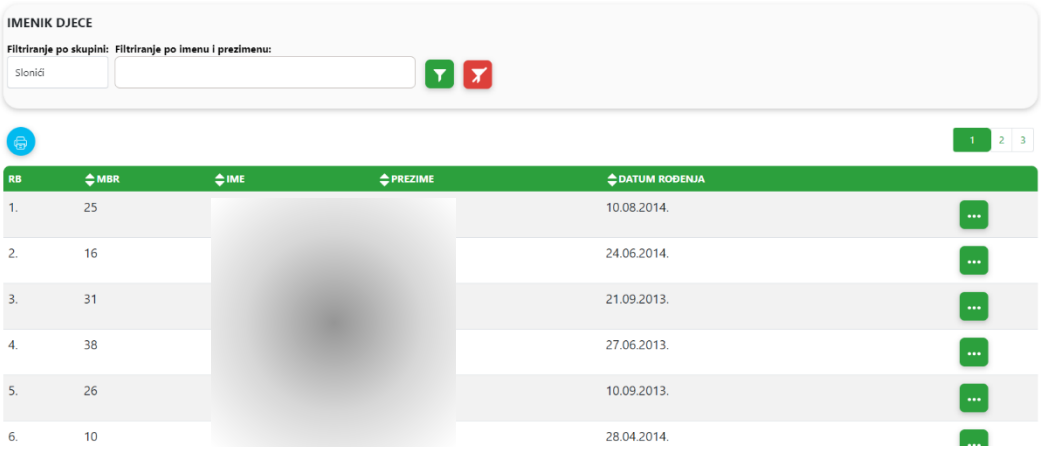
Slika 75. Izbornik za korisničku ulogu *Odgojitelj*

(Izvor: vlastita izrada)

1.9.3.1. Imenik djece

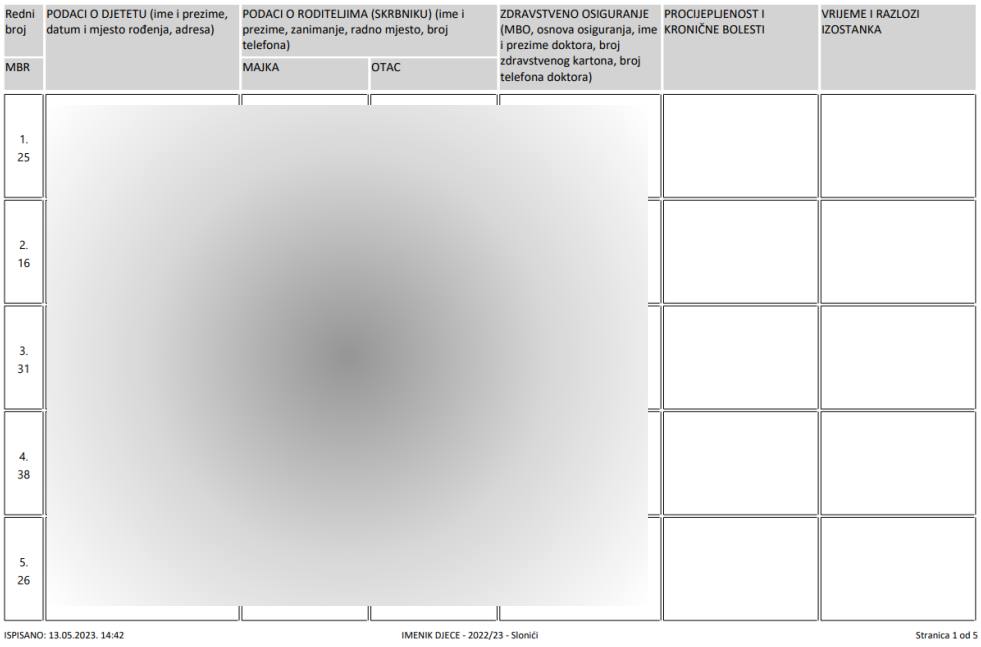
Funkcionalnost imenika djece sastoji se od dva različita ekrana: prikaz popisa djece imenika (slika 76.) i detaljni pregled podataka u imeniku za pojedino dijete (slika 78.).

Ekran popisa djece u imeniku djece na vrhu ima filter za filtriranje prema skupini (*select form element*) te prema imenu i/ili prezimenu (*input form element*). Ispod filtera nalazi se gumb za ispis imenika vidljiv na slici 77. te tablica s djecom u imeniku. Prikazani atributi djeteta mogu se sortirati, a u svakom redu postoji gumb za detaljan pregled informacija o djetetu.



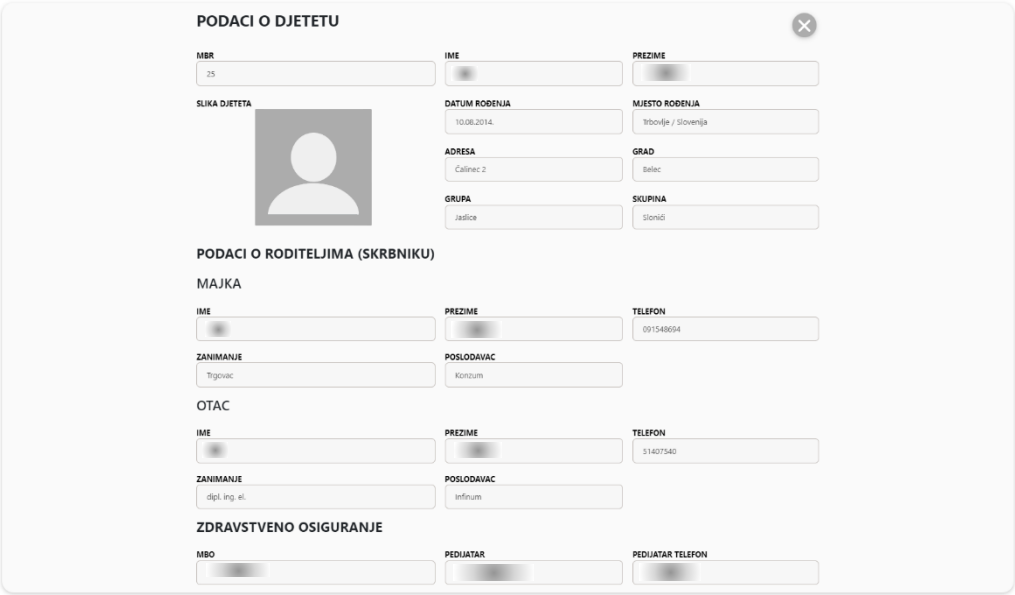
Slika 76. Imenik djece

(Izvor: vlastita izrada)



Slika 77. Ispis imenika djece

(Izvor: vlastita izrada)



Slika 78. Detalji imenika djece za dijete

(Izvor: vlastita izrada)

1.9.3.2. Evidencija nazočnosti djece

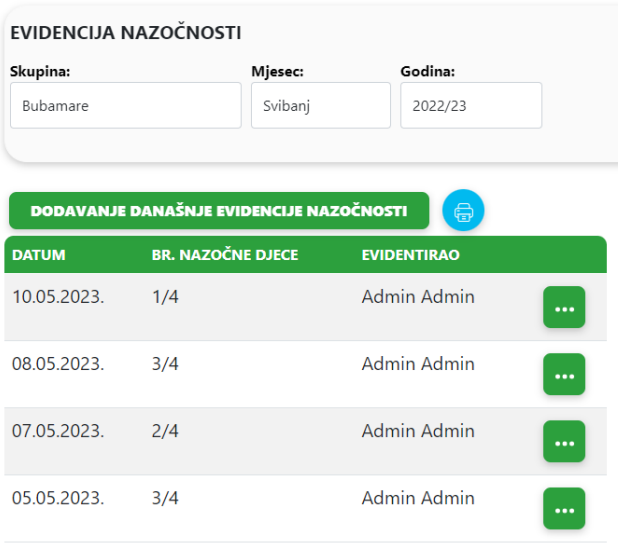
Funkcionalnost evidencije nazočnosti djece sastoji se od jednog ekrana koji na vrhu ima filter za filtriranje prema skupini, mjesecu i pedagoškoj godini (*select form elementi*). Ispod filtera nalazi se gumb za dodavanje vidljiv samo ako izabrana skupina još nema otvorenu evidenciju za trenutni dan. Pored njega se nalazi gumb za ispis mjesečne evidencije nazočnosti za izabranu skupinu čiji ispis je vidljiv na slici 79. Ispod spomenutih gumbi nalazi se tablica s do sad provedenim evidencijama. Opisani slučaj ekrana vidljiv je na slici 80.

Skupina: Bubamare Odgojitelji: Iva Ivić Zara Zarić Tara Tarić Mjesec/Godina: 5./2023. Pedagoška godina: 2022/23

IME I PREZIME DJETETA:	1.	2.	3.	4.	5.	6.	7.	8.	9.	10.	11.	12.	13.	14.	15.	16.	17.	18.	19.	20.	21.	22.	23.	24.	25.	26.	27.	28.	29.	30.	31.	Uk.
1.					-		+	+		-			-																			2
2.					+		-	+		-			+																			3
3.					+		-	-		+			-																			2
4.					+		+	+		-			-																			3
UKUPAN BROJ DJECE	0	0	0	0	3	0	2	3	0	1	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	

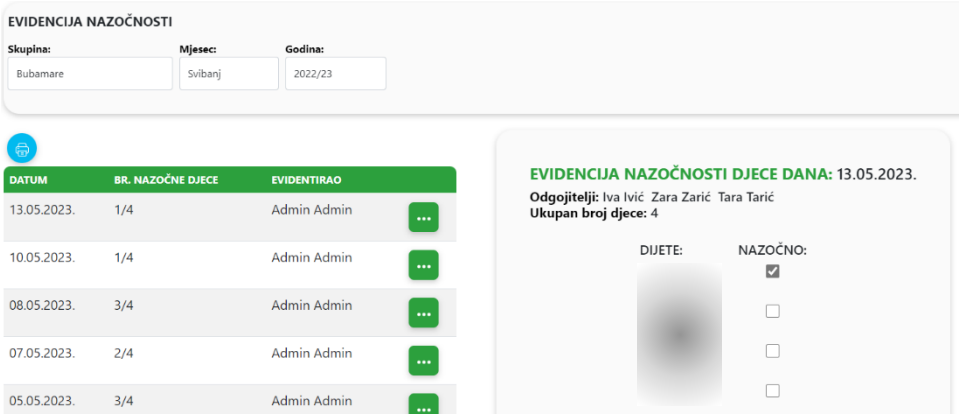
Slika 79. Ispis evidencije nazočnosti djece

(Izvor: vlastita izrada)



Slika 80. Popis evidencija nazočnosti djece - Slučaj bez evidencije trenutnog dana (Izvor: vlastita izrada)

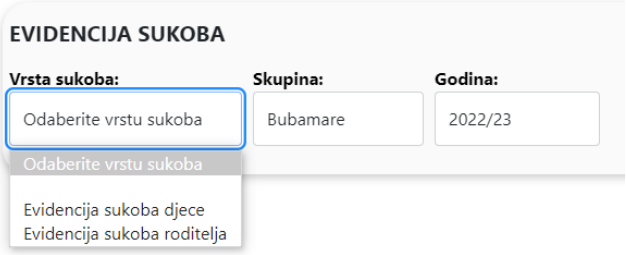
Nakon pritiska na gumb za dodavanje nove evidencije nazočnosti, isti gumb nestaje, a desno od tablice, već provedenih evidencija, pojavljuje se forma s popisom djece izabrane skupine. Pored svakog djeteta nalazi se *checkbox* element za označavanje je li dijete prisutno. Forma se sprema nakon promjene stanja svakog *checkbox* elementa (*onChange*). Prikaz opisanog slučaja nalazi se na slici 81. Ako se žele pregledavati stare evidencije, pritisne se na gumb za detalje pored određene evidencije iz tablice popisa provedenih evidencija. Nakon pritiska na spomenuti gumb, stara evidencija se prikazuje na mjestu forme današnje evidencije. Ako je korisnik u korisničkoj ulozi *Odgojitelj*, tada stare evidencije može samo pregledavati te su *checkbox* elementi onemogućeni za promjene (*disabled*), a ako se radi o korisniku administartorske uloge, tada su oni omogućeni za promjene.



Slika 81. Popis evidencija nazočnosti djece - Slučaj s evidencijom trenutnog dana (Izvor: vlastita izrada)

1.9.3.3. Evidencija sukoba

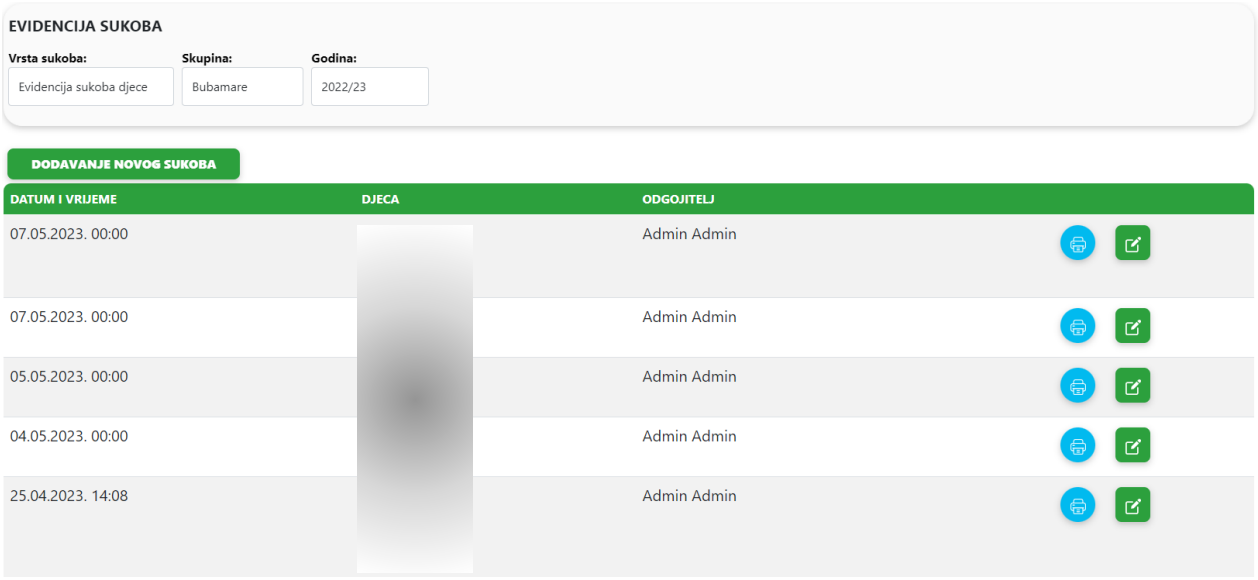
Otvaranjem funkcionalnosti evidencije sukoba pojavljuje se filter za odabir vrste sukoba (među djecom ili među roditeljima) te za skupinu i pedagošku godinu (*select form elementi*). Spomenuti filter vidljiv je na slici 82.



Slika 82. Izbor vrste sukoba za pregled evidencija sukoba

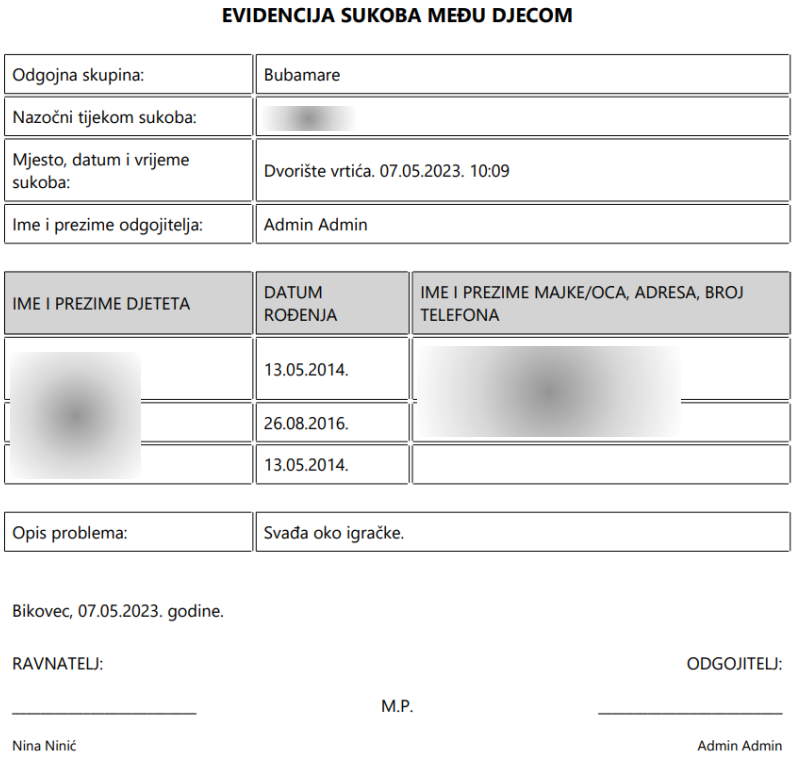
(Izvor: vlastita izrada)

Odabirom evidencije sukoba među djecom pojavljuje se ekran vidljiv na slici 83. na vrhu kojeg se nalazi već opisani filter. Ispod filtera nalazi se gumb za dodavanje nove evidencije sukoba među djecom te tablica s postojećim sukobima. U svakom redu tablice postoje dva gumba za ispis (prikazan na slici 84.) i za uređivanje pripadajućeg sukoba.



Slika 83. Pregled sukoba među djecom

(Izvor: vlastita izrada)



EVIDENCIJA SUKOBA MEĐU DJECOM

Odgojna skupina:	Bubamare
Nazočni tijekom sukoba:	[redacted]
Mjesto, datum i vrijeme sukoba:	Dvorište vrtića. 07.05.2023. 10:09
Ime i prezime odgojitelja:	Admin Admin

IME I PREZIME DJETETA	DATUM ROĐENJA	IME I PREZIME MAJKE/OCA, ADRESA, BROJ TELEFONA
[redacted]	13.05.2014.	[redacted]
	26.08.2016.	
	13.05.2014.	

Opis problema:	Svađa oko igračke.

Bikovec, 07.05.2023. godine.

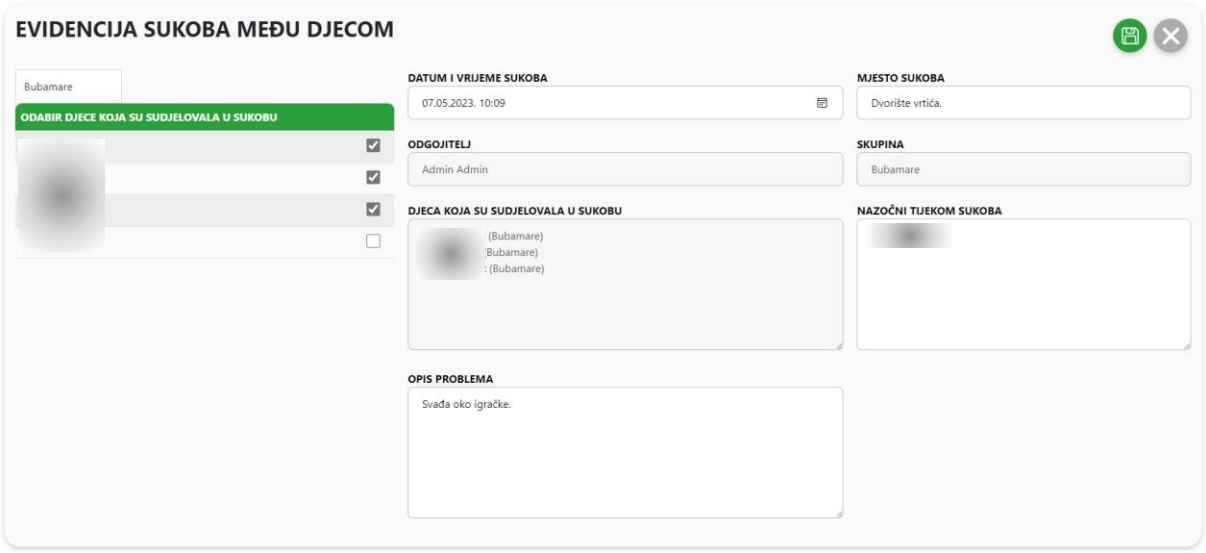
RAVNATELJ: ODGOJITELJ:

M.P.

Nina Ninić Admin Admin

Slika 84. Ispis evidencije sukoba među djecom
(Izvor: vlastita izrada)

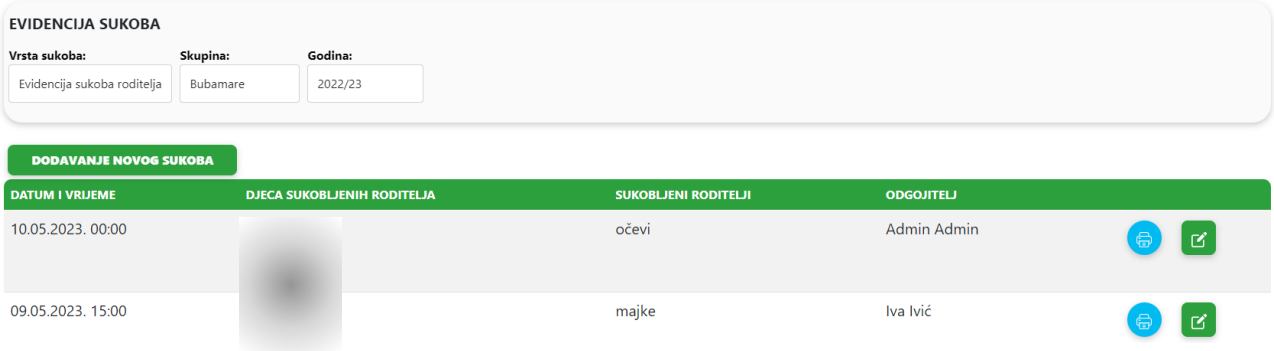
Pritiskom na gumb za dodavanje ili uređivanje pojavljuje se ekran s formom prikazanom na slici 85. Forma ima tablicu s nabrojanom djecom grupiranom po skupinama (iznad nje se nalazi filter za skupinu u obliku *select form* elementa). Pored svakog djeteta nalazi se *checkbox* element za odabir djeteta ako je sudjelovao u sukobu. Nakon što se dijete izabere, njegovo ime, prezime i skupina kojoj pripada pojavljuju se u onemogućenom (*disabled*) *textarea form* elementu za djecu koja su sudjelovala u sukobu. Osim spomenutog, forma ima dva *input* elementa za upis datuma (*type date*) i mjesta sukoba, dva *textarea* elementa za upis nazočnih tijekom sukoba i opis nastalog problema te dva onemogućena *input* elementa koji se automatski popunjavaju ovisno o prijavljenom korisniku i skupini koju je korisnik izabrao u filteru na prethodnom ekranu. Forma se sprema pritiskom na gumb za spremanje, briše pritiskom na gumb za brisanje koji je vidljiv tek ako su odznačena sva djeca koja su sudjelovala u sukobu, a zatvara s pritiskom na gumb za zatvaranje.



Slika 85. Dodavanje/uređivanje evidencije sukoba među djecom

(Izvor: vlastita izrada)

Odabirom evidencije sukoba među roditeljima pojavljuje se sličan ekran vidljiv na slici 86. na vrhu kojeg se nalazi već opisani filter. Ispod filtera nalazi se gumb za dodavanje nove evidencije sukoba među roditeljima te tablica s postojećim sukobima. U svakom redu tablice postoje dva gumba za ispis (prikazan na slici 87.) i za uređivanje pripadajućeg sukoba.



Slika 86. Pregled sukoba među roditeljima

(Izvor: vlastita izrada)

EVIDENCIJA SUKOBA MEĐU RODITELJIMA

Odgojna skupina:	Bubamare
Roditelji koji su sudjelovali u sukobu:	očevi
Mjesto, datum i vrijeme sukoba:	10.05.2023. 00:00
Ime i prezime odgojitelja:	Admin Admin

IME I PREZIME DJETETA	DATUM ROĐENJA	IME I PREZIME MAJKE/OCA, ADRESA, BROJ TELEFONA
[redacted]	26.08.2016.	[redacted]
[redacted]	13.05.2014.	
	13.05.2014.	

Opis problema:	sdfsf

Bikovec, 10.05.2023. godine.

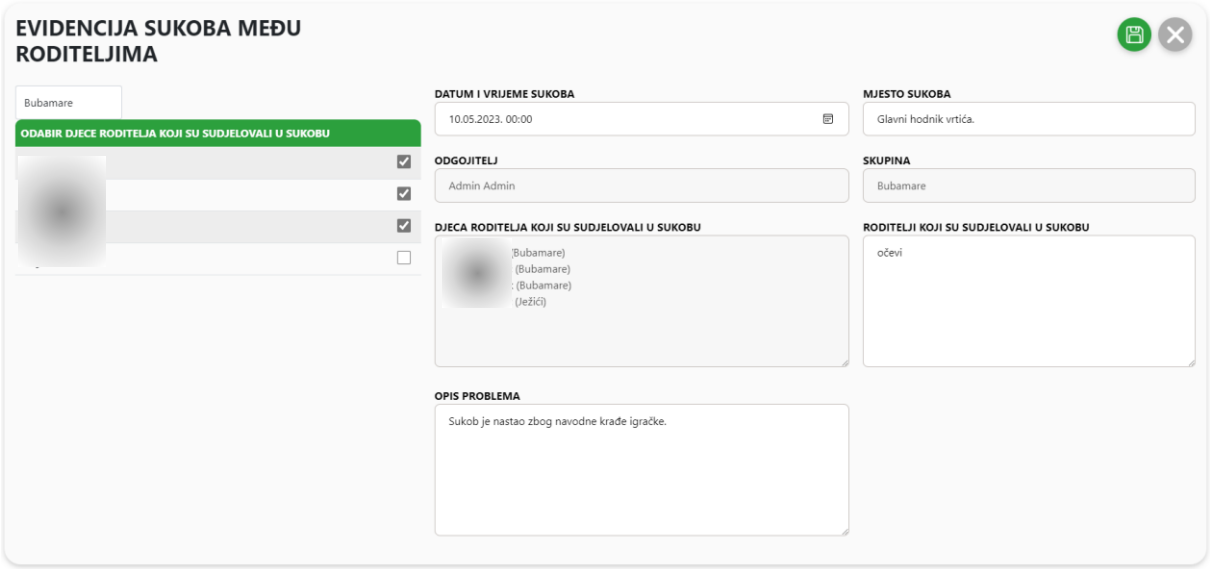
RAVNATELJ: ODGOJITELJ:

M.P.

Nina Ninić Admin Admin

Slika 87. Ispis evidencije sukoba među roditeljima
(Izvor: vlastita izrada)

Pritiskom na gumb za dodavanje ili uređivanje pojavljuje se ekran s formom prikazanom na slici 88. Forma ima tablicu s nabrojanom djecom grupiranom po skupinama (iznad nje se nalazi filter za skupinu u obliku *select form* elementa). Pored svakog djeteta nalazi se *checkbox* element za odabir djeteta ako je njegov roditelj sudjelovao u sukobu. Nakon što se izabere dijete, njegovo ime, prezime i skupina kojoj pripada pojavljuju se u onemogućenom (*disabled*) *textarea form* elementu za djecu čiji su roditelji sudjelovali u sukobu. Osim spomenutog, forma ima dva *input* elementa za upis datuma (*type date*) i mjesta sukoba, dva *textarea* elementa za upis roditelja koji su sudjelovali u sukobu i opis nastalog problema te dva onemogućena *input* elementa koji se automatski popunjavaju ovisno o prijavljenom korisniku i skupini koju je korisnik izabrao u filteru na prethodnom ekranu. Forma se sprema pritiskom na gumb za spremanje, briše pritiskom na gumb za brisanje, koji je vidljiv tek kad se sva djeca čiji su roditelji sudjelovali u sukobu odznače, a zatvara s pritiskom na gumb za zatvaranje.



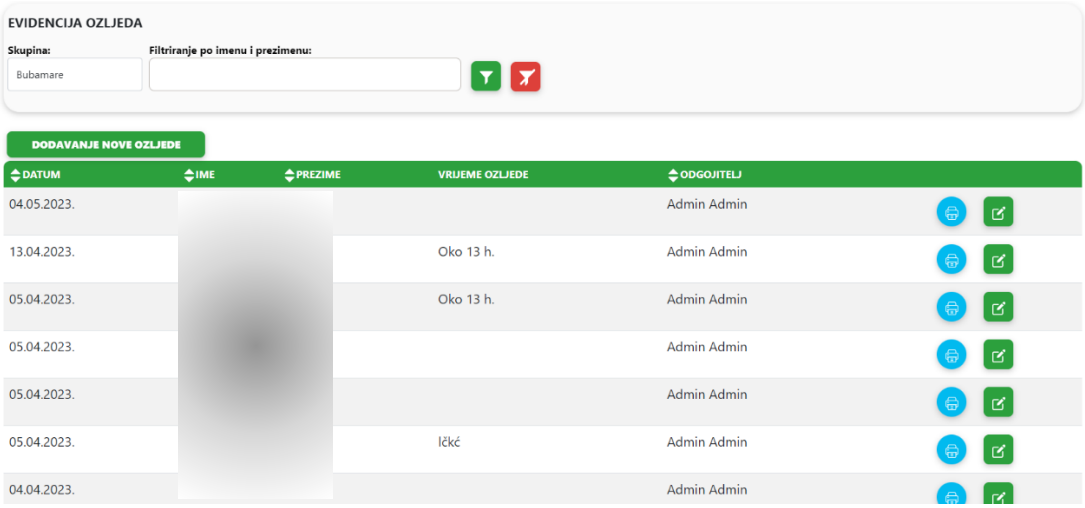
Slika 88. Dodavanje/uređivanje evidencije sukoba među roditeljima

(Izvor: vlastita izrada)

1.9.3.4. Evidencija ozljeda djece

Funkcionalnost evidencije ozljeda djece sastoji se od tri različita ekrana: prikaz popisa ozljeda (slika 89.), dodavanje ozljede (slika 91. i slika 92.) i uređivanje ozljede (slika 93.).

Ekran evidencija ozljeda djece na vrhu ima filter za filtriranje prema skupini (*select form element*) te prema imenu i/ili prezimenu (*input form element*). Ispod filtera nalazi se gumb za dodavanje nove ozljede te tablica s postojećim ozljedama djece. Prikazani atributi ozljede mogu se sortirati, a u svakom redu postoje dva gumba za ispis evidencije (prikazan na slici 90.) i za uređivanje pripadajuće ozljede.



Slika 89. Pregled ozljeda djece

(Izvor: vlastita izrada)

EVIDENCIJA OZLJEDE DJETETA

Odgojna skupina:	Bubamare
Broj nazočne djece:	3
Ime i prezime djeteta:	
Datum rođenja djeteta:	13.05.2014.
Ime i prezime roditelja:	
Adresa:	Ulica 11
Kontakt majke:	0911111256
Kontakt oca:	0915642698
Vrsta ozljede:	Dijete se igralo s ostalom djecom iz skupine. U pitanju je bila igra skrivača. Tijekom trčanja, spotaknulo se i palo na koljena. Ogrebalo je desno koljeno, a lijevo blago natuklo.
Vrijeme ozljede:	Oko 13 h.
Mjesto ozljede:	Dvorište dječjeg vrtića.
Poduzete mjere zbrinjavanja:	Dijete je umireno i stavljen je oblog na natučeno koljeno. Oderano koljeno je sanirano dezinfekcijskim sprejem i stavljen je flaster. Nije bilo potrebe zvati roditelje.

Bikovec, 13.04.2023. godine.

RAVNATELJ: M.P. ODGOJITELJ:

_____ _____

Nina Ninić Admin Admin

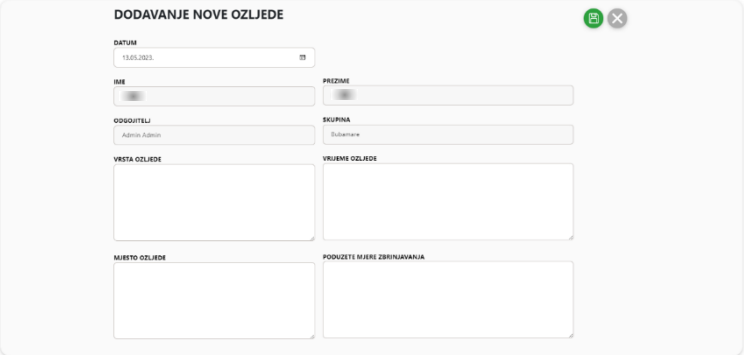
Slika 90. Ispis evidencije ozljede djeteta

(Izvor: vlastita izrada)

Pritiskom na gumb za dodavanje pojavljuje se ekran s formom prikazanom na slici 91. Forma ima tablicu s nabrojanom djecom skupine izabrane u filteru prethodnog ekrana te tri onemogućena *input* elementa s automatski ispunjenim trenutnim datumom, korisnikom koji je trenutno prijavljen i skupini koja je prethodno izabrana. Nakon što se pritisne na gumb za označavanje pored djeteta koje se ozlijedilo, pojavljuje se ekran s formom prikazan na slici 92. Forma koja se pojavila ima jedan *input* element tipa *date* za datum ozljede, četiri onemogućena *input* elementa s automatski ispunjenim imenom i prezimenom izabranog djeteta, korisnikom koji je trenutno prijavljen i skupini kojoj dijete pripada, četiri *textarea* elementa za upis vrste ozljede, vremena ozljede, mjesta ozljede i poduzetih mjera zbrinjavanja. Forma se sprema pritiskom na gumb za spremanje, a zatvara s pritiskom na gumb za zatvaranje.

Slika 91. Dodavanje ozljede djeteta - 1.dio

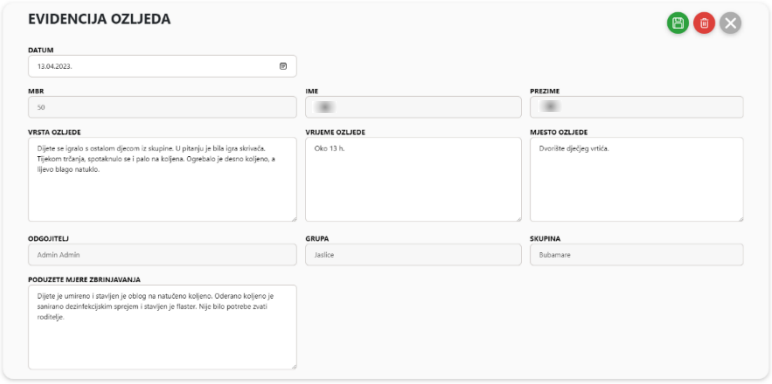
(Izvor: vlastita izrada)



Slika 92. Dodavanje ozljede djeteta - 2.dio

(Izvor: vlastita izrada)

Pritiskom na gumb za uređivanje ozljede pojavljuje se ekran s formom prikazanom na slici 93. Forma ima jedan *input* element tipa *date* za datum ozljede, šest onemogućenih *input* elemenata s matičnim brojem (MBR), imenom i prezimenom djeteta, korisnikom koji je evidentirao ozljedu, grupom i skupinom kojoj dijete pripada te četiri *textarea* elementa za upis vrste ozljede, vremena ozljede, mjesta ozljede i poduzetih mjera zbrinjavanja. Forma se sprema pritiskom na gumb za spremanje, briše pritiskom na gumb za brisanje, a zatvara s pritiskom na gumb za zatvaranje.

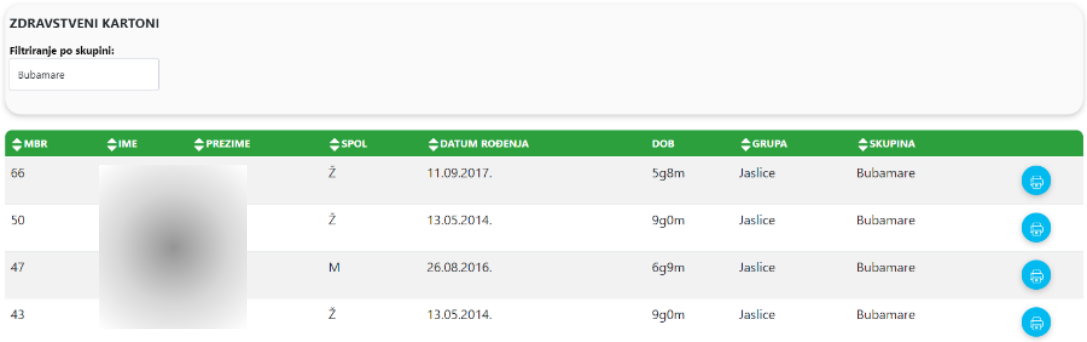


Slika 93. Detalji/uređivanje ozljede djeteta

(Izvor: vlastita izrada)

1.9.3.5. Zdravstveni kartoni djece

Funkcionalnost zdravstvenih kartona djece sastoji se od jednog ekrana s prikazom popisa djece (slika 94.). Na vrhu ekrana nalazi se filter za filtriranje prema skupini (*select form elementi*), a ispod filtera nalazi se tablica s djecom. U svakom redu postoji gumb za ispis zdravstvenog kartona. Ispis zdravstvenog kartona prikazan je na slici 95.



Slika 94. Pregled djece

(Izvor: vlastita izrada)

Slika 95. Ispis zdravstvenog kartona

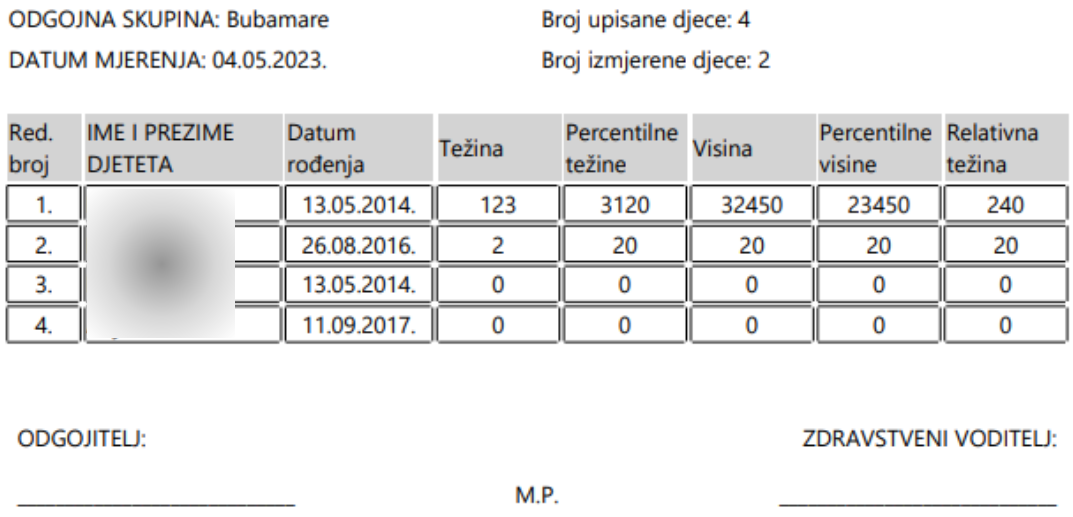
(Izvor: vlastita izrada)

1.9.3.6. Evidencija antropometrijskog mjerenja djece

Funkcionalnost evidencije antropometrijskih mjerenja djece sastoji se od dva različita ekrana: prikaz popisa antropometrijskih mjerenja (slika 96.) i dodavanje/uređivanje antropometrijskih mjerenja (slika 98.).

Ekran s popisima *Evidencija antropometrijskih mjerenja djece* na vrhu ima filter za filtriranje prema skupini i pedagoškoj godini (*select form elementi*). Ispod filtera nalazi se gumb za dodavanje novog mjerenja te tablica s postojećim antropometrijskim mjerenjima djece. U svakom redu postoje dva gumba za ispis evidencije (prikazan na slici 97.) i za uređivanje pripadajućeg antropometrijskog mjerenja.

Slika 96. Pregled antropometrijskih mjerenja djece
(Izvor: vlastita izrada)



ODGOJNA SKUPINA: Bubamare
Broj upisane djece: 4
DATUM MJERENJA: 04.05.2023.
Broj izmjerene djece: 2

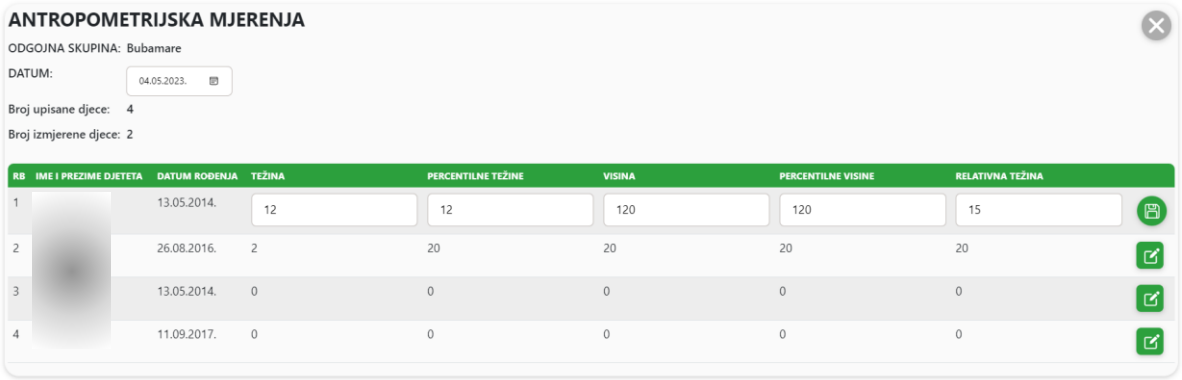
Red. broj	IME I PREZIME DJETETA	Datum rođenja	Težina	Percentilne težine	Visina	Percentilne visine	Relativna težina
1.		13.05.2014.	123	3120	32450	23450	240
2.		26.08.2016.	2	20	20	20	20
3.		13.05.2014.	0	0	0	0	0
4.		11.09.2017.	0	0	0	0	0

ODGOJITELJ: _____
M.P.
ZDRAVSTVENI VODITELJ: _____

Slika 97. Ispis antropometrijskog mjerenja
(Izvor: vlastita izrada)

Pritiskom na gumb za dodavanje i uređivanje antropometrijskog mjerenja pojavljuje se ekran s formom prikazanom na slici 98. Forma ima jedan *input* element tipa *date* za datum

antropometrijskog mjerenja, informacije o skupini (izabranoj u filteru na prethodnom ekranu) i broju upisane te izmjerene djece. Ispod spomenutog nalazi se tablica s djecom i njihovim mjerama koje se mogu uređivati pritiskom na gumb za uređivanje u redu s imenom pojedinog djeteta. Pritiskom na spomenuti gumb, pojavljuju se *editabilni input* elementi za težinu, percentilnu težinu, visinu, percentilnu visinu i relativnu težinu. Promjene u redu se spremaju pritiskom na gumb za spremanje. Dijete se smatra izmjerenim tek kad ima unesenu vrijednost i za težinu i za visinu.



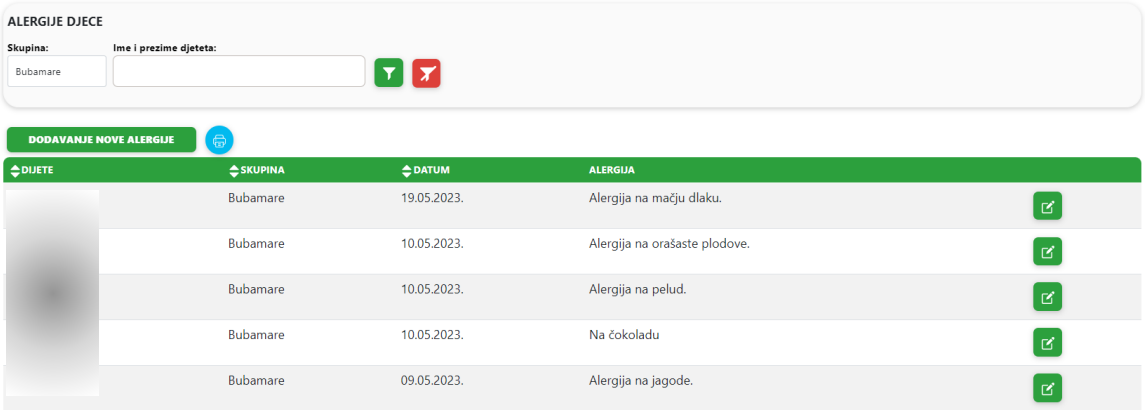
Slika 98. Dodavanje/uređivanje antropometrijskog mjerenja

(Izvor: vlastita izrada)

1.9.3.7. Alergije djece

Funkcionalnost evidencije alergija djece sastoji se od dva različita ekrana: prikaz popisa alergija djece (slika 99.) i dodavanje/uređivanje alergija (slika 101. i slika 102.).

Ekran s popisom alergija djece na vrhu ima filter za filtriranje prema skupini te prema imenu i/ili prezimenu (*input form element*). Ispod filtera nalazi se gumb za dodavanje nove alergije, ispis popisa alergija djece (prikazan na slici 100.) te tablica s postojećim alergijama djece. U svakom redu postoji gumb za uređivanje pripadajuće alergije.



Slika 99. Pregled alergija djece

(Izvor: vlastita izrada)

ALERGIJE DJECE

DIJETE	SKUPINA	DATUM	OPIS ALERGIJE
	Bubamare	19.05.2023.	Alergija na mačju dlaku.
	Bubamare	10.05.2023.	Alergija na orašaste plodove.
	Bubamare	10.05.2023.	Alergija na pelud.
	Bubamare	10.05.2023.	Na čokoladu
	Bubamare	09.05.2023.	Alergija na jagode.

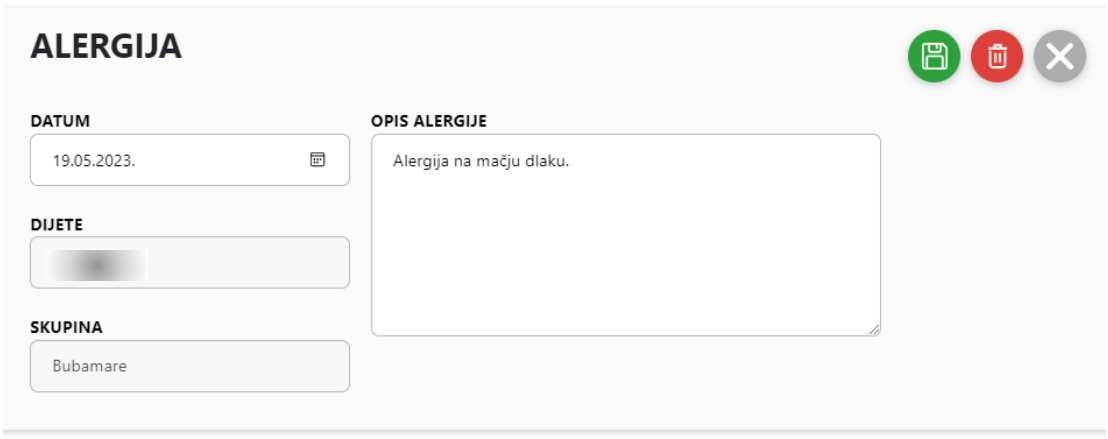
Slika 100. Ispis alergija djece

(Izvor: vlastita izrada)

Pritiskom na gumb za dodavanje pojavljuje se ekran s formom prikazanom na slici 101. Forma ima tablicu s nabrojanom djecom skupine izabrane u filteru prethodnog ekrana te dva onemogućena *input* elementa s automatski ispunjenim trenutnim datumom i skupinom koja je prethodno izabrana. Nakon što se pritisne na gumb za označavanje, pored djeteta za koje se dodaje alergija, pojavljuje se ekran s formom prikazan na slici 102. Forma koja se pojavila ima jedan *input* element tipa *date* za datum dijagnoze alergije, dva onemogućena *input* elementa s automatski ispunjenim imenom i prezimenom izabranog djeteta te skupinom kojoj dijete pripada i jedan *textarea* element za upis opisa alergije. Forma se sprema pritiskom na gumb za spremanje, a zatvara s pritiskom na gumb za zatvaranje. Ista forma prikazuje se i prilikom uređivanja alergije samo što još postoji gumb za brisanje alergije.

Slika 101. Dodavanje alergije djeteta - 1.dio

(Izvor: vlastita izrada)



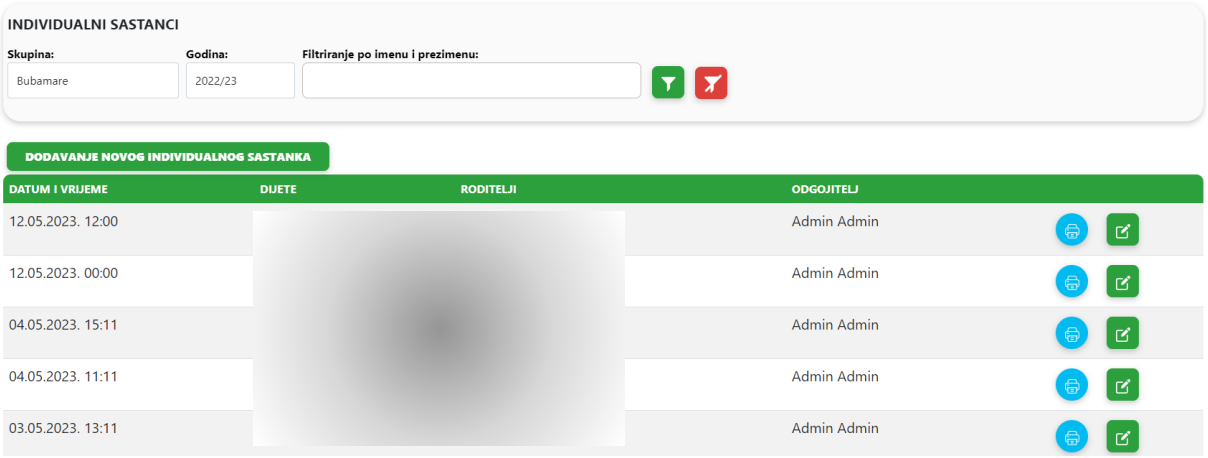
Slika 102. Dodavanje alergije djeteta - 2. dio i uređivanje alergije djeteta

(Izvor: vlastita izrada)

1.9.3.8. Evidencija individualnih sastanaka s roditeljima

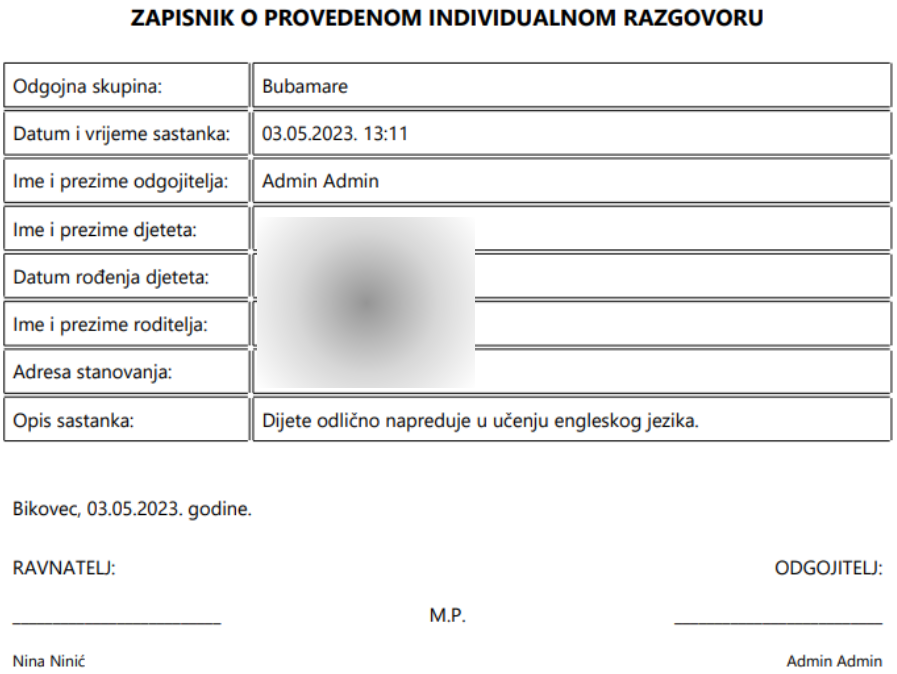
Funkcionalnost evidencije individualnih sastanaka sastoji se od dva različita ekrana: prikaz popisa individualnih sastanaka (slika 103.) i dodavanje/uređivanje individualnog sastanka (slika 105., slika 106. i slika 107.).

Ekran s popisom individualnih sastanaka na vrhu ima filter za filtriranje prema skupini, pedagoškoj godini (*select form elementi*) te prema imenu i/ili prezimenu (*input form* element). Ispod filtera nalazi se gumb za dodavanje novog individualnog sastanka te tablica s provedenim individualnim sastancima. U svakom redu postoje dva gumba za ispis individualnog sastanka (prikazano na slici 104.) i za uređivanje pripadajućeg individualnog sastanka.



Slika 103. Pregled individualnih sastanaka

(Izvor: vlastita izrada)



ZAPISNIK O PROVEDENOM INDIVIDUALNOM RAZGOVORU

Odgojna skupina:	Bubamare
Datum i vrijeme sastanka:	03.05.2023. 13:11
Ime i prezime odgojitelja:	Admin Admin
Ime i prezime djeteta:	
Datum rođenja djeteta:	
Ime i prezime roditelja:	
Adresa stanovanja:	
Opis sastanka:	Dijete odlično napreduje u učenju engleskog jezika.

Bikovec, 03.05.2023. godine.

RAVNATELJ: ODGOJITELJ:

_____ M.P. _____

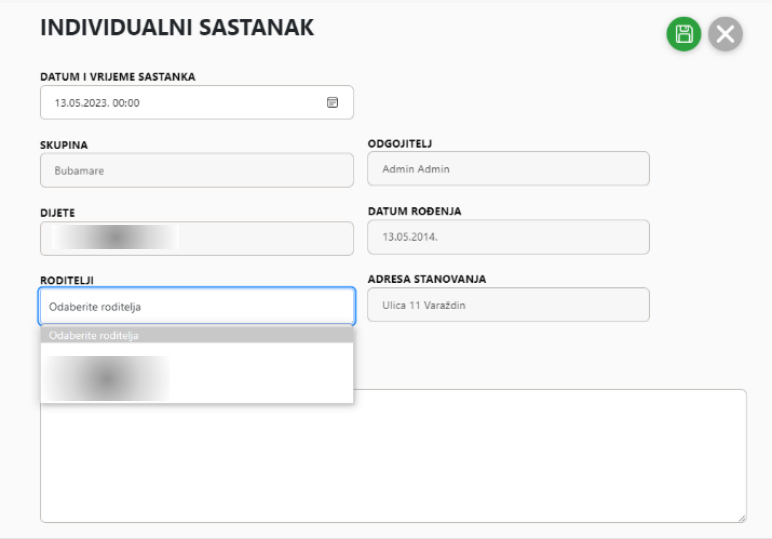
Nina Ninić Admin Admin

Slika 104. Ispis individualnog sastanka
(Izvor: vlastita izrada)

Pritiskom na gumb za dodavanje pojavljuje se ekran s formom prikazanom na slici 105. Forma ima tablicu s nabrojanom djecom skupine izabrane u filteru prethodnog ekrana te tri onemogućena *input* elementa s automatski ispunjenim trenutnim datumom, skupinom koja je prethodno izabrana i trenutno prijavljenim korisnikom. Nakon što se pritisne na gumb za označavanje pored djeteta s čijim roditeljem se obavlja individualni sastanak, pojavljuje se ekran s formom prikazanom na slici 106. Forma koja se pojavila ima jedan *input* element tipa *datetime* za datum i vrijeme početka individualnog sastanka, pet onemogućenih *input* elementa s automatski ispunjenim podacima o nazivu skupine kojoj dijete pripada, odgojitelju koji je zabilježio sastanak, imenu i prezimenu izabranog djeteta, njegovom datumu rođenja i adresi stanovanja, jedan *select form* element za roditelja s kojim je sastanak obavljen (ponuđene su tri opcije majka, otac te otac i majka - nakon što se odabere jedna od opcija, *form* element se pretvara u *input* element te omogućuje promjenu opcije u željeni sadržaj) te jedan *textarea* element za upis opisa sastanka. Forma se sprema pritiskom na gumb za spremanje, a zatvara s pritiskom na gumb za zatvaranje. Ista forma prikazuje se i prilikom uređivanja individualnog sastanka samo što još postoji gumb za brisanje sastanka (slika 107.).

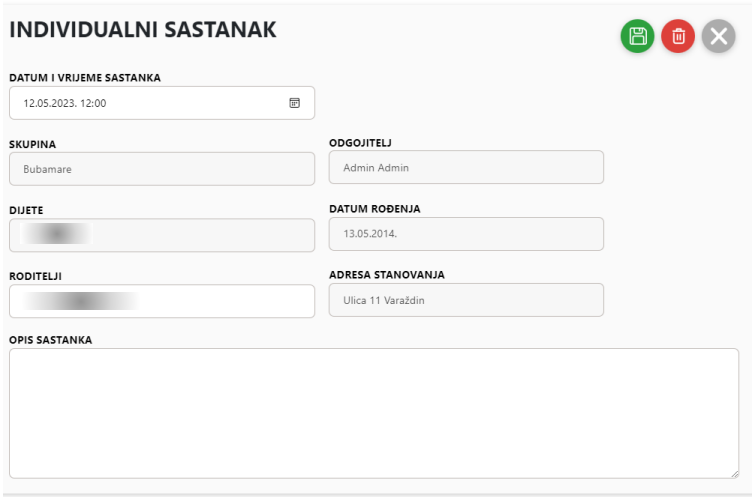
Slika 105. Dodavanje individualnog sastanka - 1.dio

(Izvor: vlastita izrada)



Slika 106. Dodavanje individualnog sastanka - 2.dio

(Izvor: vlastita izrada)



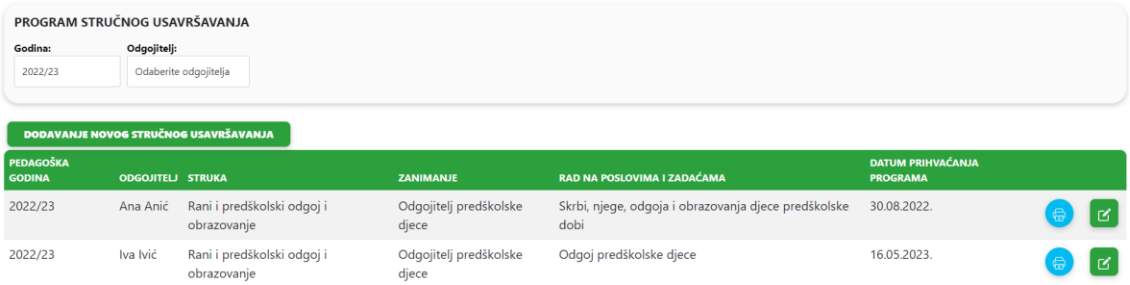
Slika 107. Uređivanje individualnog sastanka

(Izvor: vlastita izrada)

1.9.3.9. Program stručnog usavršavanja odgojitelja

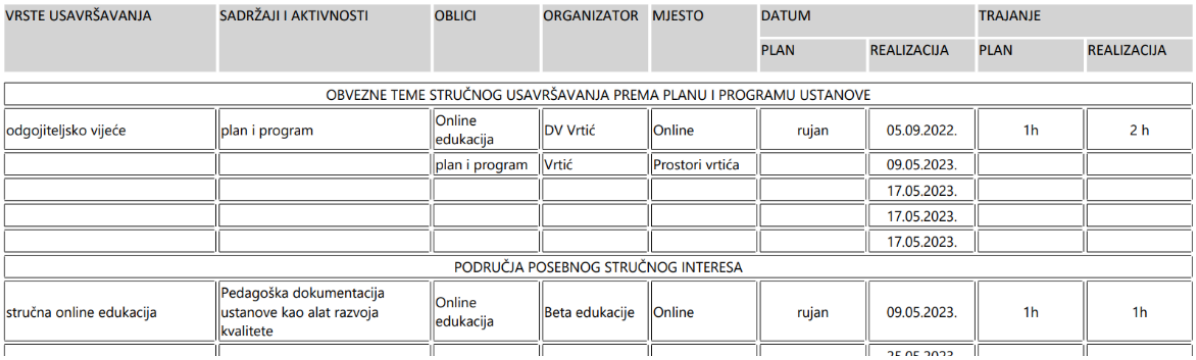
Funkcionalnost programa stručnog usavršavanja odgojitelja sastoji se od tri različita ekrana: prikaz popisa programa stručnog usavršavanja (slika 108.), dodavanje/uređivanje programa stručnog usavršavanja (slika 110.) i dodavanje/uređivanje teme u programu stručnog usavršavanja (slika 111.).

Ekran s popisom programa stručnog usavršavanja na vrhu ima filter za filtriranje prema pedagoškoj godini i odgojitelju (*select form elementi*). Ispod filtera nalazi se gumb za dodavanje novog programa stručnog usavršavanja te tablica s postojećim programima stručnog usavršavanja. U svakom redu postoje dva gumba za ispis programa stručnog usavršavanja (prikazano na slici 109.) i za uređivanje pripadajućeg programa stručnog usavršavanja.



Slika 108. Popis programa za stručno usavršavanje odgojitelja

(Izvor: vlastita izrada)

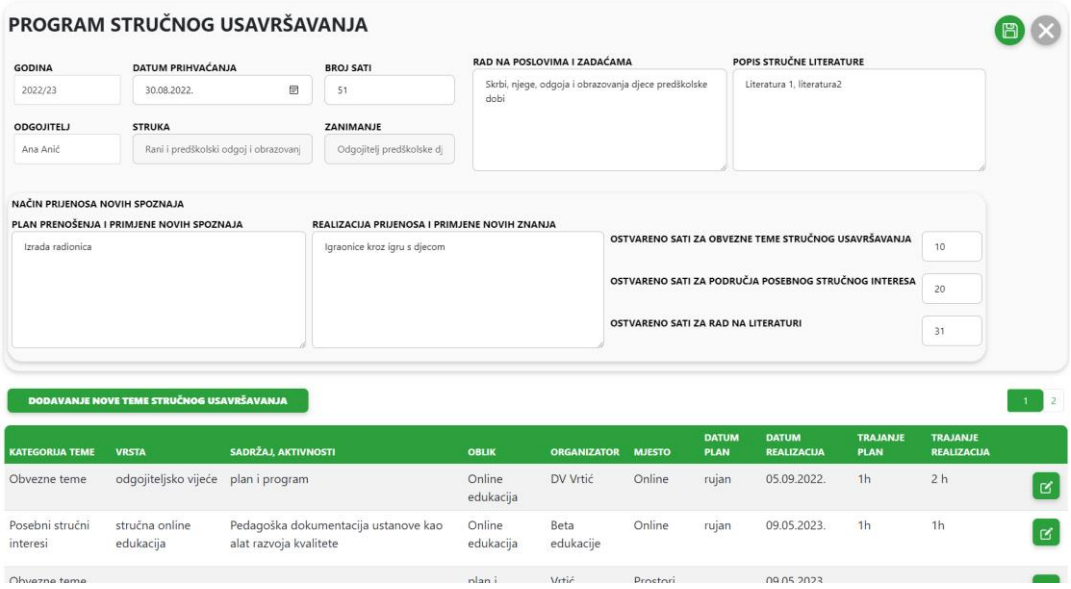


VRSTE USAVRŠAVANJA	SADRŽAJI I AKTIVNOSTI	OBLICI	ORGANIZATOR	MJESTO	DATUM		TRAJANJE	
					PLAN	REALIZACIJA	PLAN	REALIZACIJA
OBVEZNE TEME STRUČNOG USAVRŠAVANJA PREMA PLANU I PROGRAMU USTANOVE								
odgojiteljsko vijeće	plan i program	Online edukacija	DV Vrtić	Online	rujan	05.09.2022.	1h	2 h
		plan i program	Vrtić	Prostori vrtića		09.05.2023.		
						17.05.2023.		
						17.05.2023.		
						17.05.2023.		
PODRUČJA POSEBNOG STRUČNOG INTERESA								
stručna online edukacija	Pedagoška dokumentacija ustanove kao alat razvoja kvalitete	Online edukacija	Beta edukacije	Online	rujan	09.05.2023.	1h	1h

Slika 109. Ispis programa stručnog usavršavanja

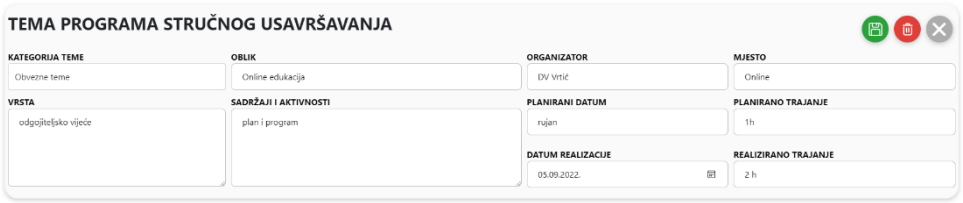
(Izvor: vlastita izrada)

Pritiskom na gumb za dodavanje ili uređivanje pojavljuje se ekran s formom prikazanom na slici 110. Forma ima pet *input* elementa za upis datuma prihvaćanja programa stručnog usavršavanja (*type date*), za broj planiranih sati, ostvarenih sati za obvezne teme, područja posebnog stručnog interesa te za rad na stručnoj literaturi (*type number*), dva onemogućena (*disabled*) *input* elementa za struku i zanimanje odgojitelja koji se automatski popunjavaju s obzirom na izabranog odgojitelja, dva *select* elementa za odabir pedagoške godine i odgojitelja, četiri *textarea* elementa za opis poslova i zadaća, za popis literature, za plan prenošenja i primjene spoznaja te za realizaciju prijenosa i primjene novih znanja. Forma se sprema pritiskom na gumb za spremanje, briše pritiskom na gumb za brisanje (vidljiv samo ako se program uređuje te nema dodanih tema), a zatvara s pritiskom na gumb za zatvaranje. Ispod spomenutih elemenata forme nalazi se gumb za dodavanje tema unutar samog programa za stručno usavršavanje te tablica već dodanih tema s pripadajućim gumbom za uređivanje.



Slika 110. Dodavanje/uređivanje programa stručnog usavršavanja
(Izvor: vlastita izrada)

Pritiskom na gumb za dodavanje ili uređivanje teme pojavljuje se ekran s formom prikazanom na slici 111. Forma ima sedam *input* elemenata za upis oblika teme/edukacije, za naziv organizatora, za mjesto održavanja, planirani datum, planirano trajanje, datum realizacije (*type date*) i realizirano trajanje, jedan *select* element za odabir kategorije teme (*Obavezne teme* ili *Posebni stručni interesi*), dva *textarea* elementa za vrstu te sadržaj i aktivnosti same teme/edukacije. Forma se sprema pritiskom na gumb za spremanje, briše pritiskom na gumb za brisanje, a zatvara s pritiskom na gumb za zatvaranje.

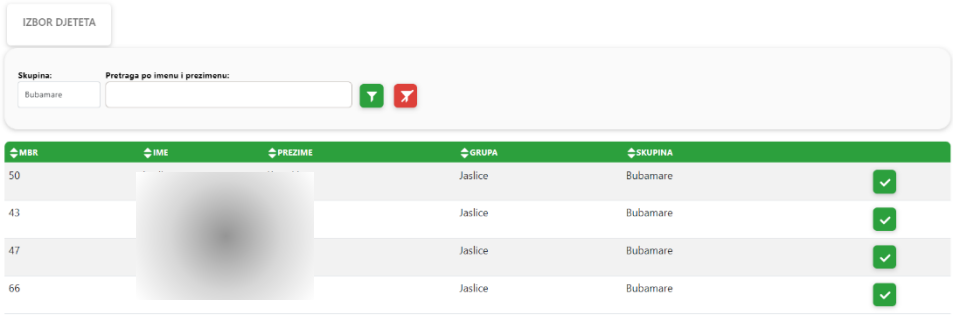


Slika 111. Dodavanje/uređivanje teme programa stručnog usavršavanja

(Izvor: vlastita izrada)

1.9.3.10. Brzi pregled djece

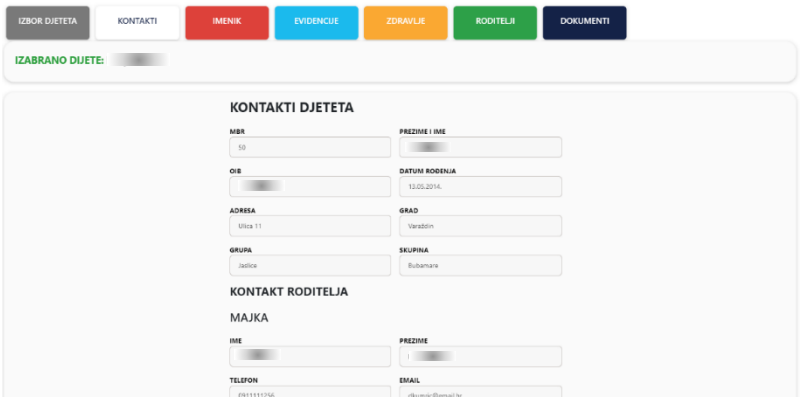
Funkcionalnost brzog pregleda djece omogućuje pregled i uređivanje svih informacija vezanih uz pojedino dijete na jednom mjestu. Ulaskom u funkcionalnost prikazuje se popis djece koja se filterom mogu filtrirati prema skupini te imenu i/ili prezimenu. Pored svakog djeteta nalazi se gumb za odabir. Navedeni ekran vidljiv je na slici 112.



Slika 112. Brzi pregled djeteta - *Izbor djeteta*

(Izvor: vlastita izrada)

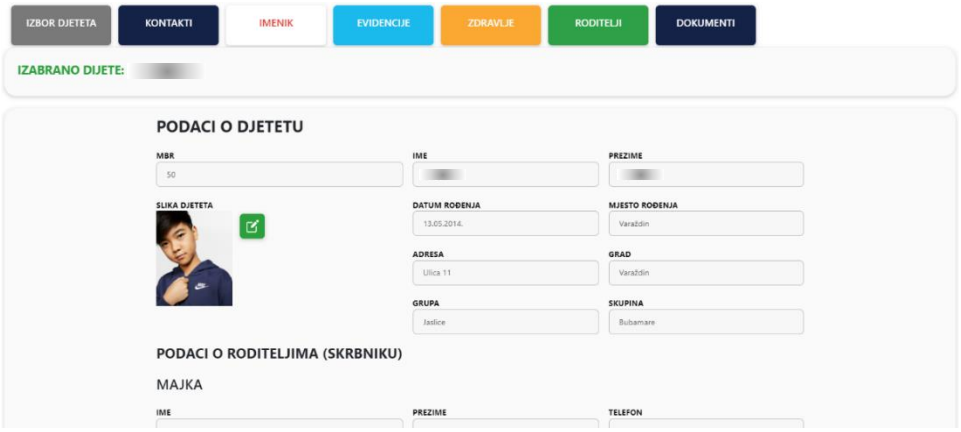
Nakon pritiska na gumb za odabir, otvara se ekran s važnim informacijama i kontaktima vezanim uz izabrano dijete prikazan na slici 113. Na vrhu ekrana nalaze se kartice za brzu navigaciju kroz funkcionalnost.



Slika 113. Brzi pregled djeteta - *Kontakti*

(Izvor: vlastita izrada)

Pritiskom na karticu *Imenik*, otvara se ekran s pregledom podataka vezanim uz imenik izabranog djeteta te je omogućena promjena djetetove slike. Opisano vidljivo je na slici 114.



Slika 114. Brzi pregled djeteta - *Imenik*

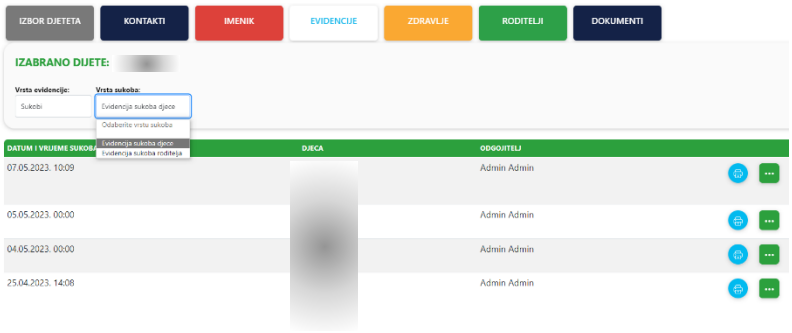
(Izvor: vlastita izrada)

Pritiskom na karticu *Evidencije*, otvara se ekran s filterom za vrstu evidencije. Ako se odabere evidencija nazočnosti, prikazuje se ekran vidljiv na slici 115. Na spomenutom ekranu vidljiva je tablica sa svim provedenim evidencijama nazočnosti te je li dijete bilo nazočno ili ne. Desno od tablice nalazi se filter za mjesec i pedagošku godinu te prikaz koliko je puta u izabranom mjesecu i godini provedena evidencija, koliko je puta dijete bilo nazočno, a koliko ne.

Slika 115. Brzi pregled djeteta - *Evidencije nazočnosti*

(Izvor: vlastita izrada)

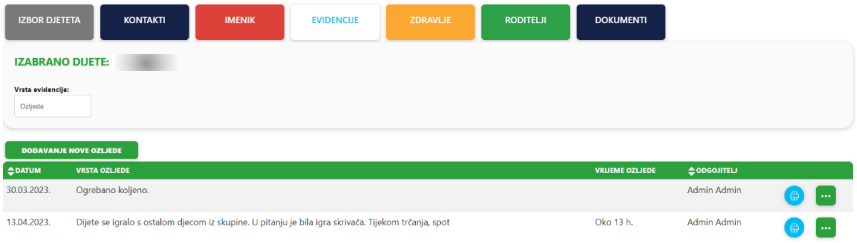
Ako se odabere evidencija sukoba, prikazuje se ekran vidljiv na slici 116. Filter dobiva još jednu opciju odabira vrste sukoba (među djecom ili među roditeljima). Na spomenutom ekranu vidljiva je tablica sa svim provedenim evidencijama sukoba te je moguće ispisati pojedini sukob ili pregledati podatke o njemu pritiskom na odgovarajuće gumbe unutar svakog retka tablice.



Slika 116. Brzi pregled djeteta - *Evidencije sukoba*

(Izvor: vlastita izrada)

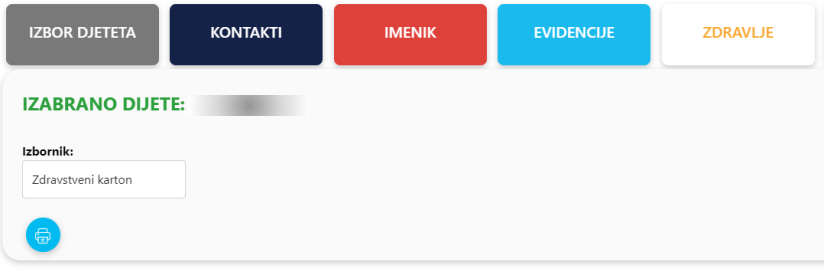
Ako se odabere evidencija ozljeda, prikazuje se ekran vidljiv na slici 117. Na spomenutom ekranu vidljiva je tablica sa svim evidentiranim ozljedama djeteta, moguće je dodati novu ozljedu te ispisati ili urediti postojeću na isti način kao što je opisano u poglavlju 5.3.8. Dio baze podataka *Ozljede djece*.



Slika 117. Brzi pregled djeteta - *Evidencije ozljeda*

(Izvor: vlastita izrada)

Pritiskom na karticu *Zdravlje*, otvara se ekran s izbornikom za pregled zdravstvenog kartona, antropometrijskih mjerenja ili alergija. Ako se odabere zdravstveni karton, prikazuje se ekran vidljiv na slici 118. koji ima gumb za ispis zdravstvenog kartona djeteta.



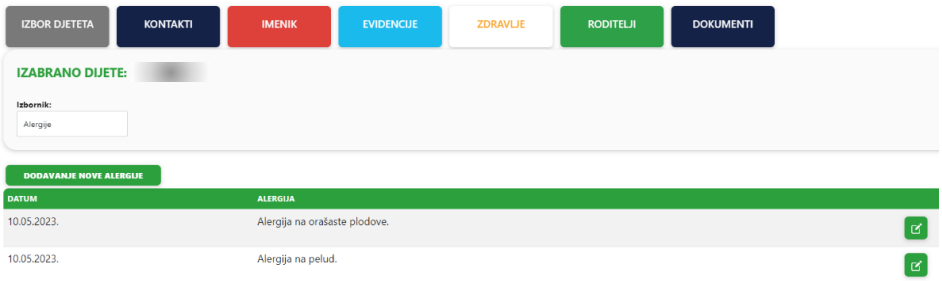
Slika 118. Brzi pregled djeteta – *Zdravstveni karton*

(Izvor: vlastita izrada)

Ako se odaberu antropometrijska mjerenja prikazuje se ekran vidljiv na slici 119. koji prikazuje tablicu s pregledom svih djetetovih antropometrijskih mjerenja.

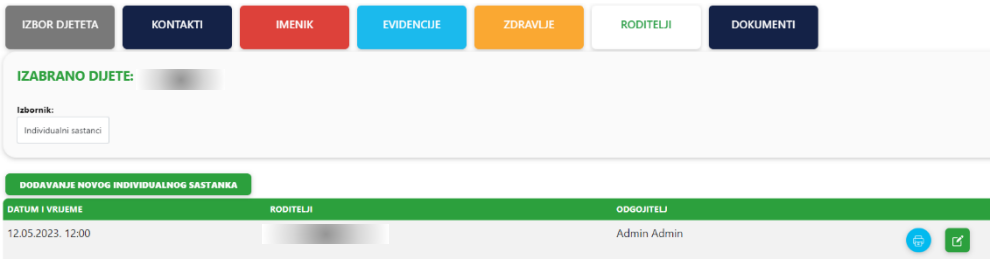
Slika 119. Brzi pregled djeteta – *Antropometrijska mjerenja*
(Izvor: vlastita izrada)

Ako se odabere opcija alergije, prikazuje se ekran vidljiv na slici 120. Na spomenutom ekranu vidljiva je tablica sa svim evidentiranim alergijama djeteta, moguće je dodati novu alergiju te urediti postojeću na isti način kao što je opisano u poglavlju 5.3.10. Dio baze podataka *Alergije djece*.



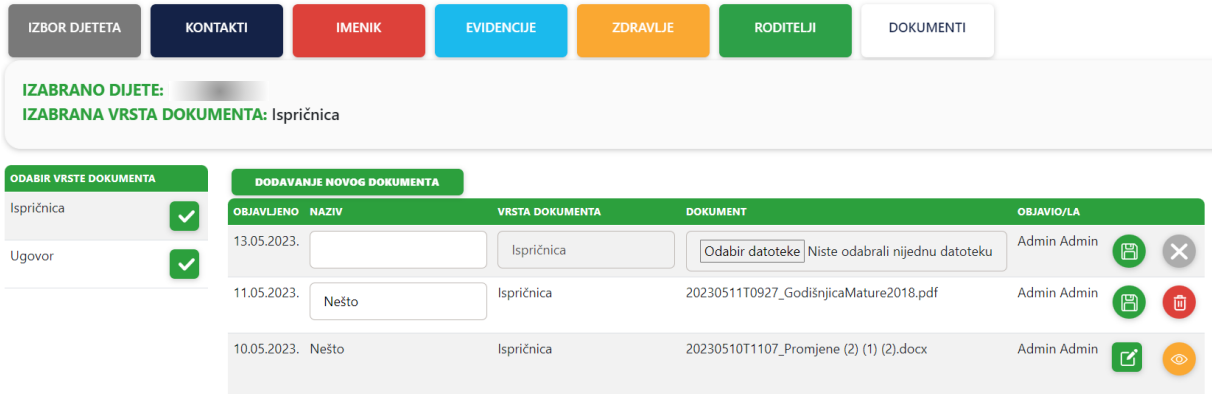
Slika 120. Brzi pregled djeteta – *Alergije*
(Izvor: vlastita izrada)

Pritiskom na karticu *Roditelji*, otvara se ekran vidljiv na slici 121. koji prikazuje tablicu s pregledom svih individualnih sastanaka provedenih s djetetovim roditeljima, moguće je dodati novi individualni sastanak te ispisati ili urediti postojeći na isti način kao što je opisano u poglavlju 5.3.11. Dio baze podataka *Individualni sastanci.*



Slika 121. Brzi pregled djeteta – *Individualni sastanci*
(Izvor: vlastita izrada)

Pritiskom na karticu *Dokumenti*, otvara se ekran vidljiv na slici 122. koji prikazuje tablicu sa svim mogućim vrstama dokumenata. Odabirom jedne od njih prikazuje se još jedna tablica s dokumentima djeteta spremljenih pod odabranom vrstom dokumenta. Također, moguće je dodati i novi dokument pritiskom na gumb za dodavanje. Prvi red tablice s dokumentima prikazuje redak koji se pojavljuje nakon pritiska na gumb za dodavanje, sadrži datum dodavanja dokumenta (dodaje se automatski prema trenutnom datumu), opisni naziv (*input form element*), vrsta dokumenta (dodaje se automatski prema izabranoj vrsti dokumenta) te sami dokument (*input form element* tipa *file*). Gumb spremanja sprema dodani dokument, a gumb zatvaranja, poništava dodavanje dokumenta te dodani red nestaje iz tablice. Drugi red tablice prikazuje redak koji se pojavljuje nakon pritiska na gumb za uređivanje postojećeg dokumenta (gumb vidljiv u trećem retku), sadrži datum dodavanja dokumenta, vrstu dokumenta i sam dokument koji se ne mogu uređivati te opisni naziv (*input form element*) koji postaje *editabilan*. Gumb spremanja sprema uređeni dokument, a gumb brisanja ga briše te uređivani red nestaje iz tablice. Treći red tablice prikazuje kako izgleda dokument kad je već dodan i nije u fazi uređivanja. Sadrži gumb za uređivanje koji red pretvara u oblik prikazan u drugom redu te gumb za prikaz koji u novom prozoru preglednika otvara konkretni dokument za pregled.



Slika 122. Brzi pregled djeteta – *Dokumenti*

(Izvor: vlastita izrada)

1.10. Izrada aplikacije

Aplikacija je razvijena *Model-View-Controller* modulom, sadrži 60 modela, 44 *View*-a i 8 kontrolera. Osim navedenog, sadrži i 20 klasa za sortiranje, jednu klasu vezanu uz *pager* (*PagerTagHelper*), 18 *ViewModel*-a, jednu klasu za opće postavke aplikacije (*AppSettings*) te glavnu programsku klasu za pokretanje aplikacije (*Program*).

Modeli u aplikaciji su: *Akcije* i *LogAktivnosti* vezane uz akcije koje korisnik provodi u aplikaciji (npr. prijava) i njihovo bilježenje u bazi podataka, *Alergije* i *AlergijeView* vezane uz alergije, *AntropMjer*, *AntropMjerDjeca*, *AntropMjerPopisView* i *AntropMjerView* vezani uz antropometrijska mjerenja djeteta, *Djeca, DjecaDokumenti, DjecaDokumentiView i DjecaView* vezane uz podatke djeteta, *IndividualniSastanci* i *IndividualniSastanciView* vezani uz individualne sastanke s roditeljima djece, *Odgojitelji*, *Skupine, OdgojiteljSkupina, SkupineView*, *Grupe* i *SkupineGrupeView* vezani uz odgojitelje, skupine i njihove grupe te poveznice između njih, *Korisnici, Uloge* i *KorisniciView* vezani uz korisnike aplikacije i njihove uloge, *Liste* za potrebe kreiranja lista za *select form* elemente, *Login* za potrebe prijave u aplikaciju, *Nazocnosti*, *NazocnostiView* i *NazocnostiZadnjiDatum* vezani uz evidenciju nazočnosti djece, *Gradovi, NeradniDani*, *Nacionalnosti, VrsteDokumenata* i *Pg* vezani uz šifrarnike u aplikaciji, *Ozljede* i *OzljedeView* vezane uz ozljede djece, *ProgStrucUsav*, *ProgStrucUsavDetalji*, *ProgStrucUsavPopisView* i *ProgStrucUsavView* vezani uz program stručnog usavršavanja odgojitelja, *SanitarniNadzorDokumenti*, *SanitarniNadzor* i *SanitarniNadzorView* vezani uz sanitarni nadzor, *Sukobi*, *SukobiDjeca*, *SukobiDjecaView*, *SukobiRoditelji*, *SukobiRoditeljiDjeca* i *SukobiRoditeljiView* vezani uz sukobe među roditeljima i među djecom, *Vrtic* vezan uz glavne podatke vrtića, *FilterDjeca*, *FilterKorisnici*, *FilterNazocnosti*, *FilterPopisGradovi*, *FilterPopisNacionalnosti*, *FilterPopisNeradniDani*, *FilterPopisPG*, *FilterPopisSkupine*, *FilterPopisVrsteDokumenata*, *FilterProgStrucUsav* vezani uz filtere u aplikaciji. Osim nabrojanog, u modelima se nalazi i kontekst baze podataka *DvContext*. On se koristi za postavljanje upita i spremanje podataka u bazu podataka. Sadrži dvije zaštićene (*protected*) metode *OnConfiguring* i *OnModelCreating*. Prva postavlja *string* za konekciju s bazom podataka, a njezin programski kod nalazi se u nastavku:

```
protected override void OnConfiguring(DbContextOptionsBuilder
optionsBuilder)
        => optionsBuilder.UseSqlServer("string");
```

Programski kod 22: *OnConfiguring* metoda u *DVContext*

(Izvor: vlastita izrada)

Druga metoda gradi modele. Primjer programskog koda metode *OnModelCreating* za model *Alergije* nalazi se u nastavku:

```
modelBuilder.Entity<Alergije>(entity =>
{
      entity.HasKey(e => e.Idalergija);
      entity.ToTable("Alergije");
      entity.Property(e =>
      e.Idalergija).ValueGeneratedNever().HasColumnName("IDAlergija")
      ;
      entity.Property(e =>
      e.AlergijaDatum).HasColumnType("datetime");
      entity.Property(e => e.AlergijaOpis).HasMaxLength(1000);
      entity.Property(e =>
      e.Iddijete).HasDefaultValueSql("((0))").HasColumnName("IDDijete
      ");
      entity.Property(e =>
      e.Idskupina).HasDefaultValueSql("((0))").HasColumnName("IDSkupi
      na");
});
```

Programski kod 23: *OnModelCreating* metoda u *DVContext*
(Izvor: vlastita izrada)

Kontroleri u aplikaciji su: *AutoCompleteController* (za brzo pretraživanje gradova unutar *select* elementa forme), *DjecaController*, *HomeController*, *KorisniciController*, *LoginController*, *OdgojiteljController*, *OstaliPopisiController* i *RavnateljController*.

View-evi u aplikaciji su grupirani po kontrolerima. *View*-evi u grupi *Djeca* su *DodajUredi* i *Index*, u grupi *Home* su *404* (stranica za *error 404*) i *Index,* u grupi *Korisnici* su *DodajUredi* i *Index*, u grupi *Login* je *Login,* u grupi *Odgojitelj* su *AlergijeDjece, AlergijeDjeceDodajUredi, AntropometrijskaMjerenja, AntropometrijskaMjerenjaDodajUredi, BrziKontaktiDjeteta, BrziPregledDjece, DokumentiDjeteta, EvidencijaNazocnosti, EvidencijaOzljeda, EvidencijaOzljedaDetaljiUredi, EvidencijaOzljedaDodaj, EvidencijaSukoba, EvidencijaSukobaDodajUredi, Evidencije, ImenikDjece, ImenikDjeceDetalji, IndividualniSastanci, IndividualniSastanciDodajUredi, ProgStrucUsavDetaljiDodajUredi, ProgStrucUsavDodajUredi, Roditelji, Zdravlje* i *ZdravstveniKartoni,* u grupi *OstaliPopisi* su *GradoviPopis*, *NacionalnostiPopis*, *NeradniDaniPopis*, *PGPopis*, *SkupinePopis* i *VrsteDokumenataPopis*, u grupi *Ravnatelj* su *MaticnaKnjigaDjece*, *MaticnaKnjigaDjeceDetalji*, *SanitarniNadzor*, *SanitarniNadzorDodajUredi* i *UpravljanjeSkupinama*, u grupi *Shared* (*View*-evi koji se pojavljuju u svakom drugom *View*-u) su *_Layout* (unutra se nalaze zaglavlje, izbornik i podnožje), *_Notification* (za poruke obavijesti *uspjesno* i *error*) čiji se primjer programskog koda za poruku *error* nalazi u nastavku:

```
@if (TempData["error"] != null)
{
      <div class="alert alert-warning alert-dismissible fade show"
      role="alert">
            <strong>@TempData["error"]</strong>
            <button type="button" class="btn-close" data-bs-
            dismiss="alert" aria-label="Close"></button>
      </div>
}
```

Programski kod 24: *View* za ispis *error* poruke - *_Notification*

(Izvor: vlastita izrada)

Klase za sortiranje u aplikaciji su: *AlergijeViewSort*, *AntropMjerenjaPopisSort*, *DjecaViewSort*, *GradoviViewSort*, *IndividualniSastanciViewSort*, *KorisniciViewSort*, *NacionalnostiViewSort*, *NazocnostiViewSort*, *NeradniDaniViewSort*, OzljedeViewSort, *PGViewSort*, *ProgStrucUsavPopisViewSort*, *ProgStrucUsavViewSort*, *SanitarniNadzorViewSort*, *SkupineSort*, *SukobiRoditeljiViewSort*, *SukobiViewSort* i *VrsteDokumenataViewSort*.

ViewModel-i za prijenos podataka s kontrolera na *View* su: *AntropMjerenjaViewModel*, *DijagramiViewModel*, *DjecaViewModel*, *GradoviViewModel*, *IsLabel*, *IndividualniSastanciViewModel*, *KorisniciViewModel*, *NacionalnostiViewModel*, *NazocnostiViewModel*, *NeradniDaniViewModel*, *OzljedeViewModel*, *PagingInfo*, *PGViewModel*, *ProgStrucUsavViewModel*, *SanitarniNadzorViewModel*, *SkupineGrupeViewModel*, *UpravljanjeSkupinamaViewModel* i *VrsteDokumenataViewModel*.

1.10.1. Prijava i autorizacija korisnika

Za potrebe prijave korisnika u aplikaciju, u kontroleru *LoginController.cs* napravljena je metoda *Login* koja predstavlja *get* metodu za istoimeni *View*. Ovdje je bitno spomenuti da se koristi *ClaimsPrincipal* koji predstavlja tip podatka za spremanje HTTP konteksta konkretnog prijavljenog korisnika. Ako je korisnikov identitet autentificiran, korisnik se preusmjerava na određeni ekran (*View*). Programski kod spomenute *get* metode nalazi se u nastavku:

```
…ClaimsPrincipal claimUser = HttpContext.User;
if (claimUser.Identity.IsAuthenticated)
{return RedirectToAction("Index", "Home");}
return View();…
```

Programski kod 25: *Get* metoda za prijavu

(Izvor: vlastita izrada)

Post Login metoda provjerava odgovaraju li upisano korisničko ime i lozinka korisniku spremljenom u bazi. Ako odgovaraju, tada se otvara nova lista *Claim* tipa te se u nju, za potrebe korisničke sesije, zapisuju željeni podaci o korisniku (u ovom slučaju korisničko ime korisnika, njegova uloga te ID). Dijelovi programskog koda navedene metode nalaze se u nastavku:

```
public IActionResult Login(Login modelLogin){
      …
      var korisnik = _db.Korisnicis.FirstOrDefaultAsync(m =>
      m.KorisnickoIme ==
      modelLogin.Username && m.KorisnikLozinka ==
      modelLogin.Password);
      if (korisnik != null){
            …
            var role = _db.Uloges.FirstOrDefaultAsync(m => m.Iduloga
            == korisnik.Iduloga);
            List<Claim> claims = new List<Claim>(){
                  new Claim(ClaimTypes.Name, modelLogin.Username),
                  new Claim(ClaimTypes.Role, role.UlogaNaziv),
                  new Claim(ClaimTypes.NameIdentifier,
                  korisnik.Idkorisnik.ToString())
            };
            ClaimsIdentity claimsIdentity = new
            ClaimsIdentity(claims,
            CookieAuthenticationDefaults.AuthenticationScheme);
            AuthenticationProperties properties = new
            HttpContext.SignIn(CookieAuthenticationDefaults.Authentic
            ationcheme, new ClaimsPrincipal(claimsIdentity),
            properties);
            …
```

Programski kod 26: *Post* metoda za prijavu

(Izvor: vlastita izrada)

Kako bi prijava i spremanje podataka korisnika u sesiju uopće radili, potrebno je uključiti servis za autentifikaciju te opcije kolačića (*cookie*) za login putanju i vrijeme moguće neaktivnosti korisnika prije nego se odjava korisnika izvrši sama. Opisani programski kod dodan je u *Program.cs*, a on je sljedeći:

```
builder.Services.AddAuthentication(
CookieAuthenticationDefaults.AuthenticationScheme).AddCookie(option
=>{option.LoginPath = "/Login/Login";option.ExpireTimeSpan =
      TimeSpan.FromMinutes(20);});
```

Programski kod 27: Dodavanje autentifikacije u *Program.cs*

(Izvor: vlastita izrada)

Za odjavu korisnika u *LoginController* dodana je sljedeća *LogOut* metoda koja prazni sadržaj sesije i prekida ju:

```
public IActionResult LogOut(){
      HttpContext.SignOut(CookieAuthenticationDefaults.Authentication
      Scheme);
      return RedirectToAction("Login", "Login");
}
```

Programski kod 28: Metoda za odjavu

(Izvor: vlastita izrada)

Kako korisnik ne bi mogao preko poveznice doći na njemu zabranjene ekrane aplikacije, bilo je potrebno na početku svake klase kontrolera dodati autorizaciju prema ulozi. Linija programskog koda za opisano nalazi se u nastavku:

```
[Authorize(Roles = "Odgojitelj, Admin")]
```

Programski kod 29: Autorizacija

(Izvor: vlastita izrada)

1.10.2. Tablice

Svi tablični prikazi u aplikaciji realizirani su na vrlo sličan način dohvaćanja podataka iz modela (migrirani iz baze podataka i smješteni u kontekst), njihovom obradom te slanjem obrađenih podataka u obliku *ViewModel*-a na pripadajući *View*. Primjer opisanog programskog koda *get* metode za popis djece nalazi se u nastavku:

```
…
var kidsContext = _db.DjecaViews.AsNoTracking();
int pagesize = appData.PageSize;
int count = kidsContext.Count ();
…
var pagingInfo = new PagingInfo
{CurrentPage = page, Sort = sort, Ascending = ascending,
ItemsPerPage = pagesize,TotalItems = count};
kidsContext = kidsContext.ApplySort(sort, !ascending);
var kids = kidsContext.Skip((page - 1) *
pagesize).Take(pagesize).ToList();
var model = new DjecaViewModel
{
      Djeca = kids, PagingInfo = pagingInfo
};
…
```

Programski kod 30: *Get* metoda za prikaz tablice djece

(Izvor: vlastita izrada)

U prethodnom programskom kodu vidljiva je varijabla `appData` iz koje se uzima vrijednost `PageSize`, a radi se o maksimalnom broju redova tablice na jednoj stranici. Osim spomenutog (`ItemsPerPage`), u `PagingInfo` smještaju se informacije o stranici (`Page`), sortiranju (`Sort`), radi li se o uzlaznom ili silaznom sortiranju (`Ascending`) te ukupnom broju redova za prikaz (`TotalItems`). *ViewModel* koji služi za prenošenje svih željenih podataka na sami *View* u ovom slučaju je `DjecaViewModel`. On, između ostalog, za potrebe ispisa tablice može primiti listu djece `Djeca` tipa `DjecaView` te `PagingInfo` vezan uz tehničku stranu prikaza tablice.

Za potrebe sortiranja stupaca tablice, za svaki od tipova podataka kreirana je klasa koja sortira podatke prema atributima. Konkretan primjer za djecu je klasa `DjecViewSort` prikazana u nastavku. Klasa prima broj koji predstavlja svojevrsni ID pojedinog atributa te informaciju želi li se stupac konkretnog atributa sortirati uzlazno ili silazno.

```
public static IQueryable<DjecaView> ApplySort(this
IQueryable<DjecaView> query, int sort, bool ascending)
{
      Expression<Func<DjecaView, object>> orderSelector = sort switch
      {
            1 => d => d.Iddijete,
            2 => d => d.DijetePrezime,
            3 => d => d.DijeteIme,
            …
      };
      if (orderSelector != null)
      {
            query = ascending ? query.OrderBy(orderSelector) :
            query.OrderByDescending(orderSelector);
      }
}
```

Programski kod 31: *DjecaViewSort*

(Izvor: vlastita izrada)

Za potrebe *pagera* tablice, kreirana je klasa `PagerTagHelper` koja upravlja njime na temelju `PageInfo` podataka. Sadrži metodu `Process` koja služi za stvaranje *pager* elementa za lakše kretanje kroz stranice tablice, metodu `BuildListItemForPage` za obradu sadržaja stranice tablice te metodu `BuildListItemForCurrentPage` za obradu sadržaja trenutne stranice tablice.

Pripadajući *View* prima poslani model iz *get* metode za prikaz popisa djece sljedećom linijom programskog koda:

```
@model DV.ViewModels.DjecaViewModel
```

Programski kod 32: Primanje *ViewModel*-a

(Izvor: vlastita izrada)

Za prikaz elementa *pager* na ekranu dodan je sljedeći programski kod:

```
<pager page-info="@Model.PagingInfo" page-filter="@Model.Filter"
class="float-end" style="padding: 0.375rem;"></pager>
```

Programski kod 33: Prikaz *pager* elementa
(Izvor: vlastita izrada)

Prikaz djece u obliku tablice nalazi se unutar *html* elementa `<table>`. Unutar zaglavlja `<thead>` tablice nalaze se nazivi atributa te linkovi za sortiranje stupaca. Sortiranje se vrši pozivanjem metode iz klase `DjecaViewSort`, a primjer pripadajućeg programskog koda za atribut matičnog broja djeteta (MBR) nalazi se u nastavku:

```
<a asp-route-sort="1" asp-route-page="@Model.PagingInfo.CurrentPage"
asp-route-ascending="@(Model.PagingInfo.Sort == 1 ?
!Model.PagingInfo.Ascending : true)">
 ...@Html.DisplayNameFor(model => model.Djeca.First().Iddijete)</a>
```

Programski kod 34: Korištenje klase *DjecaViewSort* za sortiranje djece po MBR
(Izvor: vlastita izrada)

Unutar tijela tablice `<tbody>` nalaze se redovi s djecom i njihovim podacima. Prolazi se petljom kroz elemente preko *View* modela poslane liste djece, a primjer pripadajućeg programskog koda nalazi se u nastavku:

```
@foreach (var obj in Model.Djeca)
{
      …
      <tr>
            <td>@obj.Iddijete</td>
            <td>@obj.DijeteIme</td>
            <td>@obj.DijetePrezime</td>
            <td>@obj.Spol</td>
            …
      </tr>
}
```

Programski kod 35: Tijelo tablice s prikazom djece
(Izvor: vlastita izrada)

1.10.3. Filteri

Svi filteri u aplikaciji realizirani su na vrlo sličan način. Za svaku vrstu filtera napravljena je klasa uz pomoć koje se provjerava je li filter postavljen (metoda `IsEmpty`), formatira sadržaj filtera u tip *string* (metoda `ToString`) te čita iz tipa *string* (metoda `FromString`) i pretvara u potrebni tip ovisan o atributu koji se filtrira. Primjer dijela programskog koda spomenute klase za filter popisa djece `FilterDjeca` nalazi se u nastavku (biti će prikazani po jedan primjer za tri tipa elementa u filteru korištena u aplikaciji - u nastavku će filter biti objašnjen prema *select* elementu za skupinu, *input* elementu za ime i/ili prezime i *radio* elementu za spol).

```
public string ImePrezime { get; set; }
public int? Skupina { get; set;}
public string Spol { get;set; }
…
public bool IsEmpty()
{
    bool active = ImePrezime != null
                  || Skupina.HasValue
                  || Spol != null
                  || …;
    return !active;
}
public override string ToString()
{
    return string.Format("{0}-{1}-{2}-…",
        ImePrezime,
        Skupina.ToString(),
        Spol, …);
}
public static FilterDjeca FromString(string s)
{
    var filter = new FilterDjeca();
    if (!string.IsNullOrEmpty(s))
    {
        string[] arr = s.Split('-', StringSplitOptions.None);

        if (arr.Length == 5)
        {
                filter.ImePrezime = arr[0];
                filter.Skupina =
                string.IsNullOrWhiteSpace(arr[1]) ? new int?()
                : int.Parse(arr[1]);
                filter.Spol = arr[2];
                …
        }
    }
    return filter;
}
```

Programski kod 36: Klasa *FilterDjeca*

(Izvor: vlastita izrada)

Forma filtera u *View*-u sadrži *select* element za skupinu (izabrana opcija se sprema u `Filter.Skupina` unutar *ViewModel*-a `DjecaViewModel`), *input* element tipa *radio* za spol (izabrana opcija sprema vrijednost *Ž* ili *M* u `Filter.Spol` unutar *ViewModel*-a `DjecaViewModel`), input element za ime i/ili prezime (upisani tekst se sprema u `Filter.ImePrezima` unutar *ViewModel*-a `DjecaViewModel`). Primjer dijela programskog koda forme za filter u *View*-u nalazi se u nastavku:

```
<form asp-action="Filter" … method="post">
    …
      <label>Filtriranje po skupini:</label>
      <select asp-for="@Model.Filter.Skupina" class="form-control"
      asp-items="ViewBag.GroupNames" onChange="this.form.submit()">
            <option disabled selected value="">Odaberite
            skupinu</option><option asp-
            for="@Model.Filter.Skupina"></option>
      </select>
      …
      <label>Filtriranje po spolu:</label><br />
      …
      <label>Ž:</label><input asp-for="@Model.Filter.Spol"
      type="radio" value="Ž" onChange="this.form.submit()" />
      <label>M:</label><input asp-for="@Model.Filter.Spol"
      type="radio" value="M" onChange="this.form.submit()" />
      …
      <label>Filtriranje po imenu i prezimenu:</label>
      <input asp-for="@Model.Filter.ImePrezime" type="text" />
      …
      <button type="submit" …><b><img src="~/images/filter.png"
      title="Filtriranje" style="height:20px;" /></b></button>
      <a asp-action="Index"><button type="button" …
      onclick="Index"><img src="~/images/nofilter.png" title="Ukloni
      filter" style="height:25px;" /></button></a>
    …
</form>
```

Programski kod 37: Forma za filter popisa djece

(Izvor: vlastita izrada)

Select elementi sadrže opcije prenesene na *View* preko *ViewBag*-a (u ovom konkretnom slučaju `ViewBag.GroupNames` za odgojno-obrazovne skupine). Spomenuti *ViewBag* puni se unutar *get* metode za prikaz popisa djece u kontroleru, a pripadajući kod nalazi se u nastavku:

```
var skupine = _db.Skupines.OrderBy(d => d.SkupinaNaziv).Select(d => new
{ d.Idskupina, d.SkupinaNaziv }).ToList();
ViewBag.GroupNames = new SelectList(skupine,
nameof(Skupine.Idskupina), nameof(Skupine.SkupinaNaziv));
```

Programski kod 38: *ViewBag.GroupNames*

(Izvor: vlastita izrada)

Post metoda, koja prima poslanu formu filtera, prosljeđuje na *get* metodu popisa djece te ako poslani filter ima sadržaj (tj. filter je postavljen), filter se pretvara u *string* i šalje u *get* metodu za popis djece. Programski kod vidljiv je u nastavku:

```
[HttpPost]
public IActionResult Filter(FilterDjeca filter, …)
{
      …
      if (filter == null)
      {
            return RedirectToAction(nameof(Index));
      }
      return RedirectToAction(nameof(Index), new {
                filter = filter.ToString(),
                …
      });
}
```

Programski kod 39: Metoda *Filter*

(Izvor: vlastita izrada)

Poslani *string* filtera obrađuje se u *get* metodi za popis djece tako što se na temelju njega filtrira sami popis djece (`kidsContext`). *String* se pomoću metode `FromString` klase `FilterDjeca` pretvara u stvarne tipove podataka (*int* za skupinu, *string* za spol te ime i prezime) te se popis djece filtrira na način traženja djece zapisane u bazi s filtriranom skupinom i/ili spolom i/ili imenom i/ili prezimenom. Filtrirana lista djece tada se ponovno šalje na *View*. Opisani programski kod nalazi se u nastavku:

```
FilterDjeca f = FilterDjeca.FromString(filter);
if (!f.IsEmpty())
{
      if(f.ImePrezime != ""||f.Skupina.HasValue||f.Spol != ""||…)
      {
            if (f.ImePrezime != "")
            {
                  kidsContext = kidsContext.Where(s =>
                  s.ImePrezime.Contains(f.ImePrezime)||
                  s.PrezimeIme.Contains(f.ImePrezime));
             }
             if (f.Skupina.HasValue)
             {
                  kidsContext = kidsContext.Where(s => s.Idskupina ==
                  f.Skupina);
             }
             if (f.Spol != "")
             {
                  kidsContext = kidsContext.Where(s =>
                  s.Spol.Equals(f.Spol));
             }
      …
```

Programski kod 40: Filtriranje popisa djece

(Izvor: vlastita izrada)

1.10.4. Forme

Sve forme u aplikaciji realizirane su na vrlo sličan način s razlikom u vrsti i/ili tipu elemenata s obzirom na atribute koji se žele dodati ili urediti u bazi. U nastavku će biti objašnjeno kako su realizirane forme za dodavanja, uređivanje i brisanje na primjeru djeteta te forma unutar tablice na primjeru grada.

1.10.4.1. Dodavanje

Forma za dodavanje djeteta prikazuje se pritiskom na gumb za dodavanje (objašnjeno u poglavlju 5.4.1.2. Upravljanje djecom), točnije spomenuti gumb pokreće *get* metodu za dodavanje i uređivanje djeteta (`DodajUredi`). Metoda stvara *ViewBag*-ove za *select* elemente forme kao što je to objašnjeno u spomenutom poglavlju (atributi koji predstavljaju vanjske ključeve na tablice *Grupe*, *Skupine*, *Gradovi* i *Nacionalnosti*). Pošto se radi o metodi koja je korištena i za dodavanje i za uređivanje, ona prima `id` djeteta. Ako je `id` prazan, tada se radi o dodavanju, instancira se novi objekt za model *Djeca* (`new Djeca()`), dodjeljuje mu se novi sljedeći `id` te se šalje na *View* u sklopu `DjecaViewModel`. Spomenuti kod vidljiv je u nastavku:

```
var kidFromDb = new Djeca();
bool dodavanje = false;
var kidViewFromDb = new DjecaView();
var model = new DjecaViewModel();

if(id == null || id == 0)
{
      dodavanje = true;
      Djeca newKid = new Djeca();
      var validateId = _db.Djecas.OrderByDescending(x =>
      x.Iddijete).First().Iddijete;
      newKid.Iddijete = validateId + 1;
      model = new DjecaViewModel
      {
            Dijete = newKid,
            Dodavanje = dodavanje
      };
      …
```

Programski kod 41: *Get* metoda *DodajUredi* za dijete - *Dodavanje*

(Izvor: vlastita izrada)

Svaki od elemenata forme sadrži *tag helper* `asp-for` (npr. `asp-for="@Model.Dijete.Ime"` za djetetovo ime) koji predstavlja atribut kojem će se upisana/izabrana vrijednost pridružiti i poslati u kontroler preko *ModelView*-a. *Select* element za gradove zadrži *autocomplete* značajku koja olakšava pronalaženje željenog grada. Za navedeno potrebno je dodati `data-autocomplete-placeholder`, a kontroler za *autocomplete* sadrži metodu koja je prikazana u nastavku:

```
var query = ctx.Gradovis.Where(a =>
a.GradNaziv.Contains(term)).OrderBy(a => a.GradNaziv).Select(a => new
IdLabel
{
      Id = a.Idgrad,
      Label = a.GradNaziv
});
var list = await query.OrderBy(d => d.Id).ThenBy(d =>
d.Label).Take(appData.AutoCompleteCount).ToListAsync();
return list;
```

Programski kod 42: Metoda za *autocomplete* gradova

(Izvor: vlastita izrada)

Osim spomenutog, forma šalje kontroleru datoteku slike *upload*-anu u *input* element tipa *file*. Za potrebe slanja slike, dodan je *tag helper* `enctype="multipart/form-data"`. Dio programskog koda koji predstavlja formu *post* metode za dodavanje djeteta nalazi se u nastavku:

```
<form id="saveForm" asp-route-id="@Model.Dijete.Iddijete"
      asp-route-dodavanje="@Model.Dodavanje"
      asp-route-filter="@ViewBag.Filter"
      asp-route-page="@ViewBag.Page"
      asp-route-sort="@ViewBag.Sort"
      asp-route-ascending="@ViewBag.Ascending"
      method="post" enctype="multipart/form-data">
            …
            <input asp-for="@Model.Dijete.Iddijete" … />
            …
            <input asp-for="@Model.Dijete.DijeteIme" … />
            …
            <input asp-for="@Model.Dijete.Spol"… type="radio"
            value="Ž" />
            …
            <input asp-for="@Model.Dijete.Spol"… type="radio"
            value="M" />
            …
            <input asp-for="@Model.Dijete.DatumRodjenja" type="date"
            … />
            <select asp-for="@Model.Dijete.Idgrad" class="form-
            control" form="saveForm" asp-items="ViewBag.Cities" data-
            autocomplete-placeholder="grad">
                  <option disabled selected value="">Odaberite
                  grad</option>
                  <option asp-for="@Model.Dijete.Idgrad"></option>
            </select>
            @if (Model.Dijete.SlikaDijete == null ||
            Model.Dijete.SlikaDijete == "")
            {
                  <input type="file" name="slika" form="saveForm" />
            }
            …
```

Programski kod 43: *Post* forma *DodajUredi* za dijete

(Izvor: vlastita izrada)

Post forma šalje se kontroleru njezinim spremanjem pritiskom na gumb za spremanje. Točnije, spomenuti gumb pokreće *post* metodu za dodavanje i uređivanje djeteta (`DodajUredi`). Metoda najprije provjerava je li primljena datoteka slike tipa `IFormFile` (tj. je li dodana slika djeteta). Ako je slika primljena, stvara se direktorij za njezino spremanje, slika se u njega sprema te se informacije o putanji spremljene slike i njezinog naziva spremaju u odgovarajuće atribute novog djeteta. Zatim se dijete dodaje u bazu u obliku objekta `Djeca`. Spomenuti programski kod vidljiv je u nastavku:

```
…
if (slika != null && slika.Length > 0)
{
      string slikaNaziv = slika.FileName;
      var directory = Path.Combine(_environment.WebRootPath,
      "dokumenti", "djeca", kids.Dijete.Oib);
      System.IO.Directory.CreateDirectory(directory);
      var path = Path.Combine(_environment.WebRootPath, "dokumenti",
      "djeca", kids.Dijete.Oib, slikaNaziv);
      using (FileStream stream = new FileStream(path,
      FileMode.Create))
      {
            slika.CopyTo(stream);
            stream.Close();
      }
          kids.Dijete.SlikaDijete = Path.Combine("/dokumenti/",
          "djeca/", kids.Dijete.Oib + "/", slikaNaziv);
          kids.Dijete.SlikaDijeteNaziv = slikaNaziv;
      }
      …
      if (dodavanje)
      {
            _db.Djecas.Add(kids.Dijete);
            _db.SaveChanges();
      }
      …
      return RedirectToAction(nameof(DodajUredi), new {
                  id = kids.Dijete.Iddijete,
                  filter,
                  page,
                  sort,
                  ascending
            });
      …
```

Programski kod 44: *Post* metoda *DodajUredi* za dijete - *Dodavanje*

(Izvor: vlastita izrada)

1.10.4.2. Uređivanje

Forma za uređivanje djeteta prikazuje se pritiskom na gumb za uređivanje (objašnjeno u poglavlju 5.4.1.2. Upravljanje djecom), točnije spomenuti gumb pokreće *get* metodu za dodavanje i uređivanje djeteta (`DodajUredi`). Metoda stvara *ViewBag*-ove za *select* elemente forme kao što je to objašnjeno u spomenutom poglavlju (atributi koji predstavljaju vanjske ključeve na tablice *Grupe*, *Skupine*, *Gradovi* i *Nacionalnosti*). Pošto se radi o metodi koja je

korištena i za dodavanje i za uređivanje, ona prima `id` djeteta. Ako je `id` pun, tada se radi o uređivanju te se prema njemu traži određeno dijete u bazi. Dijete iz baze sprema se u obliku objekta `Djeca` i `DjecaView` te šalje na *View* putem *ViewModel*-a `DjecaViewModel`. Spomenuti programski kod vidljiv je u nastavku:

```
kidFromDb = _db.Djecas.Where(d => d.Iddijete == id).First();
kidViewFromDb = _db.DjecaViews.Where(k => k.Iddijete ==
id).ToList().First();
model = new DjecaViewModel
{Dijete = kidFromDb, DjecaView = kidViewFromDb, … };…
```

Programski kod 45: *Get* metoda *DodajUredi* za dijete - *Uređivanje*

(Izvor: vlastita izrada)

Spomenuti gumb pokreće *post* metodu za dodavanje i uređivanje djeteta (`DodajUredi`). Metoda najprije provjerava je li primljena datoteka slike tipa `IFormFile` (tj. je li uređena slika djeteta) te sprema novu sliku na isti način kao i kod dodavanja djeteta. Zatim se djetetovi podaci u bazi mijenjaju s novim podacima što je prikazano programskim kodom u nastavku:

```
…
_db.Djecas.Update(kids.Dijete);
_db.SaveChanges();
…
```

Programski kod 46: *Post* metoda *DodajUredi* za dijete - *Uređivanje*

(Izvor: vlastita izrada)

1.10.4.3. Brisanje

Forma za brisanje šalje se u kontroler pritiskom na gumb za brisanje, točnije spomenuti gumb pokreće metodu za brisanje djeteta (`Obrisi`) što je zadano *tag helperom* `asp-action` (`asp-action="Obrisi"`). Opisani programski kod vidljiv je u nastavku:

```
<form id="deleteForm" asp-action="Obrisi" method="post" asp-route-
id="@Model.Dijete.Iddijete" asp-route-page="@ViewBag.CurrentPage"
asp-route-sort="@ViewBag.Sort" asp-route
ascending="@ViewBag.Ascending">
</form>
```

Programski kod 47: *Post* forma za brisanje djeteta

(Izvor: vlastita izrada)

Spomenuti gumb najprije pokreće *confirm* poruku želi li se dijete stvarno obrisati. Programski kod gumba za brisanje nalazi se u nastavku:

```
<button type="submit" name="obrisi" form="deleteForm" onclick="return
confirm('Obrisat će se dijete: MBR @Model.Dijete.Iddijete
@Model.Dijete.DijeteIme @Model.Dijete.DijetePrezime')" …><img
src="~/images/delete.png" …/></button>
```

Programski kod 48: Gumb za brisanje djeteta

(Izvor: vlastita izrada)

Nakon potvrde *confirm* poruke za brisanje djeteta, pokreće se *post* metoda za brisanje (`Obrisi`). Metoda prima `id` djeteta te ga prema njemu traži u bazi podataka. Kad se pronađe traženo dijete, ono se briše iz baze što je prikazano programskim kodom u nastavku:

```
public IActionResult Obrisi(int id, bool dodavanje)
{
      var kidsIzDb = _db.Djecas.Find(id);
      _db.Remove(kidsIzDb);
      _db.SaveChanges();
…
```

Programski kod 49: *Post* metoda *Obrisi* za dijete

(Izvor: vlastita izrada)

1.10.4.4. Forme unutar tablice

Forme unutar tablice pojavljuju se kod *Šifrarnika* i *Dokumenata djece* gdje se kod dodavanja red direktno dodaje u tablicu na ekranu s *form* elementima, kod uređivanja u redu s podacima koji se žele uređivati polja se pretvaraju u *form* elemente, dok brisanje funkcionira na isti način kao i kod do sad opisanih formi. Za vizualno dodavanje redova i mijenjanje redova u drugačiji način prikaza, korišten je *javascript* koji se poziva *onClick* metodom. U nastavku će biti objašnjeno kako su realizirane forme unutar tablice za dodavanja i uređivanje na primjeru grada.

Za potrebe dodavanja grada u tablicu dodan je skriveni (*hidden*) red s *input* elementima za upis naziva grada i njegovog poštanskog broja. Programski kod vidljiv je u nastavku:

```
<tr id="trDodaj" hidden>
      <td>
            <input asp-for="Grad.GradNaziv" form="dodavanjeForm" />
      </td>
      <td>
            <input asp-for="Grad.PostBr" form="dodavanjeForm" />
      </td>
      <td>
            <form id="dodavanjeForm" asp-action="GradoviPopis"
```

```
            … method="post">
                  <input type="hidden" asp-for="@Model.Grad.Idgrad"
                  value="0" />
                   <button class="button-primary" type="submit" …><img
                  src="~/images/save.png" … /></button>
            </form>
       </td>
            …
```

Programski kod 50: *Post* forma za dodavanje grada

(Izvor: vlastita izrada)

Skriveni red se otkriva pokretanjem dodavanja klikom na gumb za dodavanje. Tada se pokreće *javascript* skripta koja miče *hidden* atribut sa skrivenog retka. *Javascript* programski kod vidljiv je u nastavku:

```
function dodavanjeMode()
{document.getElementById('trDodaj').removeAttribute('hidden');}
```

Programski kod 51: Skripta za prikaz reda za dodavanja grada

(Izvor: vlastita izrada)

Za potrebe uređivanja grada u tablici su za svaki atribut dodana dva polja od kojih je jedan prikazan i jedan skriven (*hidden*). Programski kod vidljiv je u nastavku:

```
<td id="@obj.Idgrad tdDisplay">@obj.GradNaziv)</td>
<td id="@obj.Idgrad tdInput" hidden>
      <input asp-for="@Model.Grad.GradNaziv" … form="@obj.Idgrad
      saveForm" value="@obj.GradNaziv" />
</td>
<td id="@obj.Idgrad tdDisplayPB">@obj.PostBr)</td>
<td id="@obj.Idgrad tdInputPB" hidden>
      <input asp-for="@Model.Grad.PostBr" … form="@obj.Idgrad
      saveForm" value="@obj.PostBr" />
</td>
<td>
      <button … id="@obj.Idgrad uredjivanje"
      onclick="uredjivanjeMode(@obj.Idgrad)" …><img
      src="~/images/edit.png" … /></button>
      <form id="@obj.Idgrad saveForm"
       … method="post">
            <input type="hidden" asp-for="@Model.Grad.Idgrad"
            value="@obj.Idgrad" />
             <button … type="submit" id="@obj.Idgrad spremanje" hidden
            onclick="displayMode()" …><img src="~/images/save.png" …
            /></button>
     </form>
      …
```

Programski kod 52: *Post* forma za uređivanje grada

(Izvor: vlastita izrada)

Pritiskom na gumb za uređivanje pokreće se *javascript* skripta s funkcijom `uredjivanjeMode()` koja prikazuje do sad skrivena polja, a skriva do sad prikazana polja. Također, skriva gumb za uređivanje i otkriva gumbe za spremanje i brisanje. *Javascript* programski kod vidljiv je u nastavku:

```
function uredjivanjeMode(id) {
      document.getElementById(id + '
      spremanje').removeAttribute('hidden');

      document.getElementById(id + '
      brisanje').removeAttribute('hidden');

      document.getElementById(id + '
      uredjivanje').setAttribute('hidden', 'hidden');

      document.getElementById(id + '
      tdInput').removeAttribute('hidden');

      document.getElementById(id + '
      tdDisplay').setAttribute('hidden', 'hidden');

      document.getElementById(id + '
      tdInputPB').removeAttribute('hidden');

      document.getElementById(id + '
      tdDisplayPB').setAttribute('hidden', 'hidden');
}
```

Programski kod 53: Skripta za pokretanja načina za uređivanje reda

(Izvor: vlastita izrada)

Pritiskom na gumb za spremanje pokreće se *javascript* skripta s funkcijom `displayMode()` koja zatvara način za uređivanje istim principom skrivanja i otkrivanja polja i gumbi kao kod funkcije `uredjivanjeMode()`. *Javascript* programski kod vidljiv je u nastavku:

```
function displayMode() {

      document.getElementById('spremanje').setAttribute('hidden',
      'hidden');

      document.getElementById('brisanje').setAttribute('hidden',
      'hidden');

      document.getElementById('uredjivanje').removeAttribute('hidden'
      );

      document.getElementById('tdInput').setAttribute('hidden',
      'hidden');

      document.getElementById('tdDisplay').removeAttribute('hidden');
```

```
document.getElementById('tdInputPB').setAttribute('hidden',
'hidden');

document.getElementById('tdDisplayPB').removeAttribute('hidden'
);}
```

Programski kod 54: Skripta za zatvaranje načina za uređivanje reda

(Izvor: vlastita izrada)

1.10.5. Generiranje PDF-a

U aplikaciji realizirano je generiranje 11 vrsti PDF izvještaja: *Evidencija ozljeda, Matična knjiga, Imenik djece, Evidencija nazočnosti, Antropometrijska mjerenja, Evidencija sukoba, Individualni sastanak, Sanitarni nadzor, Popis djece s alergijama, Program stručnog usavršavanja* i *Zdravstveni karton*. Za potrebe generiranja PDF-a, instalirani su *MS Visual Studio* paketi *PDFSharpCore*, *PDFsharp-MicroDoc* i *Polybioz*.*HtmlRender*.*PdfSharp*.Core. Način generiranja PDF izvještaja biti će objašnjen u nastavku na primjeru matične knjige djece.

Ono što će biti prikazano u PDF-u obrađeno je i pridruženo *html* sadržaju (`htmlcontent`). Konkretno za matičnu knjigu, *foreach* petljom prolazi se popisom djece te se ispisuju njihovi podaci unutar `<table>` *html* elementa. Opisani programski kod nalazi se u nastavku:

```
string htmlcontent = "<div style='width:100%; text-align:center'>";
…
htmlcontent += "<table style='text-align:left;width:100%;font-
size:10px'>";
foreach (var dijete in djeca){
      htmlcontent += "<style type='text/css'> table { page-break-
      inside: avoid; } </style>";
      htmlcontent += "<tr>";
      htmlcontent += "<td style='padding:5px;border:0.1px solid
      #000;width:30px;height:50px'>" + dijete.Iddijete + "</td>";
      htmlcontent += "<td style='padding:5px;border:0.1px solid
      #000;width:100px;'>" + dijete.ImePrezime + "</td>";
      htmlcontent += "<td style='padding:5px;border:0.1px solid
      #000;width:100px;'>" +
      dijete.DatumRodjenjaDijete.Value.ToString("dd.MM.yyyy.") + " "
      + dijete.MjestoRodjenja + "</td>";
      …
```

Programski kod 55: Formiranje *html* sadržaja

(Izvor: vlastita izrada)

Postavke generiranog PDF-a poput orijentacije (*Portrait* ili *Landscape*), veličine stranice te margina postavljaju se u obliku `PdfGenerateConfig` modela. Programski kod za spomenuto te za formiranje naziva PDF-a i njegovo generiranje nalazi se u nastavku:

```
filename = "002_" +
int.Parse(User.FindFirst(ClaimTypes.NameIdentifier).Value).ToString("
D3") + ".pdf";
var config = new PdfGenerateConfig();
config.PageOrientation =
(PdfSharpCore.PageOrientation)PageOrientation.Landscape;
config.PageSize = (PdfSharpCore.PageSize)PageSize.A4;
config.MarginTop = 93;
config.MarginBottom = 35;
config.MarginLeft = 45;
config.MarginRight = 40;
pdf = PdfGenerator.GeneratePdf(htmlcontent, config);
```

Programski kod 56: Postavke i generiranje PDF-a
(Izvor: vlastita izrada)

Pošto traženi ispis matične knjige zahtjeva više stranica prikaza tablice djece s podacima, potrebno je na svakoj stranici dodati zaglavlje tablice te podnožje s informacijom o broju stranice, tipu izvještaja te datumom ispisa. Kako bi se navedeno uvijek nalazilo na istim mjestima svake stranice, koristi se `XGraphic` za „crtanje“ na već generiranom PDF-u.

Najprije se crtaju oblici (za potrebe zaglavlja i podnožja) uz pomoć `XRect` (prima poziciju x-osi, y-osi, širinu i visinu) te `XGraphic DrawRectangle` funkcija. Zatim se upisuje željeni *string* tekst unutar nacrtanih oblika uz pomoć `XTextFormatter DrawString` funkcije. Programski kod za opisane radnje nalazi se u nastavku:

```
XFont font = new XFont("Calibri", 10);
XFont fontPodnozje = new XFont("Calibri", 8);
XBrush brush = XBrushes.Black;
int noPages = pdf.Pages.Count - 1;
for (int i = 1; i < pdf.Pages.Count; ++i)
{
      PdfPage pdfPage = pdf.Pages[i];
      XRect layoutRectangle1 = new XRect(47.5, font.Height * 4,
      pdfPage.Width - 813, font.Height * 3);
      XRect layoutRectangle2 = new XRect(79.5, font.Height * 4,
      pdfPage.Width - 743, font.Height * 3);
      …
      using (XGraphics gfx = XGraphics.FromPdfPage(pdfPage))
      {
```

```
gfx.DrawRectangle(XPens.LightGray, XBrushes.LightGray,
layoutRectangle1);
gfx.DrawRectangle(XPens.LightGray, XBrushes.LightGray,
layoutRectangle2);
…
XTextFormatter textFormatter = new XTextFormatter(gfx);
textFormatter.DrawString("MBR", font, brush,
layoutRectangle1,   XStringFormats.TopLeft);
textFormatter.DrawString("Ime i prezime", font, brush,
layoutRectangle2, XStringFormats.TopLeft);
textFormatter.DrawString("Dan, mjesec, godina i mjesto
rođenja", font, brush, layoutRectangle3,
XStringFormats.TopLeft);
…
XRect layoutRectangle10 = new XRect(-40, pdfPage.Height -
fontPodnozje.Height * 6, pdfPage.Width,
fontPodnozje.Height);
…
gfx.DrawString("Stranica " + (i).ToString() + " od " +
noPages,
fontPodnozje, brush, layoutRectangle10,
XStringFormats.BottomRight);
…
```

Programski kod 57: Crtanje zaglavlja i podnožja PDF-a

(Izvor: vlastita izrada)

Za spremanje generiranog PDF-a u direktorij i njegovo otvaranje koristi se `MemoryStream` i `Process`. Programski kod nalazi se u nastavku:

```
byte[]? response = null;
using (MemoryStream ms = new MemoryStream())
{
      pdf.Save(ms);
      response = ms.ToArray();
}
File(response, "application/pdf", filename);
pdf.Save(_environment.WebRootPath + "\\printing\\" + filename);
var p = new Process();
p.StartInfo   =   new   ProcessStartInfo(_environment.WebRootPath   +
"\\printing\\" + filename)
{
      UseShellExecute = true
};
p.Start();
```

Programski kod 58: Spremanje i otvaranje PDF-a

(Izvor: vlastita izrada)

Zaključak

Danas sve više struka teži digitalizaciji i automatizaciji svojeg poslovanja, dječji vrtići ni u tome nisu iznimka. Dječji vrtići, pošto rade s djecom, nastoje što više vremena i energije usmjeriti k svakom pojedinom djetetu. Ponajviše je važno, da ako dođe do incidenta bilo koje vrste (sukobi roditelja, sukobi djece, ozljeda djeteta, alergijska reakcija, tvrdnja da dijete nije bilo nazočno kada je i sl.), imati dokumentaciju i svojevrsnu potvrdu o nastaloj situaciji. Sve u svemu, čuvanje svih podataka na jednom mjestu te lakoća rada s njima prioriteti su koji jedan dobro izrađeni ERP sustav može zadovoljiti.

Najbitnije značajke za dobro razvijen ERP sustav za dječje vrtiće su: dostupnost aplikacije, zadovoljavanje potreba i zahtjeva klijenata, funkcionalnosti koje prate zakonski određene radnje te obrasce propisane za dječje vrtiće, jednostavnost korištenja i skalabilnost.

U sklopu ovog diplomskog rada razvijen je jedan dio aplikacije za ERP sustav za dječje vrtiće imajući na umu sve bitne značajke izdvojene u teorijskom dijelu rada i funkcionalnosti analizirane kod već postojećeg ERP sustava za dječje vrtiće Next ERP Preschool. Pošto je sama aplikacija rađena u suradnji s jednim privatnim vrtićem u Varaždinskoj županiji, direktni kontakt sa strukom bio je ključan faktor u kvalitetnoj izradi te u konačnici stvaranju efikasnog i pristupačnog proizvoda.

Na kraju je zaključak vrlo jasan, ERP sustavi su sadašnjost i budućnost poslovanja, a dječji vrtići u Hrvatskoj više nego ikad trebaju ERP, kako administracija ne bi oduzimala vrijeme njihovom najbitnijem poslu, a to je rad s djecom.

Popis literature

[1] “About draw.io.” https://www.drawio.com/about (accessed May 19, 2023).

[2] “What is Microsoft SQL Server? A definition from WhatIs.com,” *Data Management*. https://www.techtarget.com/searchdatamanagement/definition/SQL-Server (accessed May 19, 2023).

[3] markingmyname, “SQL Server Management Studio (SSMS) - SQL Server Management Studio (SSMS),” Mar. 31, 2023. https://learn.microsoft.com/en-us/sql/ssms/sql-server-management-studio-ssms (accessed May 19, 2023).

[4] A. B. Santos, “Microsoft Visual Studio: Everything You Need to Know,” *SoftwareKeep*. https://softwarekeep.com/help-center/what-is-microsoft-visual-studio-where-can-i-download-it (accessed May 19, 2023).

[5] “ASP.NET Core MVC: Tutorial | Simplilearn,” *Simplilearn.com*. https://www.simplilearn.com/tutorials/asp-dot-net-tutorial/asp-net-core-mvc (accessed May 19, 2023).

[6] https://www.facebook.com/Investopedia, “Enterprise Resource Planning (ERP): Meaning, Components, and Examples,” *Investopedia*. https://www.investopedia.com/terms/e/erp.asp (accessed May 22, 2023).

[7] D. Insights, “Selecting the Best ERP System: What You Need to Know to Make the Right Decision,” *ArcherPoint*, Jan. 17, 2022. https://archerpoint.com/selecting-the-best-erp-system-what-you-need-to-know-to-make-the-right-decision/ (accessed May 22, 2023).

[8] J. Desai, “Web Application Vs Desktop Application: Pros and Cons,” *POSITIWISE*, Jan. 12, 2022. https://positiwise.com/blog/web-application-vs-desktop-application-pros-and-cons (accessed May 22, 2023).

[9] “Which is better, a desktop app based ERP solution or a web app based ERP solution?,” *Quora*. https://www.quora.com/Which-is-better-a-desktop-app-based-ERP-solution-or-a-web-app-based-ERP-solution (accessed May 22, 2023).

[10] S. Firs, “TOP 5 najboljih poslovnih ERP sustava,” *Novi jednostavni ERP*, May 01, 2020. https://www.newsaperp.com/hr/blog-542-top-5-najboljih-poslovnih-erp-sustava (accessed Jun. 03, 2023).

[11] “What is SAP ERP?,” *SearchSAP*. https://www.techtarget.com/searchsap/definition/SAP (accessed Jun. 03, 2023).

[12] “Oracle ERP,” *Terillium*. https://terillium.com/oracle-erp/ (accessed Jun. 03, 2023).

[13] https://www.facebook.com/Tipalti, “What is Workday ERP? - Integration, Pricing, Features, & More | Tipalti,” Oct. 07, 2020. https://tipalti.com/erp-integrations/workday-erp/ (accessed Jun. 03, 2023).

[14]“Sage Murano es ahora Sage 200 | Sage España,” *sage-murano*. https://www.sage.com/es-es/productos/sage-murano/ (accessed Jun. 03, 2023).
[15]https://www.facebook.com/Tipalti, “What is Infor ERP & How Much Does it Cost? | Tipalti,” Sep. 25, 2020. https://tipalti.com/erp-integrations/infor-erp/ (accessed Jun. 03, 2023).
[16]“8 Step Checklist for Selecting the Correct ERP Software for Schools.” https://openeducat.org/blog/our-blog-1/post/8-step-checklist-for-selecting-the-correct-erp-software-for-schools-55 (accessed May 22, 2023).
[17]“Top 6 Kindergarten Management System you should Know,” *Viindoo Technology Joint Stock Company*, Jan. 27, 2023. https://viindoo.com/blog/business-management-3/kindergarten-management-system-611 (accessed May 23, 2023).
[18]NextERP, “Preschool ERP, Free Preschool Management Software (2023).” https://www.nexterp.in/preschool-erp-software.html (accessed May 24, 2023).
[19]“Vodič za SQL Server – Praktični vodič za SQL Server,” *SQL Server Tutorial*. https://www.sqlservertutorial.net/ (accessed May 23, 2023).
[20]“What is SQL Server,” *SQL Server Tutorial*. https://www.sqlservertutorial.net/getting-started/what-is-sql-server/ (accessed May 23, 2023).
[21]“How To Install SQL Server Developer Edition,” *SQL Server Tutorial*. https://www.sqlservertutorial.net/install-sql-server/ (accessed May 23, 2023).
[22]“ASP.NET Core: What Is It?,” *Under The Hood Learning*, Nov. 24, 2021. https://underthehoodlearning.com/asp-net-core-what-is-it/ (accessed May 23, 2023).
[23]“What Kind of Applications Can You Build Using Microsoft .NET Core?” https://www.cisin.com/coffee-break/technology/what-kind-of-applications-can-you-build-using-microsoft-net-core.html (accessed May 23, 2023).
[24]anandmeg, “Tutorial: Create a C# ASP.NET Core web app in Visual Studio - Visual Studio (Windows),” Feb. 24, 2023. https://learn.microsoft.com/en-us/visualstudio/get-started/csharp/tutorial-aspnet-core (accessed May 19, 2023).
[25]“ASP.NET Core Connection String | Overview & applications with Example,” *EDUCBA*, Jun. 05, 2022. https://www.educba.com/asp-dot-net-core-connection-string/ (accessed May 19, 2023).
[26]“Generating a model from an existing database.” https://www.learnentityframeworkcore.com/walkthroughs/existing-database (accessed May 19, 2023).
[27]M. Maleković and K. Rabuzin, Uvod u baze podataka. Varaždin: Fakultet organizacije i informatike, Sveučilište u Zagrebu, 2016.
[28]K. Rabuzin, SQL: napredne teme. Varaždin : Fakultet organizacije i informatike. 2014.

Popis slika

Slika 1. Moduli ERP sustava....3
Slika 2. Funkcionalnost izvješćivanja o incidentima - Next ERP Preschool....7
Slika 3. Funkcionalnost evidencije dopuštenja roditelja - Next ERP Preschool....8
Slika 4. Funkcionalnost podrške za hibridnu učionicu - Next ERP Preschool....8
Slika 5. Povezivanje na MS SQL Server....9
Slika 6. Create a new project....10
Slika 7. Configure a new project....11
Slika 8. Additional information....11
Slika 9. *Packages*....12
Slika 10. ERA model *Korisnici aplikacije*....15
Slika 11. ERA model *Šifrarnici*....16
Slika 12. ERA model *Djeca*....18
Slika 13. ERA model *Sanitarni nadzor*....20
Slika 14. ERA model *Nazočnost djece*....21
Slika 15. ERA model *Sukobi među djecom*....22
Slika 16. ERA model Sukobi među roditeljima....23
Slika 17. ERA model *Ozljede djece*....24
Slika 18. ERA model Antropometrijska mjerenja djece....25
Slika 19. ERA model *Alergija djece*....26
Slika 20. ERA model Individualni sastanci....26
Slika 21. ERA model *Dokumenti djece*....27
Slika 22. ERA model Program stručnog usavršavanja odgojitelja....28
Slika 23. KorisniciView....30
Slika 24. Ispis SQL skripte *KorisniciView*....31
Slika 25. Ispis SQL skripte *Odgojitelji*....31
Slika 26. SifrarniciView....31
Slika 27. *DjecaView*....32
Slika 28. Ispis SQL skripte *DjecaView*....34
Slika 29. SanitarniNadzorView....34
Slika 30. Ispis SQL skripte *SanitarniNadzorView*....35
Slika 31. NazocnostiView....35
Slika 32. Ispis SQL skripte *NazocnostiView*....36
Slika 33. Ispis SQL skripte *NazocnostiZadnjiDatum*....36
Slika 34. *SukobiDjecaView*....37
Slika 35. Ispis SQL skripte *SukobiDjecaView*....38
Slika 36. *SukobiRoditeljiView*....38
Slika 37. Ispis SQL skripte *SukobiRoditeljiView*....39

Slika 38. *OzljedeView*....39
Slika 39. Ispis SQL skripte *SukobiRoditeljiView*....40
Slika 40. *AntropMjerView* 41
Slika 41. Ispis SQL skripte *AntropMjerView*....41
Slika 42. Ispis SQL skripte *AntropMjerPopisView*....42
Slika 43. *AlergijeView*....42
Slika 44. Ispis SQL skripte *AntropMjerPopisView*....43
Slika 45. *IndividualniSastanciView*....43
Slika 46. Ispis SQL skripte *IndividualniSastanciView*....44
Slika 47. *DjecaDokumentiView*....45
Slika 48. Ispis SQL skripte *DjecaDokumentiView* 45
Slika 49. *ProgStrucUsavView*....46
Slika 50. Ispis SQL skripte *ProgStrucUsavView* 47
Slika 51. Ispis SQL skripte *ProgStrucUsavView* 47
Slika 52. Prijava u aplikaciju 48
Slika 53. Početna stranica aplikacije....48
Slika 54. Gumbi....49
Slika 55. Popis korisnika....50
Slika 56. Dodavanje/uređivanje korisnika 50
Slika 57. Popis djece 51
Slika 58. Dodavanje/uređivanje djeteta - *Podaci o djetetu* 52
Slika 59. Dodavanje/uređivanje djeteta - *Podaci o roditeljima*....52
Slika 60. Dodavanje/uređivanje djeteta - *Podaci o zdravlju*....53
Slika 61. Upravljanje skupinama....53
Slika 62. Upravljanje vrstama dokumenata....54
Slika 63. Upravljanje nacionalnostima 54
Slika 64. Upravljanje neradnim danima 54
Slika 65. Upravljanje gradovima 55
Slika 66. Upravljanje pedagoškim godinama 55
Slika 67. Izbornik za korisničku ulogu *Ravnatelj* 56
Slika 68. Matična knjiga djece 56
Slika 69. Detalji matične knjige za dijete....57
Slika 70. Ispis matične knjige....57
Slika 71. Upravljanje skupinama djece 58
Slika 72. Popis sanitarnih nadzora....58
Slika 73. Ispis sanitarnog nadzora....59
Slika 74. Dodavanje/uređivanje sanitarnih nadzora 60
Slika 75. Izbornik za korisničku ulogu *Odgojitelj* 60
Slika 76. Imenik djece....61

Slika 77. Ispis imenika djece....61
Slika 78. Detalji imenika djece za dijete....62
Slika 79. Ispis evidencije nazočnosti djece....62
Slika 80. Popis evidencija nazočnosti djece - Slučaj bez evidencije trenutnog dana....63
Slika 81. Popis evidencija nazočnosti djece - Slučaj s evidencijom trenutnog dana....63
Slika 82. Izbor vrste sukoba za pregled evidencija sukoba....64
Slika 83. Pregled sukoba među djecom....64
Slika 84. Ispis evidencije sukoba među djecom....65
Slika 85. Dodavanje/uređivanje evidencije sukoba među djecom....66
Slika 86. Pregled sukoba među roditeljima....66
Slika 87. Ispis evidencije sukoba među roditeljima....67
Slika 88. Dodavanje/uređivanje evidencije sukoba među roditeljima....68
Slika 89. Pregled ozljeda djece....68
Slika 90. Ispis evidencije ozljede djeteta....69
Slika 91. Dodavanje ozljede djeteta - 1.dio....70
Slika 92. Dodavanje ozljede djeteta - 2.dio....70
Slika 93. Detalji/uređivanje ozljede djeteta....70
Slika 94. Pregled djece....71
Slika 95. Ispis zdravstvenog kartona....71
Slika 96. Pregled antropometrijskih mjerenja djece....72
Slika 97. Ispis antropometrijskog mjerenja....72
Slika 98. Dodavanje/uređivanje antropometrijskog mjerenja....73
Slika 99. Pregled alergija djece....73
Slika 100. Ispis alergija djece....74
Slika 101. Dodavanje alergije djeteta - 1.dio....74
Slika 102. Dodavanje alergije djeteta - 2. dio i uređivanje alergije djeteta....75
Slika 103. Pregled individualnih sastanaka....75
Slika 104. Ispis individualnog sastanka....76
Slika 105. Dodavanje individualnog sastanka - 1.dio....77
Slika 106. Dodavanje individualnog sastanka - 2.dio....77
Slika 107. Uređivanje individualnog sastanka....77
Slika 108. Popis programa za stručno usavršavanje odgojitelja....78
Slika 109. Ispis programa stručnog usavršavanja....78
Slika 110. Dodavanje/uređivanje programa stručnog usavršavanja....79
Slika 111. Dodavanje/uređivanje teme programa stručnog usavršavanja....80
Slika 112. Brzi pregled djeteta - *Izbor djeteta*....80
Slika 113. Brzi pregled djeteta - *Kontakti*....80
Slika 114. Brzi pregled djeteta - *Imenik*....81
Slika 115. Brzi pregled djeteta - *Evidencije nazočnosti*....81

Slika 116. Brzi pregled djeteta - *Evidencije sukoba* 82
Slika 117. Brzi pregled djeteta - *Evidencije ozljeda*.....82
Slika 118. Brzi pregled djeteta – *Zdravstveni karton* 82
Slika 119. Brzi pregled djeteta – *Antropometrijska mjerenja* 83
Slika 120. Brzi pregled djeteta – *Alergije* 83
Slika 121. Brzi pregled djeteta – *Individualni sastanci*.....83
Slika 122. Brzi pregled djeteta – *Dokumenti* 84

Popis programskih kodova

Programski kod 1: *Appsettings.json* - Povezivanje s bazom podataka....12
Programski kod 2: *Program.cs* - Povezivanje s bazom podataka....12
Programski kod 3: Migracija baze podataka....13
Programski kod 4: Model *Alergije*....13
Programski kod 5. SQL skripta *KorisniciView*....30
Programski kod 6. SQL skripta *Odgojitelji*....31
Programski kod 7. SQL skripta *DjecaView*....34
Programski kod 8. SQL skripta *SanitarniNadzorView*....35
Programski kod 9. SQL skripta *NazocnostiView*....36
Programski kod 10. SQL skripta *NazocnostiZadnjiDatum*....36
Programski kod 11. SQL skripta *SukobiDjecaView*....37
Programski kod 12. SQL skripta *SukobiRoditeljiView*....39
Programski kod 13. SQL skripta funkcije *FormatDD_MM_YYYY*....40
Programski kod 14. SQL skripta *OzljedeView*....40
Programski kod 15. SQL skripta *AntropMjerView*....41
Programski kod 16. SQL skripta *AntropMjerPopisView*....42
Programski kod 17. SQL skripta *AlergijeView*....43
Programski kod 18. SQL skripta *IndividualniSastanciView*....44
Programski kod 19. SQL skripta *DjecaDokumentiView*....45
Programski kod 20. SQL skripta *ProgStrucUsavView*....47
Programski kod 21. SQL skripta *ProgStrucUsavPopisView*....47
Programski kod 22: *OnConfiguring* metoda u *DVContext*....85
Programski kod 23: *OnModelCreating* metoda u *DVContext*....86
Programski kod 24: *View* za ispis *error* poruke - *_Notification*....87
Programski kod 25: *Get* metoda za prijavu....87
Programski kod 26: *Post* metoda za prijavu....88
Programski kod 27: Dodavanje autentifikacije u *Program.cs*....88
Programski kod 28: Metoda za odjavu....89
Programski kod 29: Autorizacija....89
Programski kod 30: *Get* metoda za prikaz tablice djece....89
Programski kod 31: *DjecaViewSort*....90
Programski kod 32: Primanje *ViewModel*-a....90
Programski kod 33: Prikaz *pager* elementa....91
Programski kod 34: Korištenje klase *DjecaViewSort* za sortiranje djece po MBR....91
Programski kod 35: Tijelo tablice s prikazom djece....91

Programski kod 36: Klasa *FilterDjeca* 92
Programski kod 37: Forma za filter popisa djece 93
Programski kod 38: *ViewBag.GroupNames* 93
Programski kod 39: Metoda *Filter* 94
Programski kod 40: Filtriranje popisa djece 94
Programski kod 41: *Get* metoda *DodajUredi* za dijete - *Dodavanje* 95
Programski kod 42: Metoda za *autocomplete* gradova 96
Programski kod 43: *Post* forma *DodajUredi* za dijete 96
Programski kod 44: *Post* metoda *DodajUredi* za dijete - *Dodavanje* 97
Programski kod 45: *Get* metoda *DodajUredi* za dijete - *Uređivanje* 98
Programski kod 46: *Post* metoda *DodajUredi* za dijete - *Uređivanje* 98
Programski kod 47: *Post* forma za brisanje djeteta 98
Programski kod 48: Gumb za brisanje djeteta 99
Programski kod 49: *Post* metoda *Obrisi* za dijete 99
Programski kod 50: *Post* forma za dodavanje grada 100
Programski kod 51: Skripta za prikaz reda za dodavanja grada 100
Programski kod 52: *Post* forma za uređivanje grada 100
Programski kod 53: Skripta za pokretanja načina za uređivanje reda 101
Programski kod 54: Skripta za zatvaranje načina za uređivanje reda 102
Programski kod 55: Formiranje *html* sadržaja 102
Programski kod 56: Postavke i generiranje PDF-a 103
Programski kod 57: Crtanje zaglavlja i podnožja PDF-a 104
Programski kod 58: Spremanje i otvaranje PDF-a 104