A heart and cross logo

Description automatically generated

Dokumentacija Projekta:

*Softverski Sistem za Manju Kliniku*

Radila:Ena Omanović

Mostar, 2024.g.

Sadržaj

[1. Uvod 3](#_Toc164731154)

[1.1. Svrha dokumenta 3](#_Toc164731155)

[1.2. Opis projekta 3](#_Toc164731156)

[1.3. Ciljevi projekta 3](#_Toc164731157)

[2. Analiza Sistemskih Zahtjeva 3](#_Toc164731158)

[2.1. Opis funkcionalnih zahtjeva 3](#_Toc164731159)

[2.2. Uspostavljanje kriterija sistema 4](#_Toc164731160)

[3. Arhitektura Sistema 4](#_Toc164731161)

[3.1. Arhitekturni dizajn 4](#_Toc164731162)

[3.2. Tehnologije korištene u implementaciji 5](#_Toc164731163)

[3.3. Struktura baze podataka 5](#_Toc164731164)

[4. Implementacija 5](#_Toc164731165)

[4.1. Detaljan opis implementacije sistema 5](#_Toc164731166)

[4.2. Opis ključnih modula i funkcionalnosti 6](#_Toc164731167)

[4.3. Testiranje sistema 6](#_Toc164731168)

[5. Korisnički Uputstva 7](#_Toc164731169)

[5.1. Uputstva za korištenje sistema 7](#_Toc164731170)

[5.2. Uputstva za ljekare 10](#_Toc164731171)

[5.3. Uputstva za razvojni tim 11](#_Toc164731172)

[6. Tehnička Dokumentacija 11](#_Toc164731173)

[6.1. Opis ključnih funkcija i klasa 11](#_Toc164731174)

[6.3. Struktura baze podataka 11](#_Toc164731175)

[7. Rezultati i Zaključci 12](#_Toc164731176)

[7.1. Analiza postignutih rezultata 12](#_Toc164731177)

[7.2. Procjena uspješnosti projekta 12](#_Toc164731178)

[7.3. Predlozi za buduće unapređenje 13](#_Toc164731179)

# 1. Uvod

## 1.1. Svrha dokumenta

Ova dokumentacija ima za svrhu pružiti detaljan pregled implementiranog softverskog sistema za manju kliniku. Kroz ovaj dokument, čitatelji će dobiti uvid u svaki aspekt projekta, uključujući analizu zahtjeva, arhitekturu sistema, proces implementacije, kao i uputstva za korištenje. Također, dokumentacija će poslužiti kao resurs za održavanje sistema i podršku korisnicima i razvojnom timu.

## 1.2. Opis projekta

Softverski sistem za manju kliniku predstavlja integriranu platformu koja omogućava evidenciju, pregled i pisanje nalaza za pacijente koji su registrovani u klinici. Sistem omogućava administraciju podataka o pacijentima i ljekarima, evidenciju prijema pacijenata, kao i kreiranje nalaza za svaki prijem. Cilj ovog projekta je olakšati rad osoblju klinike pružajući im efikasan alat za upravljanje pacijentima i njihovim zdravstvenim podacima.

## 1.3. Ciljevi projekta

Glavni ciljevi projekta su:

* Razviti softverski sistem koji će omogućiti administraciju podataka o pacijentima i ljekarima.
* Implementirati funkcionalnosti za evidenciju prijema pacijenata sa detaljnim informacijama o datumu, vremenu, pacijentu i nadležnom ljekaru.
* Omogućiti kreiranje nalaza za svaki prijem sa tekstualnim opisom i automatskim dodjeljivanjem datuma i vremena.
* Osigurati korisnički interfejs koji će biti intuitivan za korištenje i koji će podržati osnovne operacije kao što su unos, izmjena i brisanje podataka.
* Pružiti mogućnost generisanja PDF dokumenata sa strukturiranim podacima o prijemu pacijenta i napisanom nalazu.

Ovim ciljevima teži se unapređenju efikasnosti i preciznosti rada u klinici, kao i poboljšanju kvaliteta usluge pružene pacijentima.

# 2. Analiza Sistemskih Zahtjeva

## 2.1. Opis funkcionalnih zahtjeva

Softverski sistem za manju kliniku treba da podržava sljedeće funkcionalnosti:

1. Evidenciju pacijenata:

* Omogućiti unos, izmjenu i brisanje podataka o pacijentima, uključujući ime, prezime, datum rođenja, spol, adresu i broj telefona.

1. Evidenciju ljekara:

* Omogućiti unos, izmjenu i brisanje podataka o ljekarima, uključujući ime, prezime, titulu i šifru.

1. Evidenciju prijema pacijenata:

* Omogućiti unos podataka o prijemu pacijenata, uključujući datum i vrijeme prijema, pacijenta, nadležnog ljekara i označiti hitni prijem.

1. Kreiranje nalaza:

* Omogućiti kreiranje nalaza za svaki prijem pacijenta sa tekstualnim opisom i automatskim dodjeljivanjem datuma i vremena.

1. Pregled evidencija:

* Omogućiti pregled liste prijema pacijenata sa osnovnim informacijama, kao što su ime pacijenta, datum i vrijeme prijema, ime i prezime nadležnog ljekara, te akcije za izmjenu i otkazivanje prijema.

1. Filtriranje prikaza:

* Omogućiti filtriranje liste prijema po datumu prijema u formatu "od - do" radi preciznog pregleda.

1. Odabir specijalista:

* Omogućiti odabir samo specijalista prilikom evidentiranja prijema pacijenata.

## 2.2. Uspostavljanje kriterija sistema

Kriteriji sistema koji će biti uspostavljeni obuhvataju:

* Validacija datuma i vremena prijema: Sistem će osigurati da datum i vrijeme prijema ne mogu biti uneseni iz prošlosti, pri čemu će minimalni datum i vrijeme prijema biti današnji datum.
* Pristup samo specijalistima: Prilikom evidentiranja prijema pacijenata, sistem će omogućiti odabir samo ljekara koji su specijalisti.
* Indikator hitnog prijema: Sistem će prikazati indikator koji označava da se radi o hitnom prijemu.
* Automatsko dodjeljivanje datuma i vremena nalaza: Datum i vrijeme nalaza će se sistemski dodjeljivati prilikom kreiranja nalaza, a neće biti uneseni od strane korisnika.
* Generisanje PDF dokumenata: Sistem će omogućiti generisanje PDF dokumenata sa strukturiranim podacima o prijemu pacijenta i napisanom nalazu, uz mogućnost preuzimanja istih putem dugmeta na listi prijema.

# 3. Arhitektura Sistema

## 3.1. Arhitekturni dizajn

Arhitektura softverskog sistema za manju kliniku se temelji na principima čiste arhitekture kako bi se postigla modularnost, fleksibilnost i lako održavanje koda. Implementiran je slojeviti dizajn koji obuhvata:

* Presentation Layer (Sloj prezentacije): Ovaj sloj predstavlja korisnički interfejs sistema i sadrži Angular aplikaciju koja omogućava korisnicima interakciju sa sistemom putem web preglednika.
* Application Layer (Sloj aplikacije): Ovaj sloj sadrži interfejse servisa koji se koriste za upravljanje poslovnim logikama aplikacije. Servisi omogućavaju izolaciju poslovnih operacija od konkretnih implementacija, olakšavajući testiranje i održavanje koda.
* Domain Layer (Sloj domena): Ovaj sloj sadrži osnovne entitete, objekte vrijednosti i poslovne logike aplikacije. Implementiran je koristeći princip generalizacije i parternog dizajna radi olakšanja proširivosti i ponovne upotrebe koda.
* Data Access Layer (Sloj pristupa podacima): Ovaj sloj obuhvata pristup podacima putem servisa koji omogućava komunikaciju sa bazom podataka.

## 3.2. Tehnologije korištene u implementaciji

* Backend: Implementacija backend dijela sistema je izvršena koristeći .NET 6 platformu. Korištena je ASP.NET Core za izgradnju web aplikacije, uz primjenu principa REST API dizajna. Kod je organiziran kroz parternu i generalizaciju kako bi se olakšalo održavanje i proširivost aplikacije.
* Frontend: Frontend dio sistema je izgrađen koristeći Angular framework. Angular aplikacija omogućava korisnicima interaktivno korištenje sistema putem web preglednika, pružajući intuitivan korisnički interfejs.
* Baza podataka: Kao baza podataka korištena je SQL relaciona baza podataka. Struktura baze podataka je zasnovana na tabelarnom modelu, što omogućava efikasno skladištenje i upravljanje podacima o pacijentima, ljekarima, prijemima i nalazima.

## 3.3. Struktura baze podataka

Struktura baze podataka je dizajnirana tako da podrži potrebne entitete i relacije kako bi se omogućila funkcionalnost sistema. Baza podataka sadrži tabele za entitete kao što su pacijenti, ljekari, prijemi i nalazi, sa odgovarajućim relacijama između njih. Ovaj dizajn omogućava efikasno skladištenje i upravljanje podacima, kao i brzu pretragu i filtriranje informacija koje koristi softverski sistem.

# 4. Implementacija

## 4.1. Detaljan opis implementacije sistema

Implementacija softverskog sistema za manju kliniku podrazumijevala je pažljivo planiranje, razvoj i integraciju različitih komponenti kako bi se osiguralo ispravno funkcionisanje sistema. Ovdje je detaljan opis procesa implementacije:

* Analiza zahtjeva: Prvi korak u implementaciji bio je detaljan pregled zahtjeva projekta kako bi se jasno definisale funkcionalnosti i arhitektonski zahtjevi sistema.
* Dizajn arhitekture: Nakon analize zahtjeva, pristupilo se dizajniranju arhitekture sistema, uključujući backend, frontend i bazu podataka. Razvijena je jasna struktura aplikacije koja omogućava modularnost i laku proširivost.
* Backend razvoj: Backend aplikacija je razvijena koristeći .NET 6 platformu i ASP.NET framework. Implementirani su kontroleri i servisi kako bi se omogućila komunikacija sa frontendom i bazom podataka.
* Frontend razvoj: Frontend aplikacija je izgrađena koristeći Angular framework. Razvijeni su korisnički interfejsi koji omogućavaju unos podataka, pregled prijema i nalaza, kao i druge funkcionalnosti sistema.
* Integracija: Nakon što su backend i frontend aplikacije razvijene, pristupilo se njihovoj integraciji kako bi se osiguralo ispravno funkcionisanje sistema kao cjeline.

## 4.2. Opis ključnih modula i funkcionalnosti

Softverski sistem za manju kliniku sastoji se od nekoliko ključnih modula i funkcionalnosti:

* Modul za upravljanje pacijentima: Omogućava unos, izmjenu i brisanje podataka o pacijentima, kao i pregled njihovih informacija.
* Modul za upravljanje ljekarima: Pruža mogućnost administriranja podataka o ljekarima, uključujući njihove titule i druge detalje.
* Modul za evidenciju prijema: Omogućava registrovanje prijema pacijenata.
* Modul za kreiranje nalaza: Pruža funkcionalnost za generisanje nalaza za svaki prijem pacijenta, uz automatsko dodjeljivanje datuma i vremena.
* Modul za pregled evidencija: Omogućava pregled liste prijema pacijenata sa osnovnim informacijama i akcijama za izmjenu i otkazivanje prijema.

## 4.3. Testiranje sistema

Testiranje sistema je izvršeno korištenjem Swagger alata za backend dio aplikacije. Ručno testiranje je obuhvatilo provjeru korisničkog interfejsa, unosa podataka i funkcionalnosti sistema. Ovo uključuje testiranje korisničkog iskustva kroz različite scenarije korištenja, kao što su unos podataka o pacijentima i ljekarima, evidencija prijema pacijenata, kreiranje nalaza, kao i pregled i filtriranje prikaza liste prijema.

Ova vrsta testiranja omogućava detaljnu provjeru svake funkcionalnosti sistema kako bi se osiguralo da korisničko iskustvo bude intuitivno i bez grešaka.

# 5. Korisnički Uputstva

## 5.1. Uputstva za korištenje sistema

Uputstva za korištenje sistema pružaju korisnicima detaljne informacije o tome kako koristiti softverski sistem za manju kliniku. Ovdje su neka od osnovnih uputstava:

* Prijava na sistem: Korisnici trebaju koristiti svoje korisničko ime i lozinku za prijavu na sistem.

A screenshot of a login screen

Description automatically generated

*Slika1. Prijava na sistem*

* Evidencija pacijenata: Korisnici mogu unositi nove pacijente, ažurirati postojeće podatke ili brisati pacijente iz sistema.

A screenshot of a computer

Description automatically generated

*Slika2. Unos pacijenta*

A screenshot of a computer

Description automatically generated

*Slika3. Ažuriranje pacijenta*

*A blue box with black text

Description automatically generated*

*Slika3. Brisanje pacijenta*

* Evidencija ljekara: Korisnici mogu dodavati nove ljekare, mijenjati informacije o njima ili ih brisati iz sistema.

A screenshot of a computer

Description automatically generated

*Slika4. Unos ljekara*

*A screenshot of a computer

Description automatically generated*

*Slika5. Ažuriranje ljekara*

*A screenshot of a computer

Description automatically generated*

*Slika6. Brisanje ljekara*

* Evidencija prijema pacijenata: Korisnici mogu registrirati prijem pacijenata, unoseći datum i vrijeme prijema, odabirući pacijenta i nadležnog ljekara, te označavajući hitni prijem ako je potrebno.

A screenshot of a computer

Description automatically generated

*Slika7. Unos prijema pacijenta*

* Kreiranje nalaza: Za svaki prijem pacijenta, korisnici mogu kreirati nalaz unoseći tekstualni opis i ostavljajući sistem da automatski dodijeli datum i vrijeme nalaza.

A screenshot of a computer

Description automatically generated

*Slika8. Unos nalaza*

* Pregled i filtriranje prijema: Korisnici mogu pregledati listu prijema pacijenata, filtrirajući ih po datumu prijema.

A screenshot of a computer

Description automatically generated

*Slika9. Pregled I filtriranje prijema*

## 5.2. Uputstva za ljekare

Uputstva za ljekare pružaju specifične smjernice o tome kako ljekari mogu koristiti sistem za upravljanje pacijentima i njihovim zdravstvenim podacima. Ovdje su neka od osnovnih uputstava:

* Pregled liste pacijenata: Ljekari mogu pregledati listu pacijenata kako bi vidjeli njihove osnovne informacije.
* Dodavanje pacijenata: Ljekari imaju mogućnost dodavanja novih pacijenata u sistem unoseći njihove osnovne informacije.
* Pisanje prijema pacijenta: Ljekari, ako su specijalisti, mogu kreirati prijem pacijenta unoseći detalje o prijemu, kao što su datum i vrijeme prijema, odabir pacijenta i drugi relevantni podaci.
* Ažuriranje prijema pacijenata: Ljekari mogu ažurirati postojeće podatke o prijemu pacijenata, uključujući izmjenu datuma i vremena prijema, promjenu pacijenta ili nadležnog ljekara, ili drugih relevantnih informacija.
* Pisanje nalaza: Ljekari imaju mogućnost kreiranja nalaza za svaki prijem pacijenta unoseći detalje o pregledu i dijagnozi.

## 5.3. Uputstva za razvojni tim

Uputstva za razvojni tim pružaju smjernice o tome kako održavati, proširivati i podržavati softverski sistem. Ovdje su neka od osnovnih uputstava:

* Održavanje sistema: Redovno ažurirajte softverski sistem kako biste osigurali sigurnost i stabilnost.
* Proširenje funkcionalnosti: Razvijajte nove funkcionalnosti sistema kako bi se zadovoljile potrebe korisnika i pružila bolja usluga.
* Optimizacija performansi: Kontinuirano optimizirajte performanse sistema kako biste osigurali brzo i efikasno korisničko iskustvo.
* Podrška korisnicima: Pružajte podršku korisnicima kroz tehničku podršku i obuke kako bi se osiguralo da korisnici ispravno koriste sistem.

# 6. Tehnička Dokumentacija

## 6.1. Opis ključnih funkcija i klasa

U implementaciji sistema za manju kliniku, ključne funkcije i klase su organizirane na sljedeći način:

* PacijentService: Ovaj servis sadrži funkcionalnosti za upravljanje podacima o pacijentima, uključujući kreiranje, čitanje, ažuriranje i brisanje.
* LjekarService: Servis za upravljanje podacima o ljekarima, omogućava dodavanje novih ljekara, ažuriranje postojećih i brisanje.
* PrijemService: Servis koji omogućava upravljanje evidencijom prijema pacijenata, uključujući kreiranje novih prijema, ažuriranje postojećih i brisanje.
* NalazService: Servis za upravljanje kreiranjem i čitanjem nalaza za pacijente.

## 6.3. Struktura baze podataka

Baza podataka je dizajnirana koristeći SQL Server i sadrži nekoliko ključnih tabela za skladištenje podataka o pacijentima, ljekarima, prijemima pacijenata i nalazima. Ovdje su neke od glavnih tabela:

* Pacijenti: Tabela koja sadrži informacije o pacijentima, uključujući ime, prezime, datum rođenja, spol i druge osnovne podatke.
* Ljekari: Tabela za čuvanje podataka o ljekarima, kao što su ime, prezime, titula i ostali relevantni detalji.
* Prijemi: Tabela za evidenciju prijema pacijenata, uključujući informacije o datumu i vremenu prijema, pacijentu, nadležnom ljekaru i drugim relevantnim podacima.
* Nalazi: Tabela za čuvanje tekstualnih opisa nalaza za svaki prijem pacijenta, zajedno sa datumom i vremenom kreiranja nalaza.

Ova struktura baze podataka omogućava efikasno skladištenje i upravljanje podacima o pacijentima i njihovim zdravstvenim podacima u sistemu.

A screenshot of a computer

Description automatically generated

*Slika10. Baza podataka*

# 7. Rezultati i Zaključci

## 7.1. Analiza postignutih rezultata

Implementacija softverskog sistema za manju kliniku omogućila je uspješno ostvarenje ciljeva projekta. Kroz detaljnu analizu postignutih rezultata, ističemo sljedeće:

Funkcionalnosti sistema: Sve funkcionalnosti sistema, uključujući evidenciju pacijenata, ljekara, prijema pacijenata i kreiranje nalaza, uspješno su implementirane prema zahtjevima projekta.

Korisničko iskustvo: Korisničko iskustvo je poboljšano kroz intuitivan korisnički interfejs i mogućnost efikasnog upravljanja podacima o pacijentima i njihovim prijemima.

Stabilnost sistema: Softverski sistem pokazuje stabilnost i pouzdanost u radu, bez značajnih problema ili nedostataka.

## 7.2. Procjena uspješnosti projekta

Projekat se može ocijeniti kao uspješan s obzirom na postignute rezultate i ispunjenje zadanih ciljeva. Implementirani softverski sistem pruža efikasno rješenje za upravljanje procesima u manjoj klinici, olakšavajući evidenciju pacijenata, organizaciju prijema i kreiranje zdravstvenih nalaza.

Uspješnost projekta također se ogleda u tome što je ispoštovan zahtjev korisnika za web baziranom aplikacijom s intuitivnim korisničkim interfejsom, prilagođenom kako za ljekare tako i za administratore klinike.

## 7.3. Predlozi za buduće unapređenje

Iako je implementirani sistem funkcionalan i zadovoljava osnovne potrebe klinike, postoje neki predlozi za buduće unapređenje:

Dodatne funkcionalnosti: Razmotriti dodavanje dodatnih funkcionalnosti poput integracije sa sistemom za zakazivanje termina, upravljanje inventarom ili fakturisanje usluga.

Optimizacija performansi: Kontinuirano optimizirati performanse sistema kako bi se osiguralo brzo odzivno vrijeme i efikasnost prilikom obrade podataka.

Korisnička podrška: Pružiti kontinuiranu podršku korisnicima kroz obuke i uputstva kako bi se osiguralo maksimalno korištenje sistema i zadovoljstvo korisnika.

Sigurnost podataka: Dodatno unaprijediti sigurnost podataka implementacijom dodatnih sigurnosnih mehanizama kako bi se zaštitili osjetljivi zdravstveni podaci pacijenata.

Ovi predlozi za buduće unapređenje mogu doprinijeti daljem poboljšanju softverskog sistema i pružiti dodatne koristi korisnicima klinike.