Univerzitet u Novom Sadu

Tehnički fakultet »Mihajlo Pupin«

Zrenjanin

*Školska 2023/24 godina*

*Nastavni predmet: Softversko Inženjerstvo 2*

**SEMINARSKI RAD**

Tema: Evidencija popravki aviona

Predmetni nastavnik:

Prof. dr Ljubica Kazi

Predmetni asistent:

MSc Vuk Amižić

Student:

Marko Smiljanić IT 19/14

Zrenjanin, 2024. godina

Sadržaj

[**1.** **ANALIZA POSLOVNOG SISTEMA, SPECIFIKACIJA ZAHTEVA I PLANIRANJE RAZVOJA** 4](#_Toc170078858)

[**1.1.** **Analiza poslovnog sistema** 4](#_Toc170078859)

[**1.1.1.** **Svrha postojanja organizacionog sistema i kategorizacija poslova** 4](#_Toc170078860)

[**1.1.2.** **Osnovni objekti obrade i opis toka poslovnog procesa** 5](#_Toc170078861)

[**1.1.3.** **Organizaciona struktura sistema i pripadnost drugim organizacionim strukturama** 7](#_Toc170078862)

[**1.1.4.** **Radna mesta i opis zaduženja** 8](#_Toc170078863)

[**1.1.5.** **Spisak korišćene dokumentacije** 10](#_Toc170078864)

[**1.1.6.** **Zakonska regulativa, standardi i interni propisi organizacionog sistema** 12](#_Toc170078865)

[**1.2.** **Specifikacija zahteva korisnika** 14](#_Toc170078866)

[**1.2.1.** **Profili korisnika i očekivane funkcionalne mogućnosti softvera** 14](#_Toc170078867)

[**1.2.2.** **Spisak potrebnih ulaznih ekranskih formi** 16](#_Toc170078868)

[**1.2.3.** **Spisak potrebnih tabelarnih prikaza sa kriterijumima filtriranja** 18](#_Toc170078869)

[**1.2.4.** **Spisak potrebnih štampi dokumenata** 20](#_Toc170078870)

[**1.2.5.** **Spisak poslovnih pravila, automatizama i ograničenja koje je potrebno ugaditi u softver** 22](#_Toc170078871)

[**1.2.6.** **Potrebne karakteristike korisničkog interfejsa i ostali nefunkcionalni zahtevi** 24](#_Toc170078872)

[**1.3.** **Elementi projektnog planiranja** 26](#_Toc170078873)

[**1.3.1.** **Granice (scope) projekta** 26](#_Toc170078874)

[**1.3.2.** **Kratak opis tehnologija, programskih jezika i razvojnih alata koji će biti korišćeni u radu** 28](#_Toc170078875)

[**2.** **POSLOVNA SISTEMSKA ANALIZA I DIZAJN SOFTVERSKOG RЕŠENJA** 29](#_Toc170078876)

[**2.1.** **Model poslovnih procesa** 29](#_Toc170078877)

[**2.1.1.** **Algoritamski model poslovnih procesa** 29](#_Toc170078878)

[**2.1.2.** **Spisak skladišta podataka** 29](#_Toc170078879)

[**2.1.3.** **Analiza jednog poslovnog dokumenta** 30](#_Toc170078880)

[**2.2.** **Dizajn softverskih funkcija** 33](#_Toc170078881)

[**2.2.1.** **Tabela preslikavanja primitivnih procesa u softverske funkcije** 33](#_Toc170078882)

[**2.2.2.** **USE CASE dijagram softverskih funkcija** 34](#_Toc170078883)

[**2.2.3.** **Specifikacija slučaja korišćenja za unos i tabelarni prikaz podataka** 34](#_Toc170078884)

[**2.3.** **Dizajn modela podataka** 38](#_Toc170078885)

[**2.3.1.** **Konceptualni model podataka** 38](#_Toc170078886)

[**2.3.2.** **Fizički (relacioni) model podataka** 38](#_Toc170078887)

[3. **IMPLEMENTACIJA, TESTIRANJE I DOKUMENTOVANJE REALIZOVANOG SOFTVERA** 39](#_Toc170078888)

[**3.1.** **Korisničko i tehničko uputstvo** 39](#_Toc170078889)

[**3.1.1.** **Korisničko uputstvo** 39](#_Toc170078890)

[**3.1.2.** **Tehničko uputstvo** 43](#_Toc170078891)

[**3.2.** **Izveštaj o testiranju softvera** 44](#_Toc170078892)

[**3.2.1.** **Test slučajevi ispravnog rada korisnika i prikaz rezultata testiranja** 44](#_Toc170078893)

[**3.2.2.** **Test slučajevi neispravnog rada korisnika i prikaz rezultata testiranja** 44](#_Toc170078894)

[**3.3.** **Dokumentovanje implementacije softverskog rešenja** 45](#_Toc170078895)

[**3.3.1.** **Modeli opisa implementacije rešenja** 45](#_Toc170078896)

[**3.3.1.1.** **Dijagram komponenti** 45](#_Toc170078897)

[**3.3.1.2.** **Djagram klasa** 46](#_Toc170078898)

[**3.3.1.3.** **Dijagram sekvenci za osnovni scenario unosa podataka i tabelarnog prikaza sa filterom** 47](#_Toc170078899)

[**3.3.2.** **Tabelarni prikaz komponenti** 48](#_Toc170078900)

[**3.3.3.** **SQL script** 49](#_Toc170078901)

[**3.3.4.** **Delovi koda sa objašnjenjima po slojevima** 51](#_Toc170078902)

[**3.3.4.1.** **Sloj za rad sa podacima** 51](#_Toc170078903)

[**3.3.4.2.** **Sloj servisa ili DTO** 54](#_Toc170078904)

[**3.3.4.3.** **Sloj poslovne logike** 55](#_Toc170078905)

[**3.3.4.4.** **Kod za osnovnu validaciju podataka sloja prezentacione logike** 56](#_Toc170078906)

[**3.3.4.5.** **Korisnički interfejs** 57](#_Toc170078907)

[**3.3.5.** **Delovi koda i opis implementacije podrške personalizaciji aplikacije i bezbednosti** 59](#_Toc170078908)

[**3.3.6.** **Delovi koda i opis implementacije osnovnih principa OOP** 60](#_Toc170078909)

[**3.3.7.** **Delovi koda i opis implementacije SOLID principa** 61](#_Toc170078910)

[**3.3.8.** **Delovi koda objašnjenja realizacije clean code pristupa** 62](#_Toc170078911)

[**4.** **LITERATURA** 63](#_Toc170078912)

# **ANALIZA POSLOVNOG SISTEMA, SPECIFIKACIJA ZAHTEVA I PLANIRANJE RAZVOJA**

## **Analiza poslovnog sistema**

### **Svrha postojanja organizacionog sistema i kategorizacija poslova**

Organizacioni sistem evidencije popravki aviona oblikovan je s ciljem efikasnog vođenja evidencije popravki i održavanja aviona u avio-industriji. Sistem je dizajniran kako bi olakšao upravljanje kompleksnim procesima održavanja aviona, prateći svaku fazu popravke od početka do kraja.

**Kategorizacija poslova:**

Osnovne delatnosti

* Osnovne delatnosti sistema obuhvataju sve aktivnosti direktno povezane s održavanjem i popravkom aviona. Ovo uključuje evidentiranje kvarova, planiranje radnih naloga, praćenje inventara potrebnih delova i alata, kao i izveštavanje o statusu popravki.

Pomoćne delatnosti

* Pomoćne delatnosti podržavaju osnovne procese održavanja aviona pružajući podršku u administrativnim, logističkim i tehničkim aspektima. To uključuje administraciju podataka, upravljanje dokumentacijom, koordinaciju između različitih timova, kao i obuku osoblja.

Upravljačke delatnosti

* Upravljačke delatnosti odnose se na sve aktivnosti vezane za vođenje evidencije popravki aviona kao organizacione jedinice. Ovo obuhvata planiranje strategija održavanja, upravljanje resursima, donošenje odluka na osnovu analize podataka i osiguranje usklađenosti sa relevantnim propisima i standardima.

Zaključak

* Organizacioni sistem evidencije popravki aviona osmišljen je kako bi efikasno podržao sve nivoe operacija vezanih za održavanje aviona. Kategorizacija poslova omogućava jasnu raspodelu odgovornosti i efikasno upravljanje resursima, što doprinosi optimizaciji procesa održavanja i povećanju sigurnosti letenja. [1]

### **Osnovni objekti obrade i opis toka poslovnog procesa**

**Osnovni objekti obrade**

Osnovni objekti obrade u domenu evidencije popravki aviona su ključni entiteti koji predstavljaju osnovne elemente podataka i operacija u sistemu. Svaki objekat obrade igra vitalnu ulogu u praćenju i upravljanju procesima održavanja aviona.

Popravka aviona

* Životni ciklus obrade: Ovaj objekat predstavlja celokupan proces popravke ili održavanja aviona, počevši od evidentiranja problema ili kvara, preko planiranja i izvršavanja radnih naloga, do zatvaranja popravke i praćenja eventualnih naknadnih aktivnosti.
* Opis posla: Uključuje evidentiranje detalja popravke, poput identifikacije aviona, opisa problema, planiranja resursa i vremena, dodeljivanja zaduženja osoblju, praćenja napretka popravke i evidentiranja utrošenih materijala i radnih sati.

Radni nalog

* Životni ciklus obrade: Radni nalog predstavlja detaljan plan aktivnosti potrebnih za izvršenje popravke aviona. Počinje kreiranjem naloga na osnovu identifikovanog problema ili redovnog održavanja, a završava se kada su sve predviđene aktivnosti uspešno okončane.
* Opis posla: Ovaj objekat uključuje specifične zadatke, potrebne resurse, raspored rada, kao i rokove za završetak. Takođe, sadrži informacije o potrebnim delovima i materijalima, kao i očekivane troškove i budžete.

Inventar

* Životni ciklus obrade: Inventar predstavlja sve materijale, delove i alate koji su neophodni za izvršenje popravki aviona. Ovaj objekat prolazi kroz ciklus nabavke, skladištenja, dodele resursa radnim nalozima, kao i evidenciju potrošnje i obnavljanja zaliha.
* Opis posla: Ovde se evidentiraju sve stavke inventara, uključujući detalje o količinama, lokacijama, datumima nabavke i isteka rokova trajanja. Takođe, prati se kretanje inventara kroz sistem, od prijema do korišćenja u popravkama aviona.

**Opis toka poslovnog procesa**

Opis toka poslovnog procesa detaljno definiše korake koji se preduzimaju u svakom od osnovnih objekata obrade kako bi se ostvario cilj, a to je uspešno održavanje aviona. Proces uključuje sledeće faze:

Identifikacija problema ili redovno održavanje aviona

* U ovoj fazi, osoblje prima zahtev ili identifikuje potrebu za popravkom ili održavanjem aviona na osnovu rutinskih pregleda ili prijavljenih problema.

Kreiranje radnog naloga

* Na osnovu identifikovanog problema, kreira se radni nalog koji detaljno definiše sve korake i resurse potrebne za izvršenje popravke.

Izvršenje radnog naloga

* Radni nalog se dodeljuje odgovarajućem osoblju koje izvršava planirane aktivnosti u skladu sa postavljenim rokovima i standardima.

Praćenje popravke

* Tokom izvršenja radnog naloga, sistem kontinuirano prati napredak popravke i omogućava realno vreme informacije o statusu i potrošnji resursa.

Zatvaranje popravke

* Nakon uspešnog završetka popravke, radni nalog se zatvara, a sve aktivnosti i troškovi se evidentiraju u sistemu.

Ovim opisom toka poslovnog procesa osigurava se transparentnost, efikasnost i tačnost u svakom koraku obrade popravki aviona, što doprinosi pouzdanosti i bezbednosti letova.

### **Organizaciona struktura sistema i pripadnost drugim organizacionim strukturama**

U poglavlju o organizacionoj strukturi sistema i pripadnosti drugim organizacionim strukturama, daje se objašnjenje kako je sistem evidencije popravki aviona interno organizovan i kako se uklapa u širu organizacionu strukturu kompanije. Razumevanje ove strukture je ključno za efikasnu upotrebu sistema, koordinaciju sa drugim odeljenjima i implementaciju održivih procedura.

Organizacija sistema

Sistem evidencije popravki aviona organizovan je kako bi podržao upravljanje i evidenciju svih aspekata popravki aviona. Organizacija sistema može se podeliti na sledeće celine:

* Korisnici: Sistem omogućava pristup različitim nivoima korisnika, uključujući administratore, tehničare, i menadžere. Svaka grupa ima definisane dozvole i zadatke.
* Moduli: Sistem sadrži nekoliko modula specijalizovanih za različite zadatke, uključujući modul za evidentiranje popravki, modul za održavanje zaliha, modul za analizu performansi i modul za izveštavanje.
* Podaci: Podaci o popravkama, avionima, delovima i održavanju se prikupljaju i organizuju u baze podataka sistema. Ove informacije su dostupne preko sistema i podložne su kontroli pristupa.

Pripadnost drugim organizacionim strukturama

Sistem evidencije popravki aviona je ključni deo šire organizacione strukture kompanije, koja može uključivati odeljenja kao što su:

* Odeljenje za tehničko održavanje: Sistem je neophodan za podršku tehničarima u evidentiranju i praćenju popravki, pružajući im pregled aktuelnih zadataka i resursa.
* Odeljenje za logistiku: Sistem pomaže u optimizaciji lanca snabdevanja, pružajući podatke o zalihe i delovima potrebnim za popravke.
* Odeljenje za planiranje: Sistem pruža informacije koje pomažu u planiranju i raspodeli resursa za nadolazeće popravke i održavanje.
* Odeljenje za analitiku: Sistem omogućava analiziranje podataka o popravkama i performansama aviona, što pomaže u donošenju strateških odluka.

Integracija sa drugim sistemima

Sistem evidencije popravki aviona može biti integrisan sa drugim informacionim sistemima unutar organizacije, kao što su sistemi za praćenje leta, finansijski sistemi, i sistemi za upravljanje radnom snagom. Ova integracija omogućava sinhronizaciju podataka i poboljšava efikasnost procesa.

Komunikacija i saradnja

Sistem takođe olakšava komunikaciju i saradnju između različitih odeljenja i organizacionih jedinica. Na primer, tehničari mogu direktno komunicirati sa menadžerima ili drugim tehničarima u vezi sa specifičnim popravkama, dok menadžeri mogu dobiti ažuriranja o statusu popravki u realnom vremenu.

Zaključak

Organizaciona struktura sistema evidencija popravki aviona igra ključnu ulogu u efikasnom funkcionisanju celokupne organizacije. Njegova povezanost sa drugim odeljenjima i sistemima doprinosi optimalnom korišćenju resursa i poboljšanju ukupnih performansi aviona i kompanije u celini.

### **Radna mesta i opis zaduženja**

U ovom poglavlju su definisane različite uloge i odgovornosti korisnika unutar sistema. Jasno definisani zadaci za svaku ulogu omogućavaju efikasno korišćenje sistema, odgovarajuće upravljanje podacima i optimalnu saradnju među korisnicima.

Radna mesta i uloge

U sistemu evidencije popravki aviona identifikovane su sledeće ključne uloge korisnika, svaka sa specifičnim zadacima i odgovornostima:

**Administrator sistema:**

Zadaci:

* Kreiranje i upravljanje korisničkim nalozima i pristupnim pravima.
* Postavljanje i prilagođavanje parametara sistema.
* Održavanje i ažuriranje sistema softverskim zakrpama i novim funkcijama.
* Nadgledanje performansi sistema i rešavanje problema.

Odgovornosti:

* Obezbeđivanje sigurnosti sistema i zaštite podataka.
* Obezbeđivanje kontinuiranog rada sistema bez prekida.

**Tehničar:**

Zadaci:

* Evidentiranje i ažuriranje informacija o popravkama aviona.
* Vođenje evidencije o delovima, alatima i materijalima korišćenim za popravke.
* Praćenje zaliha delova i materijala potrebnih za popravke.

Odgovornosti:

* Obezbeđivanje tačnosti i ažurnosti podataka unetih u sistem.
* Pridržavanje procedura za rad i bezbednost pri obavljanju popravki.

**Menadžer za održavanje:**

Zadaci:

* Pratiti i nadgledati rad tehničara u vezi sa popravkama aviona.
* Koordinirati raspored i zadatke tehničara.
* Pripremati izveštaje i analize o popravkama i performansama aviona.

Odgovornosti:

* Osigurati kvalitet obavljenih popravki i pridržavanje propisanih standarda.
* Analizirati podatke i trendove za donošenje strateških odluka.

**Logističar:**

Zadaci:

* Upravljanje zalihama delova i materijala potrebnih za popravke.
* Naručivanje i obnavljanje zaliha u skladu sa potrebama tehničara i menadžera.
* Praćenje tokova materijala i delova kroz proces popravki.

Odgovornosti:

* Obezbeđivanje da su svi potrebni delovi i materijali dostupni tehničarima.
* Usklađivanje logističkih aktivnosti sa potrebama popravki.

Radne procedure i standardi

Svaka radna uloga treba da se pridržava propisanih procedura i standarda kako bi se obezbedila tačnost i efikasnost rada. Uloga administratora uključuje obezbeđivanje pridržavanja ovih standarda i pružanje podrške korisnicima sistema.

Zaključak

Jasno definisane uloge i odgovornosti unutar sistema evidencije popravki aviona omogućavaju strukturiran rad, efikasnu upotrebu resursa i poboljšanu komunikaciju i koordinaciju između različitih korisničkih grupa. Ova struktura obezbeđuje postizanje maksimalne efikasnosti i kvaliteta rada unutar sistema.

### **Spisak korišćene dokumentacije**

Jasno definisanje radne dokumentacije i njenog korišćenja omogućava efikasno prikupljanje, praćenje i analiziranje podataka u vezi sa popravkama aviona.

Knjige evidencije

Knjige evidencije su elektronski ili fizički dokumenti koji se koriste za prikupljanje i organizovanje podataka o popravkama aviona. U sistemu evidencije popravki aviona koriste se sledeće knjige evidencije:

* Knjiga evidencije popravki aviona:

Sadrži informacije o svakoj pojedinačnoj popravki aviona, uključujući datum i vreme popravke, vrstu popravke, zaduženog tehničara i upotrebljene delove i materijale.

* Knjiga evidencije delova i materijala:

Sadrži podatke o delovima i materijalima korišćenim za popravke, uključujući zalihe, porudžbine i korišćenje delova u konkretnim popravkama.

* Knjiga evidencije tehničara:

Prati radne sate i zadatke tehničara u vezi sa popravkama aviona, uključujući njihovu efikasnost i performanse.

Obrasci

Obrasci su šabloni koji olakšavaju prikupljanje i unos podataka u sistem. Koriste se za standardizaciju unosa podataka i poboljšanje tačnosti evidencije. U sistemu evidencije popravki aviona koriste se sledeći obrasci:

* Obrazac za prijavu popravke:

Koristi se za evidentiranje nove popravke aviona, uključujući detalje o avionu, vrsti kvara i potrebnim popravkama.

* Obrazac za praćenje popravke:

Koristi se za praćenje napretka popravke, uključujući faze rada i zadatke tehničara.

* Obrazac za naručivanje delova i materijala:

Koristi se za naručivanje novih delova i materijala potrebnih za popravke, uključujući detalje o količini i specifikacijama.

Arhiva dokumenata

Arhiva dokumenata predstavlja sistematski organizovanu kolekciju dokumenata koji se čuvaju za buduću upotrebu, analizu ili proveru. U sistemu evidencije popravki aviona arhiva dokumenata uključuje:

* Arhivu evidencija popravki:

Sadrži istoriju svih popravki aviona, uključujući sve podatke unete u sistem, kao što su datumi popravki, detalji o avionu, delovima i tehničarima.

* Arhiva izveštaja i analiza:

Sadrži kopije izveštaja i analiza generisanih iz sistema, koji su korišćeni za donošenje strateških odluka ili kao dokaz o obavljenim popravkama.

* Arhiva logističkih podataka:

Sadrži podatke o zalihama delova i materijala, uključujući istoriju porudžbina i isporuka.

Zaključak

Korišćenje dokumentacije, obrazaca i arhiva u sistemu evidencije popravki aviona omogućava strukturiran i efikasan pristup podacima o popravkama aviona. Ove prakse obezbeđuju tačnost i kontinuitet informacija, što je ključno za poboljšanje procesa održavanja aviona i donošenje informisanih odluka.

### **Zakonska regulativa, standardi i interni propisi organizacionog sistema**

Predstojeće poglavlje se bavi pravilima i procedurama koje uređuju sistem evidencije popravki aviona u skladu sa zakonskim zahtevima Republike Srbije, relevantnim standardima i internim propisima organizacije. Ova regulativa obezbeđuje usklađenost sistema sa važećim zakonima, kao i kvalitet i bezbednost u oblasti održavanja aviona.

Zakonska regulativa

* Zakon o vazdušnom saobraćaju Republike Srbije:

Sistem evidencije popravki aviona mora biti usklađen sa Zakonom o vazdušnom saobraćaju Republike Srbije, koji propisuje uslove i pravila u vezi sa održavanjem i popravkama aviona.

Ova regulativa obuhvata zahteve za bezbednost, kvalitet, licenciranje i sertifikaciju aviona, kao i zahteve za obuku i ovlašćenje tehničkog osoblja.[2]

* Pravilnici i propisi u oblasti vazduhoplovstva:

Osim zakona, sistem treba da prati relevantne pravilnike i propise u vezi sa održavanjem aviona, kao što su Pravilnik o održavanju i popravci vazduhoplova, Pravilnik o licenciranju i kvalifikaciji tehničkog osoblja, i drugi propisi koji se tiču tehničke dokumentacije i procedura.

Standardi

* Standardi bezbednosti u vazduhoplovstvu:

Sistem mora da podržava primenu međunarodnih i domaćih standarda bezbednosti u vazduhoplovstvu, uključujući standarde Međunarodne organizacije za civilno vazduhoplovstvo (ICAO) i Evropske agencije za vazduhoplovnu bezbednost (EASA).[2][3]

* Standardi za upravljanje kvalitetom:

Sistem treba da prati standarde upravljanja kvalitetom kao što su ISO 9001, koji obezbeđuju kvalitetne procese i dokumentaciju u vezi sa popravkama aviona.

Interni propisi organizacije

* Politike i procedure organizacije:

Sistem evidencije popravki aviona treba da bude usklađen sa internim politikama i procedurama organizacije, uključujući pravila o radu, sigurnosti i zaštiti podataka.

* Pravila o etici i profesionalnom ponašanju:

Svi korisnici sistema treba da se pridržavaju pravila o etici i profesionalnom ponašanju u vezi sa evidentiranjem i praćenjem popravki aviona.

* Politike za obuku i kvalifikaciju:

Sistem treba da podržava interne politike za obuku i kvalifikaciju tehničkog osoblja, uključujući praćenje njihovih sertifikacija i licenci.

Zaključak

Pridržavanje zakonske regulative, standarda i internih propisa organizacionog sistema je ključno za bezbedno, kvalitetno i efikasno upravljanje popravkama aviona u sistemu evidencije popravki aviona. Osiguravanje usklađenosti sa ovim zahtevima obezbeđuje usluge visokog kvaliteta i sigurnosti u oblasti vazduhoplovstva.

## **Specifikacija zahteva korisnika**

### **Profili korisnika i očekivane funkcionalne mogućnosti softvera**

Ovaj segment se bavi različitim kategorijama korisnika sistema evidencije popravki aviona i njihovim očekivanjima u vezi sa funkcionalnostima softvera. Definisanje očekivanih funkcionalnih mogućnosti pruža okvir za izgradnju softverskog proizvoda koji zadovoljava potrebe korisnika i olakšava efikasno upravljanje popravkama aviona.

Profili korisnika

**Administrator sistema:**

* Administratori su odgovorni za upravljanje korisničkim nalozima, pristupnim pravima i podešavanjima sistema.
* Očekuju intuitivne alate za upravljanje korisnicima, dozvolama i podešavanjima sistema.
* Žele funkcije za monitoring i izveštavanje o performansama sistema.

**Tehničar:**

* Tehničari su zaduženi za evidentiranje i praćenje popravki aviona, kao i za vođenje evidencije o delovima i materijalima.
* Očekuju jednostavan i brz pristup podacima o popravkama i avionima.
* Žele funkcije za praćenje zaliha i delova, uključujući naručivanje i korišćenje materijala.

**Menadžer za održavanje:**

* Menadžeri su odgovorni za koordinaciju i nadgledanje rada tehničara i procesa održavanja.
* Očekuju alate za analizu podataka o popravkama i performansama aviona.
* Žele funkcije za izveštavanje o stanju aviona, planiranje održavanja i upravljanje resursima.

**Logističar:**

* Logističari su zaduženi za upravljanje zalihama i nabavkom delova i materijala.
* Očekuju funkcije za praćenje i naručivanje zaliha, kao i kontrolu troškova.
* Žele mogućnosti za integraciju sa dobavljačima i praćenje isporuka.

Očekivane funkcionalne mogućnosti softvera

* Upravljanje korisničkim nalozima i pristupnim pravima:

Sistem treba da podržava stvaranje, upravljanje i brisanje korisničkih naloga, kao i postavljanje pristupnih prava i dozvola.

* Evidencija popravki aviona:

Očekuje se funkcija za unos i praćenje informacija o popravkama aviona, uključujući detalje o avionima, popravkama i tehničarima.

* Upravljanje delovima i zalihama:

Sistem treba da podržava evidenciju i praćenje delova i zaliha, uključujući informacije o dostupnosti, korišćenju i nabavci.

* Izveštaji i analize:

Očekuju se funkcije za generisanje izveštaja i analiza o popravkama, performansama aviona, radnim zadacima tehničara i stanju zaliha.

* Integracija sa drugim sistemima:

Softver treba da bude sposoban za integraciju sa drugim sistemima unutar organizacije, kao što su finansijski i logistički sistemi.

* Bezbednost i zaštita podataka:

Korisnici očekuju visok nivo bezbednosti i zaštite podataka, uključujući šifrovanje i kontrolu pristupa.

* Prilagođavanje i konfiguracija:

Očekuje se mogućnost prilagođavanja softvera specifičnim potrebama organizacije, uključujući podešavanja i preferencije korisnika.

Zaključak

Jasno definisani profili korisnika i očekivane funkcionalne mogućnosti softvera pružaju osnovu za razvoj sistema evidencije popravki aviona koji će efikasno zadovoljavati potrebe različitih korisničkih grupa. Fokus na ispunjavanju očekivanja korisnika doprinosi poboljšanoj funkcionalnosti, efikasnosti i produktivnosti unutar organizacije.

### **Spisak potrebnih ulaznih ekranskih formi**

Različite ulazne ekranske forme koje su potrebne za pravilno prikupljanje i unos podataka u sistem evidencije popravki aviona su definisane u nastavku. One omogućavaju korisnicima unos informacija o popravkama, avionima, delovima, radnicima i drugim važnim aspektima sistema.

Ekranske forme za unos informacija o popravkama

**Forma za prijavu popravke aviona:**

* Omogućava unos podataka o avionu, uključujući identifikacione brojeve i model.
* Prijava kvara ili problema na avionu.
* Omogućava dodavanje opisa popravke i potrebnih delova ili materijala.

**Forma za praćenje popravke:**

* Omogućava ažuriranje informacija o napretku popravke.
* Prikazuje zadatke koje tehničar treba da izvrši i rokove za svaki zadatak.

Ekranske forme za unos informacija o avionima

**Forma za dodavanje i upravljanje avionima:**

* Omogućava unos podataka o novim avionima u sistem, uključujući broj registracije, model, serijski broj i datum kupovine.
* Prikazuje istoriju popravki i održavanja svakog aviona.

Ekranske forme za evidenciju delova i zaliha

**Forma za unos delova i materijala:**

* Omogućava unos podataka o delovima i materijalima, uključujući specifikacije, serijske brojeve, količine i zalihe.
* Podržava unos informacija o dobavljačima i cenama.

**Forma za praćenje zaliha:**

* Omogućava pregled trenutnih zaliha delova i materijala.
* Prikazuje informacije o korišćenju delova u popravkama i istoriji narudžbi.

Ekranske forme za unos informacija o tehničarima i radnicima

**Forma za unos tehničara i radnika:**

* Omogućava unos podataka o tehničarima i radnicima, uključujući identifikacione podatke, kvalifikacije i radno iskustvo.
* Podržava unos informacija o obuci i sertifikacijama.

**Forma za praćenje radnih zadataka:**

* Omogućava evidentiranje radnih zadataka koje su tehničari obavljali tokom popravki.
* Podržava unos informacija o radnom vremenu i efektivnosti.

Ekranske forme za administraciju i konfiguraciju

**Forma za upravljanje korisnicima i pristupnim pravima:**

* Omogućava administratorima upravljanje korisničkim nalozima, uključujući kreiranje, uređivanje i brisanje naloga.
* Omogućava postavljanje pristupnih prava i dozvola za korisnike.

**Forma za konfiguraciju sistema:**

* Omogućava administratorima postavljanje i ažuriranje podešavanja sistema.
* Podržava prilagođavanje sistema specifičnim potrebama organizacije.

Zaključak

Ulazne ekranske forme su ključne za efikasno prikupljanje i unos podataka u sistem evidencije popravki aviona. Svaka forma treba da bude intuitivna i prilagođena potrebama korisnika kako bi omogućila tačno i pravovremeno evidentiranje podataka, što doprinosi poboljšanoj funkcionalnosti i upravljanju popravkama aviona.

### **Spisak potrebnih tabelarnih prikaza sa kriterijumima filtriranja**

Sledi opis tabelarnih prikaza koji su potrebni u sistemu evidencije popravki aviona kako bi se olakšao pregled, organizacija i analiza podataka. Ovi prikazi omogućavaju korisnicima da brzo pretražuju i filtriraju podatke prema različitim kriterijumima, što doprinosi efikasnom upravljanju popravkama aviona.

Tabelarni prikazi za evidenciju popravki aviona

**Tabelarni prikaz popravki:**

* Prikazuje listu svih popravki aviona u sistemu, uključujući informacije kao što su broj popravke, datum i vreme, identifikacioni broj aviona, vrsta popravke, zaduženi tehničar i status popravke.
* Kriterijumi filtriranja: Datum, vrsta popravke, status, tehničar, avion.

**Tabelarni prikaz napretka popravki:**

* Prikazuje listu popravki sa informacijama o trenutnom napretku, uključujući faze rada, zadatke tehničara i rokove.
* Kriterijumi filtriranja: Napredak, faza rada, tehničar, avion, rok.

Tabelarni prikazi za evidenciju delova i zaliha

**Tabelarni prikaz delova i materijala:**

* Prikazuje listu svih delova i materijala u sistemu, uključujući informacije kao što su naziv dela, serijski broj, opis, količina, cena i dobavljač.
* Kriterijumi filtriranja: Naziv dela, serijski broj, kategorija, dobavljač.

**Tabelarni prikaz zaliha:**

* Prikazuje trenutne zalihe delova i materijala sa informacijama o dostupnosti, narudžbama i potrošnji.
* Kriterijumi filtriranja: Naziv dela, količina, stanje zaliha, kategorija.

Tabelarni prikazi za evidenciju tehničara i radnika

**Tabelarni prikaz tehničara i radnika:**

* Prikazuje listu svih tehničara i radnika u sistemu, uključujući informacije kao što su ime, prezime, identifikacioni broj, kvalifikacije i radno iskustvo.
* Kriterijumi filtriranja: Ime, prezime, kvalifikacije, radno iskustvo.

**Tabelarni prikaz radnih zadataka:**

* Prikazuje listu radnih zadataka obavljenih od strane tehničara, uključujući informacije o zadacima, vremenu provedenom na zadacima i rezultatima.
* Kriterijumi filtriranja: Tehničar, zadatak, datum, trajanje, status.

Tabelarni prikazi za administraciju i konfiguraciju

**Tabelarni prikaz korisnika:**

* Prikazuje listu svih korisnika u sistemu, uključujući informacije kao što su korisničko ime, uloga, i status naloga.
* Kriterijumi filtriranja: Korisničko ime, uloga, status naloga.

**Tabelarni prikaz sistema:**

* Prikazuje listu svih sistemskih konfiguracija i postavki, uključujući informacije kao što su naziv postavke, vrednost i opis.
* Kriterijumi filtriranja: Naziv postavke, kategorija, vrednost.

Zaključak

Tabelarni prikazi sa kriterijumima filtriranja omogućavaju korisnicima da brzo pronađu i analiziraju relevantne podatke u sistemu evidencije popravki aviona. Ovi prikazi doprinose efikasnosti i produktivnosti korisnika, omogućavajući im da brzo pristupe potrebnim informacijama i donose informisane odluke na osnovu tačnih i ažurnih podataka.

### **Spisak potrebnih štampi dokumenata**

U svrhu olakšanja procesa donošenja odluka i analize podataka u sistemu evidencije popravki aviona definisane su različite vrste izveštaja i funkcija za izvoz podataka. Ove funkcionalnosti omogućavaju korisnicima da generišu različite vrste izveštaja i izvode podatke iz sistema radi daljeg pregleda i obrade.

Štampa dokumenata i izveštaja

**Izveštaj o popravkama aviona:**

* Izveštaj koji sadrži informacije o svim popravkama aviona u određenom periodu, uključujući detalje o avionima, popravkama, zadacima tehničara i rezultatima popravki.
* Omogućava pregled istorije popravki za svaki avion.

**Izveštaj o delovima i materijalima:**

* Izveštaj koji sadrži podatke o delovima i materijalima, uključujući informacije o dostupnosti, zalihama, cenama i upotrebi u popravkama.
* Omogućava praćenje zaliha i potrošnje materijala.

**Izveštaj o tehničarima i radnicima:**

* Izveštaj koji prikazuje podatke o tehničarima i radnicima, uključujući informacije o kvalifikacijama, radnom iskustvu i radnim zadacima.
* Omogućava praćenje performansi tehničara i njihove efikasnosti.

Statistički izveštaji za podršku odlučivanju

**Izveštaj o performansama aviona:**

* Izveštaj koji analizira performanse aviona na osnovu podataka o popravkama, učestalosti kvarova i popravkama, kao i stanju delova i materijala.
* Pomaže u donošenju odluka o održavanju aviona i njihovoj upotrebi.

**Izveštaj o trendovima popravki:**

* Izveštaj koji analizira trendove u popravkama aviona, uključujući uobičajene kvarove, učestalost popravki i trajanje popravki.
* Pomaže u identifikaciji ključnih problema i planiranju proaktivnog održavanja.

**Izveštaj o troškovima održavanja:**

* Izveštaj koji prikazuje podatke o troškovima održavanja aviona, uključujući troškove delova, materijala i rada.
* Pomaže u donošenju odluka o upravljanju budžetom i troškovima.

Eksport podataka

**Eksport podataka u CSV/Excel format:**

* Omogućava korisnicima izvoz podataka o popravkama, delovima, tehničarima i drugim informacijama u CSV ili Excel format.
* Ovi formati su pogodni za dalju obradu, analizu i izveštavanje.

**Eksport podataka u PDF format:**

* Omogućava korisnicima izvoz podataka o popravkama, delovima i drugim informacijama u PDF format.
* Ovaj format je pogodan za deljenje i štampu dokumenata.

Zaključak

Funkcionalnosti za štampu dokumenata, statističke izveštaje i eksport podataka iz sistema evidencije popravki aviona su ključne za podršku donošenja odluka i analizu podataka. Ove funkcionalnosti omogućavaju korisnicima pristup preciznim informacijama o popravkama i stanju aviona, što im pomaže u optimizaciji operacija i poboljšanju kvaliteta rada.

### **Spisak poslovnih pravila, automatizama i ograničenja koje je potrebno ugaditi u softver**

Ovo poglavlje obuhvata definisane smernice i parametre koji regulišu ponašanje sistema evidencije popravki aviona. Pravila, automatizmi i ograničenja zauzimaju centralnu ulogu u osiguranju usklađenosti sa zakonima i standardima, kao i za povećanje efikasnosti i preciznosti procesa unutar sistema.

Poslovna pravila

**Pravilo o praćenju zaliha delova i materijala:**

* Sistem treba da prati zalihe delova i materijala, uključujući postavljanje minimalnih nivoa zaliha svakog artikla.
* Omogućava obaveštavanje korisnika kada zalihe dostignu kritične nivoe.

**Pravilo o dodeli zadataka tehničarima:**

* Pravila o dodeli zadataka treba da osiguraju ravnomernu raspodelu zadataka među tehničarima na osnovu njihove dostupnosti i kvalifikacija.
* Treba da se uspostave procedure za eskalaciju zadataka ako nije moguće dodeliti zadatak odgovarajućem tehničaru.

**Pravilo o evidentiranju i autorizaciji popravki:**

* Prikazivanje popravki treba da zahteva autentifikaciju tehničara ili menadžera održavanja pre nego što se popravka evidentira kao završena.
* Autorizacija treba da uključuje proveru ispravnosti obavljenih radova i korišćenih delova.

Automatizmi

**Automatizacija izveštaja o popravkama:**

* Sistem treba automatski da generiše periodične izveštaje o popravkama aviona, uključujući izveštaje o performansama, trendovima i troškovima održavanja.
* Izveštaji se mogu slati korisnicima putem e-mail-a na redovnoj osnovi.

**Automatizacija poručivanja zaliha:**

* Kada zalihe delova i materijala padnu ispod unapred definisanog nivoa, sistem može automatski generisati porudžbinu prema relevantnom dobavljaču.
* Uključuje slanje obaveštenja logističarima i menadžerima.

**Automatizacija praćenja rokova:**

* Sistem treba automatski da prati rokove za popravke i obaveštava tehničare i menadžere kada se približavaju rokovi ili ako su prekoračeni.

Ograničenja

**Ograničenja pristupa:**

* Sistem treba da ima stroga ograničenja pristupa, tako da korisnici mogu pristupati samo informacijama i funkcijama koje su u skladu sa njihovom ulogom i ovlašćenjima.
* Ograničenja treba da se primenjuju na sve nivoe korisnika, uključujući administratore, tehničare, menadžere i logističare.

**Ograničenja na izmene podataka:**

* Sistem treba da ograniči izmene podataka o popravkama, delovima i tehničarima samo na ovlašćene korisnike.
* Svi pokušaji izmene treba da budu praćeni i evidentirani.

**Ograničenja na unos novih aviona i tehničara:**

* Unos novih aviona i tehničara treba da bude ograničen na ovlašćene korisnike, kao što su menadžeri ili administratori.
* Podaci o avionima i tehničarima treba da se unose samo kada su ispunjeni svi potrebni uslovi, kao što su sertifikacije i provere.

Zaključak

Ugradnja poslovnih pravila, automatizama i ograničenja u sistem evidencije popravki aviona osigurava usklađenost sa standardima i pravilima, kao i efikasno upravljanje popravkama aviona. Ove funkcionalnosti pomažu korisnicima da se pridržavaju procedura i pravila, poboljšavajući kvalitet rada i sigurnost u održavanju aviona.

### **Potrebne karakteristike korisničkog interfejsa i ostali nefunkcionalni zahtevi**

Karakteristike korisničkog interfejsa i nefunkcionalni zahtevi su neophodni za pružanje kvalitetnog korisničkog iskustva u sistemu evidencije popravki aviona. Fokus na jednostavnosti, doslednosti, pristupačnosti i performansama doprinosi efikasnosti rada korisnika i uspešnosti sistema u celini.

Karakteristike korisničkog interfejsa

**Jednostavnost i intuitivnost:**

* Korisnički interfejs treba da bude jednostavan za upotrebu, sa jasnim i intuitivnim navigacionim strukturama.
* Forme i meniji trebaju biti organizovani na logičan način, kako bi korisnici lako pronašli informacije koje traže.

**Doslednost:**

* Dizajn interfejsa treba da bude dosledan u svim delovima softvera, uključujući upotrebu boja, fontova i ikona.
* Doslednost u interfejsu olakšava korisnicima da se brzo prilagode radu sa softverom.

**Prilagodljivost:**

* Interfejs treba da omogućava prilagođavanje preferencijama korisnika, kao što su promene izgleda ekrana i rasporeda elemenata.
* Prilagodljivost povećava korisničku udobnost i produktivnost.

**Pristupačnost:**

* Interfejs treba da bude pristupačan osobama sa različitim invaliditetima, uključujući podršku za čitače ekrana, kontrastne boje i opcije za povećanje fonta.
* Ova karakteristika osigurava da svi korisnici mogu koristiti softver bez problema.

**Vizuelna hijerarhija:**

* Korisnički interfejs treba da koristi vizuelne elemente kao što su boje, veličine i raspored za naglašavanje važnih informacija i akcija.
* Vizuelna hijerarhija olakšava korisnicima da brzo pronađu ključne podatke.

Ostali nefunkcionalni zahtevi

**Performanse:**

* Softver treba da bude brz i responzivan, sa minimalnim vremenom učitavanja za stranice i forme.
* Performanse su važne za efikasnost rada i zadovoljstvo korisnika.

**Pouzdanost:**

* Softver treba da bude stabilan i otporan na greške, sa minimalnim padovima sistema.
* Podaci se trebaju redovno rezervisati kako bi se sprečio gubitak informacija.

**Bezbednost:**

* Softver treba da obezbedi zaštitu podataka korisnika i aviona kroz šifrovanje i kontrolu pristupa.
* Ova funkcionalnost osigurava poverljivost i integritet informacija.

**Skalabilnost:**

* Sistem treba da bude skalabilan kako bi podržao rast broja korisnika, aviona i podataka tokom vremena.
* Skalabilnost omogućava sistemu da se prilagodi potrebama organizacije.

**Kompatibilnost:**

* Softver treba da bude kompatibilan sa različitim uređajima i operativnim sistemima kako bi korisnici mogli pristupati sistemu sa različitih platformi.
* Kompatibilnost povećava fleksibilnost korišćenja softvera.

## **Elementi projektnog planiranja**

### **Granice (scope) projekta**

Jasno definisanje opsega projekta pomaže razvojnom timu da se usredsredi na krucijalne zahteve i funkcionalnosti, dok istovremeno razmatra koje funkcije nisu uključene u trenutni projekat. To omogućava efikasnije upravljanje projektom i resursima.

U okviru opsega projekta:

**Login forma i personalizovani meni u odnosu na profil korisnika:**

* Biće implementirana forma za prijavu korisnika koja omogućava personalizovani meni prema ulogama i profilima korisnika.

**Barem 1 admin profil korisnika:**

* U sistem će biti dodan admin profil korisnika koji ima pristup svim funkcijama i može upravljati drugim korisničkim nalozima.

**Ekranska forma za unos sa combo boxom koji dinamički čita podatke:**

* Implementacija forme za unos podataka sa dinamičkim combo boxom koji učitava podatke iz baze podataka.

**Implementacija osnovnih validacija u okviru unosa podataka:**

* Biće implementirane osnovne validacije podataka tokom unosa, uključujući proveru potpunosti, ispravnosti i jedinstvenosti unetih podataka.

**Implementacija poslovne logike:**

* Poslovna logika, kao što su ograničenja ili automatizmi, biće integrisana u sistem tokom unosa podataka.

**Ekranska forma sa tabelarnim prikazom i filterom:**

* Biće pružena forma sa tabelarnim prikazom podataka i filterom za pretragu i pregled informacija.

**Funkcije izmene i brisanja podataka:**

* Omogućiće se funkcije za izmene i brisanje podataka kako bi se osiguralo ažuriranje informacija u sistemu.

**Štampa spiska podataka:**

* Korisnici će imati mogućnost štampe spiska podataka prema izabranim kriterijumima.

**Parametarska štampa jednog ili manje grupe zapisa:**

* Omogućiće se štampanje zapisa na osnovu vrednosti filtera, uključujući parametarsku štampu jedne ili manje grupe zapisa.

Izvan opsega projekta:

**Napredna analitika i izveštaji:**

* Napredna analitika, uključujući složene izveštaje i vizualizacije podataka, neće biti obuhvaćeni u ovoj fazi projekta.

**Integracija sa eksternim sistemima:**

* Integracija sa eksternim sistemima, kao što su sistemi za održavanje aviona ili menadžment zaliha, nije uključena u opseg projekta.

**Mobilna aplikacija:**

* Razvoj i podrška mobilnih aplikacija za pristup sistemu neće biti obuhvaćeni u ovoj fazi projekta.

**Napredna korisnička personalizacija:**

* Napredne funkcije za personalizaciju korisničkog iskustva, kao što su korisničke teme ili proširene opcije konfiguracije, neće biti realizovane u ovoj fazi.

**Višejezičnost:**

* Podrška za višejezičnost i lokalizaciju softvera nije uključena u opseg projekta.

### **Kratak opis tehnologija, programskih jezika i razvojnih alata koji će biti korišćeni u radu**

* C# (C-Sharp):

Opis: C# je objektno-orijentisani programski jezik razvijen od strane Microsofta. Namenjen je za razvoj raznovrsnih aplikacija, uključujući desktop, web i mobilne aplikacije.

Primena u projektu: Projekat je razvijen koristeći C#, što omogućava visok stepen fleksibilnosti i efikasnosti u razvoju.

* Microsoft Visual Basic:

Opis: Visual Basic je događajno-orijentisani programski jezik i razvojno okruženje razvijeno od strane Microsofta. Uglavnom se koristi za izradu Windows aplikacija.

Primena u projektu: Visual Basic je korišćen za određene delove projekta, pružajući brz i jednostavan način za razvoj korisničkog interfejsa i poslovne logike.

* SQL Server Express:

Opis: SQL Server Express je besplatna verzija Microsoft SQL Servera, koja pruža osnovne funkcionalnosti za upravljanje bazama podataka. Idealna je za manje aplikacije i projekte.

Primena u projektu: SQL Server Express je korišćen kao instanca DBMS-a za upravljanje bazama podataka projekta.

* SQL Server Management Studio (SSMS):

Opis: SSMS je integrisano razvojno okruženje za upravljanje SQL Server infrastrukturom. Pruža alate za konfiguraciju, upravljanje i administraciju baza podataka.

Primena u projektu: SSMS je korišćen za administraciju, razvoj i održavanje baza podataka, kao i za izvršavanje SQL upita.

* Microsoft Word:

Opis: Microsoft Word je program za obradu teksta razvijen od strane Microsofta. Omogućava pisanje, uređivanje i formatiranje tekstualnih dokumenata.

Primena u projektu: Microsoft Word je korišćen za pisanje dokumentacije projekta, uključujući tehničku dokumentaciju, korisničke priručnike i ostale relevantne materijale.

Ove tehnologije i alati omogućili su efikasan razvoj, upravljanje i dokumentovanje projekta, pružajući stabilnu osnovu za realizaciju svih funkcionalnosti i ciljeva projekta.

# **POSLOVNA SISTEMSKA ANALIZA I DIZAJN SOFTVERSKOG RЕŠENJA**

## **Model poslovnih procesa**

### **Algoritamski model poslovnih procesa**

A diagram of a company

Description automatically generated

*Slika 1. – Algoritamski model poslovnih procesa*

### **Spisak skladišta podataka**

U OKVIRU POSLOVNOG PROCESA EVIDENCIJE POPRAVKI AVIONA

Inventar

Operateri

Podaci o održavanju / popravci letelice

### **Analiza jednog poslovnog dokumenta**

A yellow and white box with black text

Description automatically generated

*Slika 2. – Primer poslovnog dokumenta (Radni nalog)[5]*

|  |  |  |  |
| --- | --- | --- | --- |
| **ELEMENTARNI PODATAK** | **TIP PODATKA** | **OBAVEZAN UNOS** | **DOMEN** |
| Naziv dokumenta | String | Da |  |
| Broj naloga | String | Da |  |
| Datum izdavanja | Date | Da |  |
| Tip letelice | String | Da |  |
| Registracija | String | Da |  |
| Serijski broj | String | Da |  |
| Vrsta zahteva | String | Da |  |
| Opis problema | String | Ne |  |
| Deo | String | Ne | Inventar delova |
| Količina | Integer | Ne |  |
| Ime i prezime operatera | String | Da | Evidencija operatera |
| Pozicija operatera | String | Da |  |
| Kontakt operatera | String | Da |  |
| Datum početka | Date | Da |  |
| Datum završetka | Date | Da |  |
| Komentari | String | Ne |  |
| Potpis operatera | Signature | Da |  |
| Potpis nadzornika | Signature | Da |  |
| Datum zatvaranja | Date | Da |  |

*Tabela 1. – Elementarni podaci poslovnog dokumenta*

**Sintaksna analiza poslovnog dokumenta**:

RadniNalog<<

<Broj naloga,

Datum izdavanja>,

<Podaci o letelici<

Tip letelice,

Registracija,

Serijski broj>>,

<Opis zahteva<

Vrsta zahteva,

Opis problema>>,

<Inventar<

Deo,

Količina>>,

<Operater<

Ime i prezime operatera,

Pozicija operatera,

Kontakt operatera>>,

<Detalji<

Datum početka,

Datum završetka,

Komentari,

Potpis operatera,

Potpis nadzornika,

Datum zatvaranja>>

>>

## **Dizajn softverskih funkcija**

### **Tabela preslikavanja primitivnih procesa u softverske funkcije**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| PRIMITIVNI POSLOVNI PROCES | Poslovni učesnik | Actor (profil korisnika koji će koristiti softverske funkcije) | Softverska funkcija 1. prioriteta | Softverska funkcija 2. prioriteta | Softverska funkcija preduslov |
| Pregled letelice | Osoblje | n/a | n/a | n/a | n/a |
| Prijem zahteva za redovno održavanje / identifikacija problema | Osoblje | Administracija | Unos podataka sa zahteva | Tabelarni prikaz zahteva i filter po registraciji letelice | Prijem izveštaja o pregledu letelice |
| Kreiranje radnog naloga | Administracija | Administracija | Unos podataka o radnom nalogu | Tabelarni prikaz radnih naloga i filter po registraciji letelice | Prijem zahteva za redovno održavanje / identifikacija problema |
| Izvršenje radnog naloga | Osoblje | n/a | n/a | n/a | Kreiranje radnog naloga |
| Praćenje popravke | Osoblje | Administracija | Tabelarni prikaz tekućih popravki | Filter tekućih popravki po registraciji letelice | Izvršenje radnog naloga |
| Zatvaranje popravke | Osoblje | Administracija | Unos podataka o završenim popravkama | Tabelarni prikaz završenih popravki, filter po registraciji letelice, štampa izveštaja | Praćenje popravke |

### **USE CASE dijagram softverskih funkcija**

A diagram of a company

Description automatically generated with medium confidence

*Slika 3. – USE CASE dijagram softverskih funkcija*

### **Specifikacija slučaja korišćenja za unos i tabelarni prikaz podataka**

A screenshot of a computer

Description automatically generated

*Slika 4. – Specifikacija slučaja korišćenja za softversku funkciju unosa podataka o zatvaranju popravke – Pre-Conditions*

A screenshot of a computer

Description automatically generated

*Slika 5. – Specifikacija slučaja korišćenja za softversku funkciju unosa podataka o zatvaranju popravke – Action steps*

A screenshot of a computer

Description automatically generated

*Slika 6. – Specifikacija slučaja korišćenja za softversku funkciju unosa podataka o zatvaranju popravke – Extension Points*

A screenshot of a computer

Description automatically generated

*Slika 7. – Specifikacija slučaja korišćenja za softversku funkciju unosa podataka o zatvaranju popravke – Exceptions*

A screenshot of a computer

Description automatically generated

*Slika 8. – Specifikacija slučaja korišćenja za softversku funkciju unosa podataka o zatvaranju popravke – Post-Conditions*

A screenshot of a computer

Description automatically generated

*Slika 9. – Specifikacija slučaja korišćenja za softversku funkciju tabelarnog prikaza podataka o zatvaranju popravke – Pre-Conditions*

A screenshot of a computer

Description automatically generated

*Slika 10. – Specifikacija slučaja korišćenja za softversku funkciju tabelarnog prikaza podataka o zatvaranju popravke – Action steps*

A screenshot of a computer

Description automatically generated

*Slika 11. – Specifikacija slučaja korišćenja za softversku funkciju tabelarnog prikaza podataka o zatvaranju popravke – Extension Points*

A screenshot of a computer

Description automatically generated

*Slika 12. – Specifikacija slučaja korišćenja za softversku funkciju tabelarnog prikaza podataka o zatvaranju popravke – Exceptions*

A screenshot of a computer

Description automatically generated

*Slika 13. – Specifikacija slučaja korišćenja za softversku funkciju tabelarnog prikaza podataka o zatvaranju popravke – Post-Conditions*

## **Dizajn modela podataka**

### **Konceptualni model podataka**

A screenshot of a computer screen

Description automatically generated

*Slika 14. – Konceptualni model podataka*

### **Fizički (relacioni) model podataka**

A screenshot of a computer program

Description automatically generated

*Slika 15 – Fizički model podataka*

# **IMPLEMENTACIJA, TESTIRANJE I DOKUMENTOVANJE REALIZOVANOG SOFTVERA**

## **Korisničko i tehničko uputstvo**

### **Korisničko uputstvo**

A screenshot of a computer

Description automatically generated

*Slika 16. – Početna stranica*

Klikom na link „PRIJAVA“, korisnik pristupa Login stranici.

A screenshot of a computer

Description automatically generated

*Slika 17. – Login forma*

Nakon uspešnog logovanja prikazuje se poruka dobrodošlice i korisniku su dostupne opcije aplikacije.

A blue and white sign

Description automatically generated

*Slika 18. – Početna strana (admin)*

U padajućim menijima „LETELICE“ i „OPERATERI“ se nalaze sledeće funkcionalnosti.

A screenshot of a computer

Description automatically generated

*Slika 19 – Opcije padajućeg menija „LETELICE“*

A screenshot of a computer

Description automatically generated

*Slika 20 – Opcije padajućeg menija „OPERATERI“*

Korisnik ima mogućnost pristupa tabelarnom prikazu svih letelica ili filtriranom prikazu letelica po nazivu.

A screenshot of a computer

Description automatically generated

*Slika 21. – Tabelarni prikaz zapisa svih letelica u bazi podataka*

*A screenshot of a computer

Description automatically generated*

*Slika 22. – Filtrirani prikaz zapisa letelice*

Unos letelice se vrši upisivanjem odgovarajućih vrednosti u ponuđena polja i klikom na dugme „SNIMI“.

A screenshot of a computer

Description automatically generated

*Slika 23. – Forma za unos zapisa o letelici*

Opcija „Izmena i brisanje letelice“ preusmerava korisnika na tabelarni prikaz svih letelica pored kojih se nalazi dugme koje preusmerava na stranicu za detaljni prikaz odabrane letelice na kojoj korisnik može izvršiti izmenu i brisanje.

A screenshot of a computer

Description automatically generated

*Slika 24. – Spisak zapisa o letelicama za ažuriranje*

A screenshot of a computer

Description automatically generated

*Slika 25. – Detaljni prikaz ažuriranja zapisa o letelici*

Štampa spiska letelica daje spisak svih zapisa o letelicama u „printer friendly“ formi dok parametarska štampa letelice prvenstveno traži naziv letelice koje želimo da štampamo i prikazuje je na isti način.

A screen shot of a document

Description automatically generated

*Slika 26. – Štampa spiska letelica*



*Slika 27. – Forma za unos naziva letelice za parametarsku štampu*

A close up of a document

Description automatically generated

*Slika 28. – Parametarska štampa na osnovu unesenog naziva letelice*

Spisak operatera nam daje uvid u sve zapise o operaterima i kojim su letelicama dodeljeni. Takođe je moguć filtrirani prikaz po prezimenu operatera.

A screenshot of a computer

Description automatically generated

*Slika 29. – Spisak operatera*

Unos operatera omogućava stvaranje novog zapisa o operateru

A screenshot of a computer

Description automatically generated

*Slika 30. – Forma za unos operatera*

Pritiskom na link „ODJAVA“ u gornjem desnom uglu korisnik biva izlogovan i preusmeren na početnu stranu aplikacije.

### **Tehničko uputstvo**

Prvi korak u ostvarivanju uspešnog rada aplikacije jeste dizanje baze podataka pomoću SQL Server Management Studio alata. Klikom na dugme “New Query” se otvara tekst editor gde korisnik treba da nalepi skripte za dizanje baze podataka kao i skripte za stored procedure iz foldera 1. Klikom na dugme “Execute Query” se ove skripte izvršavaju i naša baza podataka je spremna za rad.

Za pravilan rad aplikacije, potrebno je pokrenuti web servise pre pokretanja same aplikacije da bi se zadata poslovna pravila na odgovarajući način primenila pri unosu novih zapisa u bazu podataka. Ovi web servisi se nalaze u folderu 3. Potrebno je ući u svaki i pokrenuti solution u Visual Basic-u nakon čega je potrebno pokrenuti oba servisa klikom na zeleno start dugme u gornjoj vertikalnoj traci sa alatima.

Sama aplikacija se pokreće otvaranjem solution-a iz foldera 4 pod nazivom “KorisničkiInterfejs”. Proces startovanja je isti kao i za web servise.

Korisnik mora imati instalirane alate Microsoft Visual Basic i SQL Server Management Studio kao i instancu SQL Server Express.

## **Izveštaj o testiranju softvera**

### **Test slučajevi ispravnog rada korisnika i prikaz rezultata testiranja**

Test slučajevi ispravnog korisničkog rukovanja su prikazani na slikama u korisničkom uputstvu gde se jasno može videti funkcionalnost aplikacije u redovnim uslovima korišćenja.

### **Test slučajevi neispravnog rada korisnika i prikaz rezultata testiranja**

A screenshot of a computer

Description automatically generated

*Slika 31. – Neuspešan unos zapisa o operateru zbog postojanja zapisa sa istim JMBG u bazi podataka*

## **Dokumentovanje implementacije softverskog rešenja**

### **Modeli opisa implementacije rešenja**

#### **Dijagram komponenti**

A diagram of a computer

Description automatically generated

*Slika 32. – Dijagram komponenti*

#### **Djagram klasa**

Na dijagramu klasa prikazane su klase za rad sa podacima koje se odnose na tabelu operater, letelica i korisnik u okviru baze podataka. Tabele operater i letelica predstavljene su kroz tri klase:

A screenshot of a computer

Description automatically generated

*Slika 33. – Dijagram klasa za tabelu operater*

A screenshot of a computer code

Description automatically generated

*Slika 34. – Dijagram klasa za tabele letelica i korisnik*

#### **Dijagram sekvenci za osnovni scenario unosa podataka i tabelarnog prikaza sa filterom**

A screenshot of a computer screen

Description automatically generated

*Slika 35. – DIjagram sekvenci za tabelarni prikaz letelica*

### **Tabelarni prikaz komponenti**

### **SQL script**

|  |
| --- |
| **USE** **[**master**]**  **GO**  **CREATE** **DATABASE** **[**EvidencijaPopravkiAviona**]**  **GO**  **USE** **[**EvidencijaPopravkiAviona**]**  **GO**  **CREATE** **TABLE** **[**dbo**].[**OPERATER**](**  **[**JMBG**]** **[**Char**]** **(**13**)** **NOT** **NULL,**  **[**Prezime**]** **[**nvarchar**](**40**)** **NOT** **NULL,**  **[**Ime**]** **[**nvarchar**](**40**)** **NOT** **NULL,**  **[**IDLetelice**]** **[**char**](**10**)** **NOT** **NULL**  **)**  **GO**  **ALTER** **TABLE** **[**dbo**].[**OPERATER**]**  **ADD** **CONSTRAINT** **[**PK\_OPERATER**]** **PRIMARY** **KEY** **CLUSTERED**  **(**  **[**JMBG**]** **ASC**  **)**  **GO**  **CREATE** **TABLE** **[**dbo**].[**LETELICA**](**  **[**RegBr**]** **[**Char**]** **(**10**)** **NOT** **NULL,**  **[**Naziv**]** **[**nvarchar**](**40**)** **NOT** **NULL,**  **)**  **GO**  **ALTER** **TABLE** **[**dbo**].[**LETELICA**]**  **ADD** **CONSTRAINT** **[**PK\_LETELICA**]** **PRIMARY** **KEY** **CLUSTERED**  **(**  **[**RegBr**]** **ASC**  **)**  **GO**  **ALTER** **TABLE** **[**dbo**].[**OPERATER**]** **ADD** **CONSTRAINT**  **[**FK\_OPERATER\_LETELICA**]** **FOREIGN** **KEY([**IDLetelice**])**  **REFERENCES** **[**dbo**].[**LETELICA**]** **([**RegBr**])**  **ON** **UPDATE** **CASCADE**  **GO**  **CREATE** **TABLE** **[**dbo**].[**Korisnik**](**  **[**ID**]** **[**int**]** **IDENTITY(**1**,**1**)** **NOT** **NULL,**  **[**Prezime**]** **[**nvarchar**](**40**)** **NOT** **NULL,**  **[**Ime**]** **[**nvarchar**](**30**)** **NOT** **NULL,**  **[**KorisnickoIme**]** **[**nvarchar**](**20**)** **NOT** **NULL,**  **[**Sifra**]** **[**nvarchar**](**30**)** **NOT** **NULL,**  **[**Status**]** **[**nvarchar**](**10**)** **NOT** **NULL**  **)**  **GO**  **INSERT** **INTO** **[**dbo**].[**Korisnik**]** **(**Prezime**,** Ime**,** KorisnickoIme**,** Sifra**,** Status**)**  **VALUES** **(**'Smiljanić'**,** 'Marko'**,** 'msmiljanic'**,** 'ms'**,** 'admin'**);**  **INSERT** **INTO** LETELICA **(**RegBr**,** Naziv**)**  **VALUES** **(**'N123AB'**,** 'H - C130 Globemaster'**);**  **INSERT** **INTO** LETELICA **(**RegBr**,** Naziv**)**  **VALUES** **(**'G-BHXY'**,** 'M - Airbus A320'**);**  **INSERT** **INTO** LETELICA **(**RegBr**,** Naziv**)**  **VALUES** **(**'D-AFTR'**,** 'L - Cessna 172'**);**  **INSERT** **INTO** OPERATER **(**JMBG**,** Prezime**,** Ime**,** IDLetelice**)**  **VALUES**  **(**'0101985850001'**,** 'Petrović'**,** 'Milan'**,** 'N123AB'**),**  **(**'1507955850023'**,** 'Jovanović'**,** 'Nikola'**,** 'N123AB'**),**  **(**'2312935850045'**,** 'Đorđević'**,** 'Marko'**,** 'N123AB'**),**  **(**'0804935850002'**,** 'Lazić'**,** 'Ana'**,** 'G-BHXY'**),**  **(**'0912885850015'**,** 'Nikolić'**,** 'Jovana'**,** 'G-BHXY'**),**  **(**'2511955850034'**,** 'Stanković'**,** 'Ivana'**,** 'D-AFTR'**);** |

*Listing 1. – SQL skripta*

### **Delovi koda sa objašnjenjima po slojevima**

#### **Sloj za rad sa podacima**

|  |
| --- |
| **public** bool OtvoriKonekciju**()**  **{**  bool uspeh**;**  pKonekcija **=** **new** SqlConnection**();**  pKonekcija**.**ConnectionString **=** DajStringKonekcije**();**  **try**  **{**  pKonekcija**.**Open**();**  uspeh **=** **true;**  **}**  **catch**  **{**  uspeh **=** **false;**  **}**  **return** uspeh**;**  **}**  **public** SqlConnection DajKonekciju**()**  **{**  **return** pKonekcija**;**  **}**  **public** void ZatvoriKonekciju**()**  **{**  pPutanjaBaze **=** ""**;**  pKonekcija**.**Close**();**  pKonekcija**.**Dispose**();**  **}** |

*Listing 2. – Metoda za otvaranje konekcije iz tehnološke klase za rad sa bazom podataka clsKonekcija*

|  |
| --- |
| **public** bool SnimiNovogOperatera**(**clsOperater objNoviOperater**)**  **{**  // LOKALNE PROMENLJIVE UVEK NA VRHU  int brojSlogova **=**0**;**  SqlConnection Veza **=** **new** SqlConnection**(**pStringKonekcije**);**  Veza**.**Open**();**  SqlCommand Komanda **=** **new** SqlCommand**(**"DodajNovogOperatera"**,** Veza**);**  Komanda**.**CommandType **=** CommandType**.**StoredProcedure**;**  Komanda**.**Parameters**.**Add**(**"@JMBG"**,** SqlDbType**.**Char**).**Value **=** objNoviOperater**.**JMBG **;**  Komanda**.**Parameters**.**Add**(**"@Prezime"**,** SqlDbType**.**NVarChar **).**Value **=** objNoviOperater**.**Prezime**;**  Komanda**.**Parameters**.**Add**(**"@Ime"**,** SqlDbType**.**NVarChar**).**Value **=** objNoviOperater**.**Ime**;**  Komanda**.**Parameters**.**Add**(**"@IDLetelice"**,** SqlDbType**.**Char**).**Value **=** objNoviOperater**.**Letelica**.**RegBr**;**      brojSlogova **=** Komanda**.**ExecuteNonQuery**();**  Veza**.**Close**();**  Veza**.**Dispose**();**    **return** **(**brojSlogova **>** 0**);**  **}** |

*Listing 3. – Metoda za snimanje novog zapisa u tabeli operater klase podataka clsOperaterDB*

|  |
| --- |
| **public** class clsOperater  **{**  // atributi  **private** string pJMBG**;**  **private** string pPrezime**;**  **private** string pIme**;**  **private** clsLetelica objLetelica**;**  // property  **public** string JMBG  **{**  **get** **{** **return** pJMBG**;** **}**  **set** **{** pJMBG **=** **value;** **}**  **}**    **public** string Prezime  **{**  **get** **{** **return** pPrezime**;** **}**  **set** **{** pPrezime **=** **value;** **}**  **}**    **public** string Ime  **{**  **get** **{** **return** pIme**;** **}**  **set** **{** pIme **=** **value;** **}**  **}**    **public** clsLetelica Letelica  **{**  **get** **{** **return** objLetelica**;** **}**  **set** **{** objLetelica **=** **value;** **}**  **}**  **}** |

*Listing 4. – Klasa podataka clsOperater*

|  |
| --- |
| **public** void DodajElementListe**(**clsOperater objNoviOperater**)**  **{**  pListaOperatera**.**Add**(**objNoviOperater**);**  **}**  **public** void ObrisiElementListe**(**clsOperater objOperaterZaBrisanje**)**  **{**  pListaOperatera**.**Remove**(**objOperaterZaBrisanje**);**  **}**  **public** void ObrisiElementNaPoziciji**(**int pozicija**)**  **{**  pListaOperatera**.**RemoveAt**(**pozicija**);**  **}**  **public** void IzmeniElementListe**(**clsOperater objStariOperater**,** clsOperater objNoviOperater**)**  **{**  int indexStarogOperatera **=** 0**;**  indexStarogOperatera **=** pListaOperatera**.**IndexOf**(**objNoviOperater**);**  pListaOperatera**.**RemoveAt**(**indexStarogOperatera**);**  pListaOperatera**.**Insert**(**indexStarogOperatera**,** objNoviOperater**);**  **}** |

*Listing 5. – Metode za rad sa listama u klasi podataka clsOperaterLista*

#### **Sloj servisa ili DTO**

|  |
| --- |
| **public** class clsMaper  **{**  // atributi  **private** string pStringKonekcije**;**  // property  // konstruktor  **public** clsMaper**(**string NoviStringKonekcije**)**  **{**  pStringKonekcije **=** NoviStringKonekcije**;**  **}**  **public** string DajRegBrLeteliceZaWebServis**(**string RegBrLeteliceIzBazePodataka**)**  **{**  string RegBrLeteliceWS **=** ""**;**  clsLetelicaDB objZvanjeDB **=** **new** clsLetelicaDB**(**pStringKonekcije**);**  string nazivLeteliceIzBazePodataka **=** objZvanjeDB**.**DajNazivPremaRegBr**(**RegBrLeteliceIzBazePodataka**);**  // Reg Br Letelice za web servis je prvo slovo naziva Letelice,  // "H" - heavy, odnosno teška letelica poput Airbus A380, C130 Globemaster, itd. -> U našem sistemu se beleži kao H - C130 Globemaster  // "M" - medium, odnosno srednja letelica poput Boeing 737, Airbus A320, itd.  // "L" - light, odnosno laka letelica poput Cessna 172, Cirrus SR22, itd.  RegBrLeteliceWS **=** nazivLeteliceIzBazePodataka**[**0**].**ToString**();**  **return** RegBrLeteliceWS**;**  **}**  **}** |

*Listing 6. – Klasa za mapiranje clsMaper*

#### **Sloj poslovne logike**

|  |
| --- |
| **public** bool DaLiImaMestaZaNovogOperatera**(**string IdLeteliceIzBazePodataka**)**  **{**  // POSLOVNO PRAVILO:  // Veličina letelice utiče na broj operatera koji mogu da je opslužuju  //Teška letelica = 3 operatera  //Srednja letelica = 2 operatera  //Laka letelica = 1 operater  bool imaMesta **=** **false;**  // ################################################################  // 1. IZRACUNAVANJE TRENUTNOG BROJA OPERATERA U BAZI PODATAKA  //  int UkupnoOperatera **=** 0**;**  clsOperaterDB objOperaterDB **=** **new** clsOperaterDB**(**pStringKonekcije**);**  UkupnoOperatera **=** objOperaterDB**.**DajUkupnoOperateraZaLetelicu**(**IdLeteliceIzBazePodataka**);**  // ################################################################  // 2. MAPIRANJE SLOJEVA - uskladjivanje Reg Br Letelice iz raznih delova programa  // Web servis ima h - heavy (teška letelica), baza podataka ima reg br - H - C130 Globemaster  string RegBrLeteliceWS **=** ""**;**  clsMaper objMaper **=** **new** clsMaper**(**pStringKonekcije**);**  RegBrLeteliceWS **=** objMaper**.**DajRegBrLeteliceZaWebServis**(**IdLeteliceIzBazePodataka**);**  // ################################################################  // 3. IZDVAJANJE MAX BROJA OPERATERA ZA ODGOVARAJUĆU LETELICU  WSOgranicenja**.**Service1 objOgranicenja **=** **new** WSOgranicenja**.**Service1**();**  int MaxBrojOperatera **=** objOgranicenja**.**DajMaxBrojOperatera**(**RegBrLeteliceWS**);**  // ################################################################  // 4. UPOREDJIVANJE TRENUTNOG BROJA I MAX BROJA OPERATERA  **if** **(**UkupnoOperatera **<** MaxBrojOperatera**)**  **{**  imaMesta **=** **true;**  **}**  **else**  **{**  imaMesta **=** **false;**  **}**  **return** imaMesta**;**  **}** |

*Listing 7. – Metoda zadužena za proveru poslovne logike iz klase clsPoslovnaPravila*

#### **Kod za osnovnu validaciju podataka sloja prezentacione logike**

|  |
| --- |
| **public** bool DaLiJeJedinstvenZapis**()**  **{**  bool JedinstvenZapis **=** **false;**  DataSet dsPodaci **=** **new** DataSet**();**  clsOperaterDB objOperaterDB **=** **new** clsOperaterDB**(**pStringKonekcije**);**  dsPodaci **=** objOperaterDB**.**DajOperateraPoJMBG**(**pJMBG**);**    **if** **(**dsPodaci**.**Tables**[**0**].**Rows**.**Count **==** 0**)**  **{**  JedinstvenZapis **=** **true;**  **}**  **else**  **{**  JedinstvenZapis **=** **false;**  **}**  **return** JedinstvenZapis**;**  **}** |

*Listing 8. – Metoda iz klase prezentacione logike clsFormaOperaterUnos zadužena za proveru jedinstvenosti zapisa pri unosu novog zapisa operatera*

#### **Korisnički interfejs**

|  |
| --- |
| **protected** void btnSnimi\_Click**(object** sender**,** EventArgs e**)**  **{**  // \*\*\*\*\*\*\*\*\*\*\*preuzimanje vrednosti sa korisnickog interfejsa  // 2. nacin - preuzimaju atributi klase prezentacione logike  objFormaOperaterUnos**.**JMBG **=** txbJMBG**.**Text**;**  objFormaOperaterUnos**.**Prezime **=** txbPrezime**.**Text**;**  objFormaOperaterUnos**.**Ime **=** txbIme**.**Text**;**  objFormaOperaterUnos**.**NazivLetelice **=** ddlLetelica**.**Text**;**    // \*\*\*\*\*\*\*\*\*\*\* provera ispravnosti vrednosti  // 1. provera popunjenosti  bool SvePopunjeno **=** objFormaOperaterUnos**.**DaLiJeSvePopunjeno**();**  // 2. provera ispravnosti - karakteri, vrednost iz domena, jedinstvenost zapisa  bool JedinstvenZapis **=** objFormaOperaterUnos**.**DaLiJeJedinstvenZapis**();**  // 3. provera ispravnosti - provera uskladjenosti podataka sa poslovnim pravilima  bool UskladjenoSaPoslovnimPravilima **=** objFormaOperaterUnos**.**DaLiSuPodaciUskladjeniSaPoslovnimPravilima**();**  // \*\*\*\*\*\*\*\*\*\* snimanje u bazu podataka  string porukaStatusaSnimanja **=** ""**;**  **if** **(**SvePopunjeno**)**  **{**  **if** **(**JedinstvenZapis**)**  **{**  **if** **(**UskladjenoSaPoslovnimPravilima**)**  **{**  // snimanje podataka  objFormaOperaterUnos**.**SnimiPodatke**();**  // priprema teksta poruke o uspehu snimanja  porukaStatusaSnimanja **=** "USPESNO SNIMLJENI PODACI!"**;**  **}**  **else**  **{**  porukaStatusaSnimanja **=** "PODACI NISU U SKLADU SA POSLOVNIM PRAVILIMA!"**;**  **}**  **}**  **else**  **{**  porukaStatusaSnimanja **=** "VEC POSTOJI OPERATER SA ISTIM JMBG!"**;**  **}**  **}**  **else**  **{**  // priprema teksta poruke o gresci  porukaStatusaSnimanja **=** "NISU SVI PODACI POPUNJENI!"**;**  txbJMBG**.**Focus**();**  **}**  // \*\*\*\*\*\*\*\*\*\* obavestavanje korisnika o statusu snimanja  lblStatus**.**Text **=** porukaStatusaSnimanja**;**  **}** |

*Listing 9. – Funkcionalni prikaz koda koji se izvršava prilikom klika na dugme snimi na formi za unos novog operatera*

### **Delovi koda i opis implementacije podrške personalizaciji aplikacije i bezbednosti**

|  |
| --- |
| **protected** void btnPRIJAVA\_Click**(object** sender**,** EventArgs e**)**  **{**  // provera korisnika  clsFormaLogin objFormaLogin **=** **new** clsFormaLogin**(**ConfigurationManager**.**ConnectionStrings**[**"NasaKonekcija"**].**ConnectionString**);**  objFormaLogin**.**KorisnickoIme **=** txbKorisnickoIme**.**Text**;**  objFormaLogin**.**Sifra **=** txbSifra**.**Text**;**  bool pronadjenKorisnik **=** objFormaLogin**.**VazeciKorisnik**();**  **if** **(**pronadjenKorisnik**)**  **{**  // TO DO  string ImePrezime **=** objFormaLogin**.**DajImePrezimeKorisnika**();**  // ubacivanje korisnika u sesiju  Session**.**Add**(**"KorisnikImePrezime"**,** ImePrezime**);**  // ucitavanje Welcome admin  Response**.**Redirect**(**"WelcomeAdmin.aspx"**);**  **}**  **else**  **{**  lblStatus**.**Text **=** "KORISNIK NIJE PRONADJEN!"**;**  **}**  **}** |

*Listing 10. – Prijava korisnika*

|  |
| --- |
| **protected** void gvSpisakZvanjaEdit\_SelectedIndexChanged**(object** sender**,** EventArgs e**)**  **{**  Response**.**Redirect**(**"LetelicaDetaljiEdit.aspx?RegBr=" **+** gvSpisakLetelicaEdit**.**Rows**[**gvSpisakLetelicaEdit**.**SelectedIndex**].**Cells**[**1**].**Text**);**  **}** |

*Listing 11. – Kod sa LetelicaTabelaEdit.aspx koji preusmerava korisnika na stranicu za detaljni prikaz izmene izabrane letelice*

### **Delovi koda i opis implementacije osnovnih principa OOP**

|  |
| --- |
| **public** DataSet DajLetelicuPoNazivu**(**string NazivLetelice**)**  **{**  // MOGU biti jos neke procedure, mogu SE VRATITI VREDNOSTI I U LISTU, DATA TABLE...  DataSet dsPodaci **=** **new** DataSet**();**  SqlConnection Veza **=** **new** SqlConnection**(**pStringKonekcije**);**  Veza**.**Open**();**  SqlCommand Komanda **=** **new** SqlCommand**(**"DajLetelicuPoNazivu"**,** Veza**);**  Komanda**.**CommandType **=** CommandType**.**StoredProcedure**;**  Komanda**.**Parameters**.**Add**(**"@NazivLetelice"**,** SqlDbType**.**NVarChar**).**Value **=** NazivLetelice**;**  SqlDataAdapter da **=** **new** SqlDataAdapter**();**  da**.**SelectCommand **=** Komanda**;**  da**.**Fill**(**dsPodaci**);**  Veza**.**Close**();**  Veza**.**Dispose**();**  **return** dsPodaci**;**  **}**  // overloading metoda - isto se zove, ima drugaciji parametar  **public** DataSet DajLetelicuPoNazivu**(**clsLetelica objLetelicaZaFilter**)**  **{**  // MOGU biti jos neke procedure, mogu SE VRATITI VREDNOSTI I U LISTU, DATA TABLE...  DataSet dsPodaci **=** **new** DataSet**();**  SqlConnection Veza **=** **new** SqlConnection**(**pStringKonekcije**);**  Veza**.**Open**();**  SqlCommand Komanda **=** **new** SqlCommand**(**"DajLetelicuPoNazivu"**,** Veza**);**  Komanda**.**CommandType **=** CommandType**.**StoredProcedure**;**  Komanda**.**Parameters**.**Add**(**"@NazivLetelice"**,** SqlDbType**.**NVarChar**).**Value **=** objLetelicaZaFilter**.**Naziv**;**  SqlDataAdapter da **=** **new** SqlDataAdapter**();**  da**.**SelectCommand **=** Komanda**;**  da**.**Fill**(**dsPodaci**);**  Veza**.**Close**();**  Veza**.**Dispose**();**  **return** dsPodaci**;**  **}** |

*Listing 12. – Method overloading u klasi podataka csLetelicaDB*

|  |
| --- |
| // Privatni atribut za konekcioni string - Enkapsulacija  **private** string pStringKonekcije**;** |

*Listing 13. – Enkapsulacija u klasi podataka csLetelicaDB*

### **Delovi koda i opis implementacije SOLID principa**

|  |
| --- |
| // Javna metoda za pribavljanje svih letelica - Single Responsibility Principle  **public** DataSet DajSveLetelice**()**  **{**  DataSet dsPodaci **=** **new** DataSet**();**  **using** **(**SqlConnection Veza **=** **new** SqlConnection**(**pStringKonekcije**))**  **{**  Veza**.**Open**();**  SqlCommand Komanda **=** **new** SqlCommand**(**"DajSveLetelice"**,** Veza**)**  **{**  CommandType **=** CommandType**.**StoredProcedure  **};**  SqlDataAdapter da **=** **new** SqlDataAdapter**(**Komanda**);**  da**.**Fill**(**dsPodaci**);**  **}**  **return** dsPodaci**;**  **}** |

*Listing 14. – Primer single responsibility principle*

|  |
| --- |
| // Konstruktor – Garantuje Dependency Injection  **public** clsLetelicaDB**(**string NoviStringKonekcije**)**  **{**  pStringKonekcije **=** NoviStringKonekcije**;**  **}** |

*Listing 15. – Primer Dependency Injection*

### **Delovi koda objašnjenja realizacije clean code pristupa**

|  |
| --- |
| //CRC karta za klasu clsLetelicaDB  //Class: clsLetelicaDB  //Responsibilities:  //Manage database connections: Održavanje i upravljanje stringom konekcije prema bazi podataka.  //Retrieve data:  //Dohvati sve letelice (DajSveLetelice).  //Dohvati naziv letelice prema registracionom broju (DajNazivPremaRegBr).  //Dohvati registracioni broj prema nazivu letelice (DajRegBrPoNazivu).  //Dohvati letelicu prema nazivu (DajLetelicuPoNazivu).  //Insert data:  //Dodaj novu letelicu (SnimiNovuLetelicu).  //Delete data:  //Obriši letelicu prema objektu (ObrisiLetelicu).  //Obriši letelicu prema registracionom broju (ObrisiLetelicu).  //Update data:  //Izmeni letelicu prema objektu (IzmeniLetelicu).  //Izmeni letelicu prema registracionom broju (IzmeniLetelicu).  //Collaborators:  //clsLetelica: Koristi se za prosleđivanje i dobijanje podataka o letelici.  //SqlConnection: Koristi se za otvaranje i zatvaranje konekcije sa bazom podataka.  //SqlCommand: Koristi se za izvršavanje SQL komandi.  //SqlDataAdapter: Koristi se za popunjavanje DataSet-a sa podacima iz baze. |

*Listing 16. – CRC karta za klasu clsLetelicaDB*

Klasa clsLetelicaDB (kao i sve ostale) demonstrira primenu Clean Code principa kroz jasan i deskriptivan naziv metoda, što poboljšava čitljivost i razumljivost koda. Svaka metoda ima specifičnu odgovornost, kao što su dodavanje, brisanje, ažuriranje i dohvatanje podataka iz baze, što prati Single Responsibility Principle (SRP) iz SOLID principa. Konekcija sa bazom podataka je enkapsulirana unutar metode, obezbeđujući da je resurs pravilno otvoren i zatvoren, čime se smanjuje mogućnost curenja resursa. Korišćenje parametara za SQL komande štiti od SQL injection napada, dok se preopterećivanje metoda koristi za pružanje fleksibilnosti u interakciji s podacima, sve ovo doprinosi održivom, sigurnom i lako razumljivom kodu.

# **LITERATURA**

[1] Liu, Yinling, et al. "The design and simulation of an autonomous system for aircraft maintenance scheduling." Computers & industrial engineering 137 (2019): 106041.

[2] Zakon o vazdušnom saobraćaju Republike Srbije <https://www.paragraf.rs/propisi/zakon-o-vazdusnom-saobracaju-republike-srbije.html> , datum pristupa: 14.06.2024

[3] ICAO vazduhoplovne regulative [Air Transport Policy and Regulation (icao.int)](https://www.icao.int/sustainability/pages/economic-policy.aspx) , datum pristupa: 14.06.2024

[4] EASA vazduhoplovne regulative [Regulations | EASA (europa.eu)](https://www.easa.europa.eu/en/regulations) , datum pristupa: 14.06.2024

[5] Airplane Aircraft Flight Logo <https://www.pngwing.com/en/free-png-xtfck> , datum pristupa: 20.06.2024