

## 1. Uvod

Uloga ovog dokumenta je osnovni vodič za vođenje projekta pod nazivom „Implementacija sustava za prijavu i upravljanje kvarovima infrastrukture“. Dokument sadrži procjenu projekta, upravljanje rizicima, raspored projekta, organizaciju sudionika projekta te mehanizme praćenja i kontrole.

### 1.1. Doseg projekta

Sustav za prijavu kvarova softversko je rješenje koje će omogućiti efikasno upravljanje prijavama kvarova. Sustav obuhvaća korisničko sučelje za prijavu kvarova, administrativni portal za upravljanje zadacima, te analitičke alate za praćenje performansi sustava i identifikaciju uzoraka kvarova.

Glavni ulazi u sustav uključuju:

- Prijava kvarova od strane korisnika putem korisničkog sučelja.
- Dodjela zadataka za uklanjanje kvarova od strane administratora ili odgovorne osobe.
- Otvaranje radnog naloga i praćenje postupka razrješavanja istog.
- Analiza podataka o pojedinim kvarovima u svrhu identificiranja uzoraka i potrebnih akcija.

Funkcioniranje obrade – obrada ulaznih podataka se odvija idućim koracima:

- Prijava kvara: Sustav prima ulazne podatke o prijavljenom kvaru kao što su opis kvara, lokacija, prioritet, područje i slične relevantne informacije.
- Dodjela zadataka: Temeljem prijavljenih kvarova sustav temeljem područja i lokacije dodjeljuje zadatke odgovornom timu ili osobi za otklanjanje kvara.
- Otvaranje i praćenje radnog naloga: Korisnici ili tehničko osoblje unose sve o postupcima rješavanja kvara te ih prate kroz sustav radi transparentnosti i praćenja napretka.
- Analiza podataka: Sustav će sve prikupljene podatke o kvarovima analizirati radi identifikacije uzroka, trendova i potencijalnih problema i omogućio preventivno djelovanje.

Opis izlaza – izlazi sustava uključuju:

- Potvrdu o uspješnoj prijavi kvara koja se šalje korisniku.
- Dodijeljeni zadatak pojedinom odgovornom timu ili osobi.
- Ažurirane informacije o statusu i napretku rješavanja samoga kvara.
- Analitički izvještaji o performansama sustava te identificiranim uzorcima kvarova.

### 1.2. Glavne funkcije programske opreme

Funkcionalna dekompozicija programske opreme pruža detaljan pregled svih ključnih funkcionalnosti koji će biti implementirani u sustav prijave kvarova. Dekompozicija pruža jasno razumijevanje pojedine funkcije te njene uloge u sustavu, a služiti će kao temelj procjene i daljnjeg razvoja sustava.

#### 1. Prijava kvara

- Korisnicima je omogućena prijava kvara kroz korisničko sučelje.

- Upisivanje podataka o samome kvaru koji uključuju opis problema, lokaciju, prioritet, područje i slične informacije
  - Sustav korisniku u obliku potvrde prijave kvara javlja uspješnost prijave te mu dodjeljuje jedinstveni identifikator kvara odnosno naloga.
2. Upravljanje zadatcima
- Odgovorna osoba ili administrator ima mogućnost dodjele zadataka odnosno naloga osobama ili timovima.
  - Zadatci se dodjeljuju ovisno o prioritetu, području, lokaciji i raspoloživim resursima.
  - Sustav prati napredak i status pojedinih zadataka tokom procesa njihovog razrješavanja.
3. Praćenje postupka rješavanja
- Tehničko osoblje unosi informacije o postupcima rješavanja kvarova u sustav, primjerice slike stanja radova, utrošeni materijal i slično.
  - Svaki postupak razrješavanja povezan s odgovarajućim kvarom i zadatkom.
  - Sustav će omogućiti pregled te analizu svih postupaka rješavanja kako bi se poboljšala učinkovitost samog procesa.
4. Analiza podataka o kvarovima
- Sustav prikuplja i analizira informacije o kvarovima koji su prijavljeni u svrhu identificiranja uzoraka, trendova i potencijalnih problema.
  - Analitički alati će pružiti mogućnost generiranja izvještaja o performansama sustava te identificiranim trendovima i uzorcima kvarova.
  - Temeljem analize sustav može predložiti mjere prevencije kako bi se spriječili budući kvarovi.

### 1.3. Zahtjevi za performansama i ponašanje programske opreme

#### Performanse sustava

- Opterećenje sustava: Sustav mora podnositi velik broj istovremenih zahtjeva korisnika radi osiguravanja neprekidnog funkcioniranja te minimalnog vremena čekanja, čak i u periodima visokog opterećenja.
- Brzina odaziva: Potrebno je da sustav osigura brz odgovor na korisničke zahtjeve za prijavu kvara, dodjelu pripadajućeg zadatka te praćenje postupka rješavanja istog. Ovaj zahtjev potrebno je ispuniti u svrhu osiguravanja efikasnog upravljanja kvarovima.

#### Pouzdanost sustava

- Stabilnost: sustav mora biti stabilan i otporan na prekide u radu u svrhu osiguravanja neprekidnog funkcioniranja, bez nepotrebnih zastoja ili prekida same usluge.
- Oporavak od pogrešaka: Ako dođe do grešaka ili neočekivanih problema, sustav treba imati ugrađene mehanizme oporavka koji osiguravaju što manji gubitak podataka te brz povratak u normalno stanje.

#### Skalabilnost sustava:

- **Fleksibilnost:** Da bi se sustav mogao prilagoditi izmjenama u zahtjevima korisnika i da bi bilo moguće povećati opseg rada bez gubitaka performansi on mora biti fleksibilan i skalabilan. Poželjno je da je sustav vertikalno i horizontalno skalabilan.

Sigurnost sustava:

- **Zaštita podataka:** Sustav mora osigurati visoku razinu zaštite podataka korisnika, informacija o prijavljenim kvarovima, posebno lokacije i slične osjetljive informacije.
- **Pravilno upravljanje pristupom:** Potrebni su mehanizmi za autentifikaciju i autorizaciju korisnika te strogu kontrolu pristupa da bi se osiguralo da samo ovlaštene osobe imaju pristup određenim funkcionalnostima i podacima.

### 1.4. Upravljanje i tehnička ograničenja

**Ograničenja resursa:** Prvo ograničenje jest financijsko, dakle projekt je ograničen definiranim budžetom što zahtijeva pažljivo upravljanje troškovima kako bi projekt bio ostvariv unutar proračuna. Drugo resursno ograničenje je vremensko. Potrebno je poštovati rok za završetak projekta kako bi se pravovremeno dostavio konačni proizvod.

**Ograničenja infrastrukture:** Sustav treba biti kompatibilan s postojećim sličnim programskim rješenjima radi osiguravanja glatke integracije i minimalnog prekida u radu. Treba imati na umu da sustav treba moći proširiti ili mijenjati prema potrebama organizacije.

**Tehnički pristup razvoju:** Bitno je primijeniti odgovarajuće tehnologije za razvoj sustava radi osiguravanja optimalnih performansi, sigurnosti i pouzdanosti. U svrhu brze prilagodbe promjenama i kontinuirane dostave vrijednosti kroz iterativni pristup odabrana je agilna metodologija razvoja.

**Sigurnost podataka:** Bitno je da se poštuju sve relevantne zakonske i regulativne zahtjeve o zaštiti podataka radi očuvanja osobnih podataka korisnika i osjetljivih podataka organizacije. Implementirati će se sustav za upravljanje identitetima i pristupom, radi osiguravanja da samo ovlaštene osobe imaju pristup određenim dijelovima sustava.

## 2. Procjena projekta

1. Cijena projekta:

Plaće zaposlenika:

- **Developeri:** Procjena 2 developera po punom radnom tijekom 3 mjeseci. Prosječna plaća developera iznosi 1750€ mjesečno. (Izvor: <https://www.netokracija.com/usporedba-placa-it-programeri-hrvatska-svijet-209637>)
- **Dizajner:** Jedan dizajner po punom radnom vremenu tijekom 1 mjeseca. Prosječna plaća dizajnera iznosi 1158€ mjesečno. (Izvor: <https://www.adorio.hr/placa/web-dizajner>)
- **Menadžer projekta:** Jedan menadžer projekta po punom radnom vremenu tijekom 3 mjeseci. Prosječna plaća menadžera iznosi 1495€ mjesečno. (Izvor: <https://www.adorio.hr/placa/it-manager>)

Računalna oprema:

- Nabava radnih stanica za developere, dizajnera i menadžera. Procijenjeni trošak za stol, računalo i stolac iznosi 8000€. Nije uključen uredski prostor jer će se projekt razvijati na daljinu.
- Kupnja poslužitelja za razvoj i testiranje sustava. Procijenjeni trošak iznosi 150€ mjesečno.

### Licenciranje softvera:

- Nabava licenci za razvojne alate i platforme potrebne za izgradnju sustava. Procijenjeni trošak iznosi 300€.
- Kupnja licenci za operacijski sustav i ostale potrebne softverske alate. Procijenjeni trošak iznosi 400€.

### Održavanje sustava:

- Procjena troškova održavanja sustava tijekom prvih 12 mjeseci nakon implementacije. Procijenjeni trošak iznosi održavanja 12000 €.

Ukupan procijenjeni budžet iznosi 36000-42000 €.

### 2. Procjena uloženog truda:

- Developeri: 2 developera po punom radnom vremenu tijekom 3 mjeseci.
- Dizajner: Jedan dizajner po punom radnom vremenu tijekom 1 mjeseca.
- Menadžer projekta: Jedan menadžer projekta po punom radnom vremenu tijekom 3 mjeseci.

### 3. Procjena vremena:

- Planirano trajanje projekta: 3 mjeseca.

## 2.1. Resursi projekta

Resursi potrebni za projekt uključuju ljudske resurse u vidu dva developera, jednog dizajnera i jednog menadžera. Od ostalih resursa potrebne su licence za korištenje odabranih programskih alata, a sklopovska oprema potrebna za izvođenje projekta su četiri računala. Programska oprema potrebna za razvoj su IntelliJ IDEA, Slack, Astah i Microsoft Project.

## 3. Upravljanje rizicima

Projekt razvoja sustava za prijavu kvarova nosi sa sobom specifične rizike koji utječu na njegovu izvedivost i konačan ishod. Stoga su ključni koraci identifikacija i analiza ovih rizika što će nam omogućiti njihovo kvalitetnije upravljanje.

### 3.1. Rizici projekta

Rizike ćemo podijeliti u četiri kategorije, a to su tehnički rizici, rizici vezani uz korisničke zahtjeve, sigurnosni rizici te rizici vezani uz upravljanje projektom. Tehnički rizici uključuju probleme s integracijom s postojećim sustavima, nekompatibilnost s različitim platformama, nedostatak podrške za neke operativne sustave te rizici vezani uz stabilnost i performanse sustava. U rizike vezane za korisničke zahtjeve ulaze mogućnost nezadovoljstva korisnika sa funkcionalnostima sustava te da karakteristike sustava nisu u skladu sa stvarnim potrebama korisnika. Također promjene u korisničkim zahtjevima tokom samog razvojnog procesa mogu zahtijevati znatne prilagodbe te izazvati kašnjenje. Što se tiče sigurnosnih rizika postoji mogućnost neovlaštenog pristupa podacima, krađe identiteta i sličnih napada. Stoga je za

prevenciju takvih incidenata ključno održavanje visokih sigurnosnih standarda. Četvrta skupna koja je vezana za upravljanje projektom uključuje nedostatak jasnih komunikacijskih kanala, nedovoljnu resursnu alokaciju te poteškoće u vođenju tima. Svi ovi rizici mogu usporiti razvoj te dovesti do kašnjenja i povećanja troška.

### 3.2. Upravljanje rizicima

U svrhu učinkovitog upravljanja rizicima aktivno će se sudjelovati u identifikaciji potencijalnih rizika uz tehniku brainstorminga. Za svaki identificirani rizik provest će se analiza istog koja uključuje procjenu vjerojatnosti njegovog nastanka, utjecaja na projekt te potencijalnih strategija za upravljanje istim. Za to će se koristiti matrica vjerojatnosti i utjecaja u svrhu rangiranja rizika prema značaju. Nakon analize planira se konkretan odgovor na rizike koji za cilj ima minimiziranje ili eliminaciju identificiranih rizika. Sastoji se od implementacije sigurnosnih protokola, redovitog ažuriranja softvera radi otklanjanja ranjivosti te osiguravanja jasnih procedura za komunikaciju te upravljanje promjenama. Idući korak jest implementacija planova odgovora koja se provodi u suradnji s relevantnim članovima tima. Potrebno je osigurati da su svi upoznati sa svojim ulogama i odgovornostima te da su planovi definirani jasno kako bi njihova primjena bila lakša. Monitoriranjem i kontrolom rizika redovito će se pratiti identificirani rizici tokom cijelog trajanja projekta, koristiti će se redovne revizije i evaluacije u svrhu osiguravanja učinkovitosti planova odgovora.

### 3.3. Tablica rizika

Ime rizika	Vjerojatnost	Utjecaj	RM3 referenca
Integracijski problemi	Visoka	Visok	RM3-01
Neusklađenost s različitim platformama	Niska	Nizak	RM3-02
Nedostatak podrške za operativne sustave	Niska	Nizak	RM3-03
Performanse i stabilnost sustava	Srednja	Visok	RM3-04
Nezadovoljstvo korisnika	Visoka	Visok	RM3-05
Neusklađenost s korisničkim potrebama	Srednja	Visok	RM3-06
Promjene u korisničkim zahtjevima	Visoka	Visok	RM3-07
Neovlašteni pristup podacima	Srednja	Visok	RM3-08
Krađa identiteta	Srednja	Visok	RM3-09
Ostali sigurnosni propusti	Srednja	Visok	RM3-10
Nedostatak jasnih komunikacijskih kanala	Srednja	Visok	RM3-11
Nedovoljna alokacija resursa	Visoka	Visok	RM3-12
Problemi s vođenjem tima	Srednja	Srednji	RM3-13

#### 4. Raspored projekta

Redni broj	Zadatak	Odgovorna osoba	Predviđeno trajanje	Početni datum	Završni datum
1	Analiza zahtjeva i projektiranje sustava	Filip Cindrić	2 tjedna	18.03.2024.	01.04.2024.
2	Izrada tehničke specifikacije	Bruno Đapić, Filip Cindrić	1 tjedan	25.03.2024.	01.04.2024.
3	Strukturirana analiza i dizajn te modeliranje podataka	Bruno Đapić, Filip Cindrić	2 tjedna	01.04.2024.	15.04.2024.
4	Razvoj korisničkog sučelja	Bruno Đapić, vanjski suradnik dizajner	2 tjedna	06.05.2024.	20.05.2024.
5	Implementacija backend sustava	Bruno Đapić	1 tjedan	20.05.2024.	27.05.2024.
6	Integracija i testiranje sustava	Filip Cindrić	1 tjedan	20.05.2024.	27.05.2024.
7	Automatizacija poslovnog procesa	Bruno Đapić, Filip Cindrić	2 tjedna	27.05.2024.	10.06.2024.
8	Puštanje u rad	Filip Cindrić	3 dana	10.06.2024.	13.06.2024.
9	Praćenje performansi	Bruno Đapić	12 mjeseci	13.06.2024.	13.06.2025.
10	Održavanje sustava	Bruno Đapić, Filip Cindrić	12 mjeseci	13.06.2024.	13.06.2025.

#### 5. Organizacija sudionika projekta

##### 5.1. Struktura tima

Redni broj	Ime člana	Uloga	Komunikacijski kanal
1	Bruno Đapić	Developer	Slack, Mail
2	Filip Cindrić	Developer	Slack, Mail
3	Branimir Ćurčić	Web dizajner(vanjski suradnik)	Slack, Mail
4	Krešimir Fertalj	Menadžer(voditelj) projekta	Sharepoint, Mail

## 5.2. Izvještavanje i komunikacija

Tijekom trajanja cijelog projekta kontinuirana komunikacija će se voditi kroz Slack. U skladu s rasporedom projekta nakon svake faze razvoja biti će održan sastanak putem Google Meet-a. Za pregled dokumentacije ostali članovi tima će evaluirati napisano te će isto biti korigirano kako se dogovori na sastanku. Što se pregleda koda aplikacije tiče koristit će se GIT tok za pregled koda kroz commit-ove i pull request-ove.

## 6. Mehanizmi praćenja i kontrole

Ovo poglavlje predstavlja tehnike koje će biti korištene za praćenje i kontrolu projekta. Bitno je naglasiti važnost osiguranja kvalitete i kontrole putem aktivnosti osiguranja kvalitete programske opreme(SQA) te upravljanje izmjenama i kontrolu aktivnosti upravljanja konfiguracije programske opreme(SCM).

### 6.1. Tehnike praćenja, osiguranje kvalitete i kontrola projekta

U svrhu praćenja i kontrole projekta biti će korištene iduće tehnike:

- Ganttov dijagram: On se koristi za vizualizaciju vremenskog rasporeda aktivnosti projekta te njihovih međusobnih ovisnosti.
- Mrežni dijagram aktivnosti(PDM): On omogućuje pregled aktivnosti s više detalja, njihove ovisnosti te kritičnih puteva u projektu.
- Earned Value Management(EVM): Metoda koja služi za mjerenje realne učinkovitosti te napretku projekta u odnosu na planirani napredak.
- Regulacijski i kontrolni alati: Korištenje Microsoft Project softwarea za praćenje napretka projekta, kontrolu troškova i resursa te praćenje izmjena.
- Definiranje standarda kvalitete: Utvrđivanje jasnih kriterija i standarda koji moraju biti zadovoljeni.
- Procesni pregledi: Redovni pregledi procesa razvoja u svrhu osiguravanja usklađenosti s postavljenim standardima i identificiranje eventualne slabosti.
- Testiranje kvalitete: Provjera performansi, funkcionalnosti i sigurnosti softverskog proizvoda za osiguranje visoke razine kvalitete.

### 6.2. Upravljanje izmjenama i kontrola

Verzioranje i kontrola promjena: praćenje verzija softverskih komponenti te kontrola promjena da bi se osigurala stabilnost i dosljednost uz pomoć GIT toka.

Identificiranje konfiguracijskih stavki: Utvrđivanje svih elemenata programskog sustava koji će biti pod kontrolom.

Upravljanje konfiguracijom: Korištenje alata za praćenje i upravljanje konfiguracije softverskih elemenata tokom cijelog životnog ciklusa.