

1. Uvod

Jednostavna aplikacija za vođenje podataka o objektima interaktivne karte hrvatskih autocesta. Aplikacija će se sastojati od namjenskog programa napisanog po načelu objektno orijentiranog modeliranja. Program će biti napisan u programskom jeziku C#, a kao okruženje ćemo koristiti Visual Studio 2022. Ideja je omogućiti pregled dosad svih unesenih podataka o objektima na autocestama.

1.1. Doseg projekta

Opis programske opreme, to jest opis glavnih ulaza, obrade te opis izlaza bez ulaženja u detalje ugradnje.

Aplikacija omogućuje unos, izmjenu i brisanje podataka o objektima na autocestama. Evidentiraju se podaci o autocestama, odmorištima, objektima autocesta (čvorovi, tuneli, mostovi, vijadukti), kamerama, ENC postajama, cjenovnicima te podaci o uslugama za gospodarstveni sektor i informacije o stanjima na cestama. Aplikacija nam daje pregled svih dosad unesenih podataka i objekta.

1.2. Glavne funkcije programske opreme

Funkcionalna dekompozicija programske opreme. Koristit će se tijekom procjene projekta i izrade vremenskog rasporeda.

Funkcije programske opreme su: unesiPodatke, izmjeniPodatke, pregledajPodatke, izbrišiPodatke. UnesiPodatke korisniku omogućuje da odabere neki od objekata i za njega unese multimedijske podatke i opis. IzmjeniPodatke pruža korisniku izmjenu trenutnih podataka nekog objekta, a izbrišiPodatke omogućuje brisanje podataka. PregledajPodatke daje pregled svih podataka zabilježenih o nekom objektu.

1.3. Zahtjevi za performansama i ponašanje programske opreme

Posebni zahtjevi za performansama i ponašanjem programske opreme. Nefunkcionalni zahtjevi kao što je vrijeme odziva, robusnost, podržanost formalnih standarda i drugo.

Aplikacija ima jednostavne funkcionalnosti i ne zahtijeva prevelike zahtjeve za performansama. Programska oprema mora pružiti jednostavno korisničko sučelje i jasno odvojene cjeline na autocestama. Cilj je da se korisnik lako snalazi na aplikaciji i da nije potrebno prethodno iskustvo da bi unio, pregledao ili izmijenio podatke.

1.4. Upravljanje i tehnička ograničenja

Posebna ograničenja koja imaju utjecaj na način provođenja projekta te tehnologiju razvoja.

Postoji vremensko ograničenje na ovaj semestar. Aplikacija mora biti funkcionalna na kraju semestra.

2. Procjena projekta

Procjena cijene, napora i vremena potrebnog za provedbu projekta.

Na projektu radi 5 osoba. Procijenjeno vrijeme izrade je jedan semestar. Nije planiran nikakav financijski trošak.

2.1. Povijesni podaci korišteni za procjenu

Opis podataka iz povijesti (eng. Historical data) koji su važni za procjenu.

Vodstvo kolegija je procijenilo da je semestar dovoljan period za izradu jednostavne CRUD aplikacije. Procjena je temeljena na dosadašnjim iskustvima sa studentskim projektima, a dio

procjene je i bila ograničenost kolegija na jedan semestar. U procjenu je uzeto u obzir i dosadašnje iskustvo studenata u ovakvim projektima.

2.2. Procjena primijenjenih tehnika i rezultata

Opis svake procijenjene tehnike i ukupna procjena.

Aktivnost	Trajanje	Čovjek /zadatku
Doseg projekta	30	2
Analiza (prikupljanje informacija, prijedlog izvedbe sustava)	40	5
Izrada (kodiranje funkcije sustava, izrada korisničkog sučelja)	170	5
Izrada baze podataka	60	2
Testiranje	20	3
Pisanje dokumentacije sustava	60	5
Uvođenje sustava	40	5
	420	

2.2.1. Procjena tehnike m

Tablice ili jednadžbe povezane s procjenom tehnike m. Odlomak 2.2.1 se ponavlja za svaku od m tehnika.

Aktivnost	Trajanje
Izrada formi korisničkog sučelja, provjera unesenih podataka	45
Funkcije za pohranu podataka	15
Funkcije pretrage podataka	15
Funkcije unosa podataka	15
Funkcije izmjene podataka	15
Razvojno testiranje	30
	135

2.3. Usklađivanje procjena

Procjena konačne cijene, truda i vremena utrošenog na projekt.

Procjena je da svaki student radi po unaprijed određenom broju ECTSa koje nosi kolegij. 5 ECTSa traži 140 rada, a projekt nosi 60% ocjene. Stoga je to minimum koji se očekuje od svakog člana.

2.4. Resursi projekta

Svi resursi potrebni za projekt (ljudski resursi, sklopovska oprema, programska oprema, licence, alati i ostali resursi potrebni za provedbu projekta).

U projektu sudjeluje 5 studenata jedne grupe. Svi zajedno pišu aplikaciju, razrađuju i analiziraju problem te pišu dokumentaciju. Dio članova će administrirati bazom podataka, a

dio će razvijati backend aplikacije. Za rad je potrebno da svaki član ima minimalno jedno računalo, na kojima treba imati instaliran osnovni paket za izradu projekta, a to su Visual Studio, SQL Server te MS Office.

3. Upravljanje rizicima

[Diskusija rizika projekta i pristup upravljanju rizicima.](#)

Osnovni rizik je da projekt neće biti završen u zadanom roku. Također je rizik da aplikacija neće ispuniti očekivanja ili da neke funkcionalnosti neće biti ispravne. Ne očekuje se preveliki rizik promjene zahtjeva jer je unaprijed najavljeno da ih neće biti.

3.1. Rizici projekta

[Opis svakog pojedinog rizika.](#)

Aplikacija mora imati tražene funkcionalnosti. Ako nema onda se to smatra kao neispunjeno očekivanje. Kašnjenje u izradi je samo po sebi problem i njegova pojava znatno utječe na prolaz kolegija. Loše komentiran kod ne predstavlja veliki problem za aplikaciju.

3.2. Tablica rizika

[Kompletna tablica rizika. Ime rizika, vjerojatnost, utjecaj i referenca na RM3 \(eng. Risk Mitigation, Monitoring, Management - RM3\).](#)

Rizik	Vjerojatnost	Uticaj
Proizvod nije u skladu s očekivanjima	30%	1
Nefunkcionalna aplikacija	25%	2
Kašnjenje s izradom	5%	2
Promjene u zahtjevima	1%	3
Loše komentiran kod	10%	4

Vrijednosti utjecaja

- 1 – katastrofalno
- 2 – kritično
- 3 – marginalno
- 4 - zanemarivo

3.3. Pregled ublažavanja, praćenja i upravljanja rizicima

[Pregled ublažavanja, praćenja i upravljanja rizika \(eng. Risk Mitigation, Monitoring, Management - RM3\). Upravljanje rizikom može biti opisano i u zasebnom dokumentu.](#)

Proizvod nije u skladu s očekivanjima

- Ublažavanje rizika
 - Rizik se pokušava ublažiti stalnim kontaktom s klijentom za kojeg radimo aplikaciju.
- Praćenje rizika
 - Kroz razvoj baze i aplikacije, održavaju se sastanci s klijentom i on daje feedback na napravljeno.
- Upravljanje rizikom
 - Ako projekt ne nudi klijentova očekivanja, treba ga prekinuti i ponovo analizirati klijentove zahtjeve.

Nefunkcionalna aplikacija

- Ublažavanje rizika
 - Rizik se pokušava riješiti temeljitim pregledom zahtjeva i razrade svih funkcionalnosti
- Praćenje rizika
 - Radi se popis traženih i realiziranih funkcionalnosti.
- Upravljanje rizikom
 - Ako se ne provedu sve funkcionalnosti, treba pogledati popis traženih i potražiti nerealizirane funkcionalnosti te ih dodati u projekt.

Kašnjenje s izradom

- Ublažavanje rizika
 - Rizik se ublažava dobrom analizom potrebnog vremena za izradu aplikacije. Važno je postaviti unaprijed rokove i pridržavati ih se.
- Praćenje rizika
 - Za svaku fazu projekta se radi raspored i nužno je da ga se pridržavamo.
- Upravljanje rizikom
 - Moguća se odgoda isporuke aplikacije, ako za njeno kašnjenje postoji valjan razlog. Ako projekt kasni, a u nekom skorom roku se ne vidi njegova realizacija, treba odustati od projekta.

Promjene u zahtjevima

- Ublažavanje rizika
 - Redovni sastanci s korisnikom i praćenje njegovih zahtjeva.
- Praćenje rizika
 - Korisnik ima pravo mijenjati zahtjeve u nekoj mjeri, ali treba razraditi plan kako će promjene zahtjeva utjecati na izradu projekta i kašnjenje.
- Upravljanje rizikom
 - Kad se dese promjene u zahtjevima, treba se sastati s korisnikom i ponovo proći njegove zahtjeve i obavijestiti ga kakve će te promjene imati utjecaj na projekt.

Loše komentiran kod

- Ublažavanje rizika
 - Loše komentiranje možemo izbjeći ako se svi pridržavamo određenih pravila koja odgovaraju standardu.
- Praćenje rizika
 - Provjeravati način komentiranja koda tijekom projekta.
- Upravljanje rizikom
 - Ako uočimo loše komentiran kod, treba ponovo proći kroz kod i popraviti nedostatke.

4. Raspored projekta

4.1. Skup projektnih zadataka

Aktivnosti procesa su pisanje zapisnika, planiranje projekta, specifikacija zahtjeva, izrada konceptualnog i fizičkog modela, izrada prvobitne verzije web-aplikacije prema prijašnje

definiranim zahtjevima, izrada konačne verzije web-aplikacije, izrada programske dokumentacije te finalno, izrada korisničke dokumentacije.

4.2. Funkcionalna dekompozicija

Pisanje zapisnika se sastoji od bilježenja zahtjeva korisnika na intervjuu te kasnije formiranje dokumenta na temelju zabilježenih zahtjeva.

Planiranje projekta se sastoji od aktivnosti dokumentiranja plana projekta.

Specifikacija zahtjeva se sastoji od evidencije poslovnih, korisničkih, funkcionalnih te nefunkcionalnih zahtjeva.

Izrada konceptualnog modela se sastoji isključivo od kreiranja konceptualnog modela.

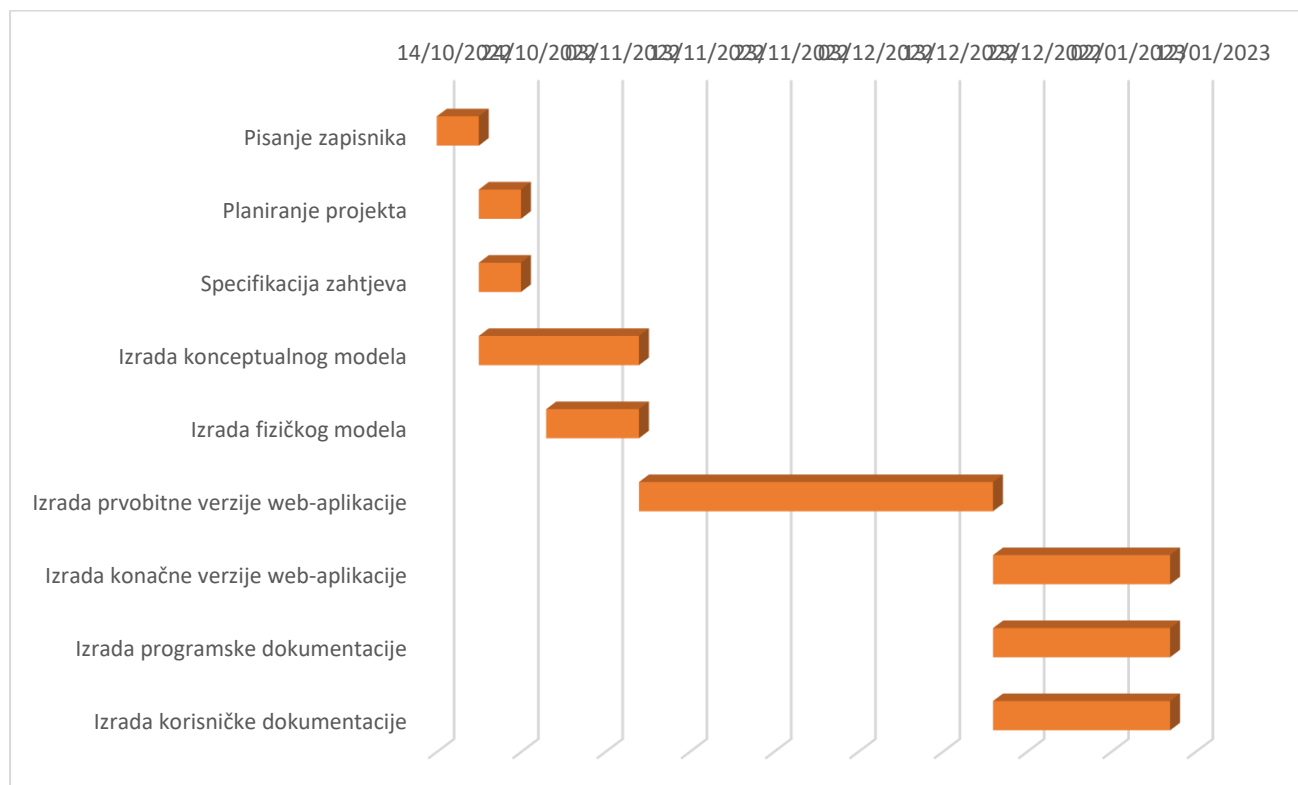
Izrada fizičkog modela se sastoji od izrade relacijskog modela podataka, implementacije relacijskog modela na zajednički SQL Server te punjenja tablica u bazi podataka podacima.

Izrada prvobitne verzije web-aplikacije se sastoji od izrade ORM modela i izrade zajedničke web aplikacije.

Izrada konačne aplikacije se sastoji od dovršavanja web-aplikacije.

Izrada programske i korisničke dokumentacije se sastoji isključivo od izrada dokumentacija.

4.3. Mrežni plan



4.4. Tablica vremenskih intervala projekta

Projektni zadaci:

Redni broj	Zadatak	Početak	Kraj
1.	Pisanje zapisnika	14.10.2022.	21.10.2022.
2.	Planiranje projekta	19.10.2022.	24.10.2022.
3.	Specifikacija zahtjeva	19.10.2022.	24.10.2022.
4.	Izrada konceptualnog modela	19.10.2022.	7.11.2022.
5.	Izrada fizičkog modela	27.10.2022.	7.11.2022.
6.	Izrada prvobitne verzije web-aplikacije	7.11.2022.	19.12.2022.
7.	Izrada konačne verzije web-aplikacije	19.12.2022.	9.1.2023.
8.	Izrada programske dokumentacije	19.12.2022.	9.1.2023.
9.	Izrada korisničke dokumentacije	19.12.2022.	9.1.2023.

5. Organizacija sudionika projekta

Na projektu sudjeluje 5 studenata računarstva te svi sudjeluju kao programeri.

5.1. Struktura tima

Programeri su zaduženi za izradu korisničkog sučelja, izvještaja, kodiranje funkcija, testiranje i pisanje dokumentacije i za izradu baze podataka.

Svi studenti u grupi (Luka Slugečić, Karlo Žižić, Ivan Žgela, Josip Kovač, Filip Šiktar) su zaduženi za navedeno: programsku implementaciju te izradu baze podataka.

Isto tako, svi sudionici su zaduženi za analizu zahtjeva te pisanje plana projekta.

5.2. Izvještavanje i komunikacija

Na projektu radi samo jedan tim, stoga komunikacija između timova nije potrebna.

Komunikacija unutar tima je verbalna i pismena putem Whatsapp grupe.

6. Mehanizmi praćenja i kontrole

S obzirom da se radi o projektu u sklopu kolegija Razvoj primjenjene programske potpore, napredak će se pratiti na tjednoj bazi, a nastavnici će ga pratiti na datume predodređenih rokova. Tim će se držati vremenskih rokova, jer će inače biti bodovno kažnjen.

6.1. Osiguranje kvalitete

- Namjena

Ovim planom je pokrivena izrada aplikacije interaktivne karte pomoću koje će se korisnici moći informirati o stanju na cestama. Plan sadrži faze pisanja zapisnika, planiranja, specifikacije zahtjeva, osmišljavanja fizičkog i konceptualnog modela, izrade web-aplikacije te pisanja dokumentacije.

- **Zaduženja**

Programeri i administratori baze podataka zaduženi su za izradu programskog sustava, testiranje i pisanje dokumentacije. Svi sudionici su zaduženi za analizu zahtjeva i pisanje plana projekta.

- **Dokumentacija**

Ovaj projekt izrađuje se na temelju Specifikacije zahtjeva.

Korisnička dokumentacija će sadržavati upute za korištenje web-aplikacije.

Programska dokumentacija će sadržavati generiranu dokumentaciju.

- **Alati, tehnike i metodologije**

Programska potpora će biti napisana objektno-orijentiranim pristupom u jeziku C#. Visual Studio Code će se koristiti kao radno okruženje. Za upravljanje bazom podataka će se koristiti SQL Server.

- **Kontrola programskog koda**

Za pohranu i kontrolu pristupa kodu koristit će se Visual Studio Source Control.

6.2. Upravljanje promjenama

Upravljanje izmjenama će izvoditi svi članovi tima.