

Obrazac za zadaću na predmetu "Uzorci dizajna" ak. god. 2024./2025.

Ime i prezime studentice: Ana Škarica

Matični broj: 0016147364

### Dio A. Osnovni podaci o zadaći

R.br.	Pitanje	Odgovor	
1.	Grupa na seminaru:	G2	
2.	Broj i naziv zadaće:	2	Zadaća 2 – Željeznički promet s voznim redom
3.	Procjena vremena za realizaciju bez decimala):	60 sati	
4.	Procjena % završenosti (bez decimala):	90 / 100%	
5.	Procjena bodova za izradu zadaće ( 1 decimala):	10.5 / (DZ2 - 13)	
6.	Želim prezentirati zadaću:	NE (već prezentirala prvu)	
7.	Koji dijelovi iz opisa zadaće nisu realizirani:	IVI2S nije implementirana u cijelosti na ispravan način	
8.	Postoji li dio zadaće koji vrijedi posebno istaknuti i zašto:	-	
9.	Postoje li dijelovi zadaće koji imaju pogrešku u radu i koje:	<p>Možda se ne broje sve greške kao stvarne greške (jer nije jasno bilo definirano u opisu koje odstupanje se smatra neispravnim, a koje je samo kašnjenje vlaka i dozvoljivo je).</p> <p>IVI2S ima dinamičke stupce, provjeru argumenata, ispisuje stanice i vlakove (između zadanog vremena) ali podaci tablice (izračuni vremena i udaljenosti) su ponekad neispravni.</p>	
10.	Da li ste koristili tuđi programski kod u realizaciji zadaće izvan spomenutih izvora na nastavi:	-	
11.	Da li ste koristili programska rješenja ili dijelove programskog koda od drugih kolega:	-	

### Dio B.1. Dokumentacija rješenja 1. zadatke (kopirano i nepromijenjeno)

Naziv uzorka dizajna	Klase koje sudjeluju u uzorku dizajna	Opis razloga odabira uzorka dizajna
Builder	VoziloDirector	Pošto postoje različite vrste željezničkih vozila (npr. lokomotiva, putnički vagon, teretni vagon za automobile, teretni kontejnerski vagon itd.) ima smisla koristiti ovaj uzorak pri kreiranju objekata vozila. Prema datoteci, iako su svi podaci vozila obavezni, sva vozila određene kategorije imaju neke podatke (stupce) ispunjenje s vrijednošću 0. Samim time taj podatak za tu grupu vozila nije bitan. Iz tog je razloga moguće imati više različitih procesa konstrukcije za vozila, od kojih će svaki pružati različitu reprezentaciju.
	VoziloBuilder	
	VoziloConcreteBuilder	
	Vozilo	
Singleton	ZeljeznickiSustav	Jer treba postojati točno 1 instanca klase sustava te mora postojati globalni pristup do nje (iz poznate točke pristupa) jer sadrži podatke koje cijeli sustav koristi dalje u radu (npr. liste učitanih podataka, ukupan broj grešaka u sustavu)
Factory Method	CsvCitacCreator	Jer se ovisno o datoteci koja je pružena kroz argumente, pored zastavice, drugačije čitaju podaci (i kreiraju objekti temeljem vrste objekata datoteke), a ako se naknadno doda još neki novi čitač, taj je proces olakšan (jer svaki CSV dokument ima drugačije podatke i proces njihova čitanja)
	CsvCitacVozilaCreator	
	CsvCitacKompozicijaCreator	
	CsvCitacStanicaCreator	
	CsvCitacProduct	
	CsvCitacVozilaProduct	
	CsvCitacKompozicijaProduct	
	CsvCitacStanicaProduct	

## Dio B.2. Dokumentacija rješenja 2. zadatka

Naziv uzorka dizajna	Klase koje sudjeluju u uzorku dizajna i u kojim ulogama	Status <sup>1</sup>	Opis razloga odabira uzorka dizajna
Singleton	ZeljeznickiSustav	P	Isti razlog kao u prvoj zadatci (dio B.1). Nadodano je još nekoliko lista, metode za njih i upravljanje novim komandama.
Builder	VoziloDirector VoziloBuilder VoziloConcreteBuilder Vozilo	S	Isti razlog kao u prvoj zadatci (dio B.1). Ništa nije mijenjano ni dodano.
Factory Method	CsvCitacCreator CsvCitacVozilaCreator CsvCitacKompozicijaCreator CsvCitacStanicaCreator CsvCitacProduct CsvCitacVozilaProduct CsvCitacKompozicijaProduct CsvCitacStanicaProduct CsvCitacOznakaDanaCreator CsvCitacOznakaDanaProduct CsvCitacVoznogRedaCreator CsvCitacVoznogRedaProduct	P	Isti razlog kao u prvoj zadatci (dio B.1). Bilo je potrebno dodati nove klase za nove datoteke – za CSV oznaka dana i CSV voznog reda – zato su dodane po 2 klase za svaku od datoteka – concrete creator i concrete product.
Composite	VozniRedComponent EtapLeaf VozniRedBaseComposite VlakComposite VozniRedComposite	N	Bio je zadan u opisu zadatka kao obavezni uzorak. Vozni red se sastoji od više vlakova, vlakovi imaju jednu ili više etapa sa stanicama što se može implementirati kao stablasta struktura – odnosno definira se hijerarhija klasa.
Visitor	VozniRedVisitorElement VozniRedVisitor IspisVlakovaVisitor IspisEtapVisitor IspisSimulacijeVisitor IspisVlakovaPoDanimaVisitor IspisVoznogRedaVisitor IspisVozRedaDoStaniceVisitor	N	Naslanja se na prethodno opisani uzorak Composite kako same klase koje u njemu sudjeluju ne bi bile prenatrpane. Korišten je jer dopušta definiranje nove operacije bez promjene klasa za elemente na kojima djeluje, što je potrebno u slučaju stablaste strukture s više traženih ponašanja (posebna struktura ispisa i drukčija logika u sastavljanju tog ispisa za svaku komandu). Na taj način je potrebno samo dodati novu concrete visitor klasu ukoliko se pojavi nova komanda u sljedećim zadacima.
Observer	ZeljeznickiObserverSubject VozniRedComponent ZeljeznickiObserver KorisnikConcreteObserver	N	Jer se u dijelu s korisnicima i simulacijom traži da se pretplaćene korisnike obavještava o stanici/vlaku koji ih prate. Na taj način kad jedan objekt mijenja stanje, svi koji ovise o njemu su obaviješteni i ažuriraju se (odnosno saznaju stanje vlaka/stanice koje se ispisuje na ekranu).
Mediator	PosrednikMediator UredIzgubljenoNadenoMediator KorisnikColleague Korisnik	N	Bio je zadan u opisu zadatka i da se mora koristiti za novu funkcionalnost. Pošto korisnici (putnici) gube stvari, umjesto da korisnik koji je pronašao ili izgubio nešto ispituje svakog korisnika je li izgubio/pronašao taj predmet uveden je posrednik koji će se baviti s time i preko kojeg će korisnici moći komunicirati. Detaljniji opis je u nastavku.

<sup>1</sup> N – dodan u 2. zadatku, P – promijenjen u 2. zadatku, S – bez promjena u 2. zadatku

## **Dio C.1. Opis promjena u odnosu na prethodnu zadaću**

Popravljen slučaj da program nije dobro prepoznao kad datoteka ne postoji – sad prepoznaje nepostojeću datoteku.

Čitač CSV datoteka sad čita još 2 dodatne datoteke (vozni red i oznake dana) i čitaju se opcionalni podaci iz nadopunjene datoteke o stanicama. U Singleton ZeljeznickiSustav dodane nove liste, komande i provjere tih komandi, kao i funkcionalnosti koje se na temelju njih događaju/pozivaju. Dodana pomoćna klasa s metodama za izračun međustanica (koristi se za IVI2S). Dodana klasa Pruge koja pomaže u nekim slučajevima te ažurirane neke stare metode da koriste tu novu klasu. Dodani novi argumenti i zastavice za pokretanje programa.

Implementiran Composite uzorak prema opisu u zadaći (vozni red -> vlak -> etapa sa stanicama), implementiran Mediator (opis u nastavku) s 2 nove komande i tematikom ureda za izgubljene i nađene stvari (i dodana potrebna klasa IzgubljeniPredmet), implementiran Observer za javljanje informacija pretplaćenim korisnicima prilikom simulacije, implementiran Visitor za razne vrste ispisa zbog puno komandi koji se naslanja na Composite strukturu (nadovezuje se).

## Dio C.2. Opis funkcionalnosti za uzorak dizajna Mediator

Mediator je korišten za funkcionalnost ureda izgubljeno/nađeno.

Kao što postoje uredi za izgubljene i nađene stvari na raznim mjestima, tako može biti i u željezničkome svijetu. Konkretno, korisnik može javiti da je izgubio neki predmet, ali i da je našao tuđi. Kako se ti korisnici (vjerojatno) međusobno ne znaju (jer mnogo ljudi svakodnevno putuje istim vlakom i prolazi stanicama) moguće je da postoji ured koji prati sve izgubljene i nađene predmete te prati koji korisnik je što našao, a koji što izgubio.

U mojoj konkretnoj implementaciji potrebno je da se korisnik prvo registrira (postojećom komandom DK ime prezime). Nakon toga se mogu unijeti 2 nove komande – NAD ime prezime predmet opis i IZG ime prezime predmet opis, npr.: NAD Marko Maric mobitel Samsung ili IZG Ana Anic mobitel Samsung. NAD komanda bi značila da je korisnik Marko Markic našao predmet tipa mobitel, a opisan je kao Samsung (opis se unosi kao 1 riječ, može se odvojiti znakom \_), dok bi IZG komanda značila da je Ana Anic izgubila predmet tipa mobitel i opisa Samsung.

Ukoliko je prvo korisnik javio da je našao predmet komandom NAD, a zatim drugi korisnik javio da je izgubio predmet IZG (isti predmeta i opis), korisnik koji je javio da je izgubio predmet preko ureda za nađene i izgubljene stvari saznaje da ga je prvi korisnik već našao. Ukoliko je prvo korisnik javio da je izgubio predmet komandom IZG, a zatim je drugi korisnik javio da je našao taj predmet komandom NAD saznaje se da je već prvi korisnik javio da je izgubio taj predmet. Naravno, ova bi se implementacija dala još detaljnije nadograditi (npr. da se Observer uzorkom još korisnik koji je javio da mu nedostaje predmet obavijesti da je njegov izgubljeni predmet nađen, no pošto je naglasak na implementaciji Mediator uzorka, ostavljena je samo ključna funkcionalnost pa se može pretpostaviti da je korisnik pri prijavi izgubljenog predmeta ostavio i kontakt informacije).

Ako se gleda sama implementacija uzorka i sudionici, Mediator sučelje jest klasa PosrednikMediator koja definira sučelje za komunikaciju s Colleague objektima, odnosno 2 metode. ConcreteMediator jest UredIzgubljenoNadenoMediator koji implementira ponašanje i logiku tih dvaju metoda. Colleague klasa jest sučelje KorisnikColleague, a ConcreteColleague je klasa Korisnik te svaka Colleague klasa poznaje svoj Mediator objekt.

## Dio D. Dijagram klasa s naglašavanjem klasa koje sudjeluju u pojedinom uzorku dizajna

