

Obrazac za zadaću na predmetu "Uzorci dizajna" ak. god. 2025./2026.

Ime i prezime studenta/ice: Mihael Baranašić

Matični broj: 0016152670

Dio A. Osnovni podaci o zadaći

R.br.	Pitanje	Odgovor	
1.	Grupa na seminaru:	G2	
2.	Broj i naziv zadaće:	3	Turistička agencija
3.	Procjena vremena za realizaciju bez decimala):	__35__ sati	
4.	Procjena % završenosti (bez decimala):	__90%__ / 100%	
5.	Procjena bodova za izradu zadaće (1 decimala):	__11__ / (DZ3 - 15)	
6.	Žalim prezentirati zadaću:	NEMA PREZENTACIJE ZADAĆE!	
7.	Koji dijelovi iz opisa zadaće nisu realizirani:	Svi dijelovi su realizirani	
8.	Postoji li dio zadaće koji vrijedi posebno istaknuti i zašto:	Ne postoji	
9.	Postoje li dijelovi zadaće koji imaju pogrešku u radu i koje:	Po mom testiranju nisam pronašao greške	
10.	Da li ste koristili tuđi programski kod u realizaciji zadaće izvan spomenutih izvora na nastavi:	Nisam	
11.	Da li ste koristili programska rješenja ili dijelove programskog koda od drugih kolega:	Nisam	
12.	Da li ste koristili alat/e generativne umjetne inteligencije u izradi zadaće, ako jeste koji/e ste koristili i za koje potrebe ste ga/ih koristili:	Koristio sam Claude AI za debuggiranje, pomoć kod nekih implementacija i konzultaciju za određene uzorke dizajna.	

Dio B.1. Dokumentacija rješenja 2. zadaće (kopirano i nepromijenjeno)

Naziv uzorka dizajna	Klase koje sudjeluju u uzorku dizajna i u kojim ulogama	Status ¹	Opis razloga odabira uzorka dizajna
Singleton	TuristicaAgencija	S	Osigurava da postoji samo jedna instance sustava turističke agencije koja upravlja svim aranžmanima, rezervacijama i njihovim stanjima. Thread-safe implementacija s double-checked locking mehanizmom sprječava istovremeno kreiranje više instanci.
Builder	Aranzman, Builder, AranzmanBuilder, AranzmanDirector	P	Aranzman ima 16 atributa od kojih je 7 obaveznih, a 9 optionalnih. Builder omogućava postupno kreiranje objekta s različitim kombinacijama atributa.
Factory method	Komanda, KomandaCreator ItakKomanda, ItapKomanda, IrtakKomanda, IroKomanda, OrtaKomanda, DrtaKomanda, IeiKomanda, ItasKomanda, OtaKomanda, UpKomanda, BpKomanda, IpKomanda, ItakKomandaCreator, ItapKomandaCreator, IrtakKomandaCreator, IroKomandaCreator, OrtaKomandaCreator, DrtaKomandaCreator, IeiKomandaCreator, ItasKomandaCreator, OtaKomandaCreator, UpKomandaCreator, BpKomandaCreator, IpKomandaCreator, KomandaFactory	N	Omogućuje kreiranje različitih komandi bez poznavanja konkretnih klasa. Svaka komanda ima svoj Creator koji enkapsulira logiku instanciranja i validacije argumenata. Factory klasa mapira tekstualne naredbe na odgovarajuće Creatore.
State	Rezervacija, RezervacijaStanje, NovaRezervacija, PrimljenaRezervacija, AktivnaRezervacija, NaCekanjuRezervacija, OdgodenaRezervacija, OtkazanaRezervacija	N	Rezervacija mijenja ponašanje ovisno o svom stanju. 6 stanja (NOVA, PRIMLJENA, AKTIVNA, NA_ČEKANJU, ODGOĐENA, OTKAZANA) imaju različita pravila prijelaza. State pattern eliminira složene if-else konstrukcije i čini logiku prijelaza jasnom i proširivom.
State	TuristicaAgencija, AranzmanStanje, UPripremiAranzman, AktivniAranzman, PopunjjenAranzman, OtkazanAranzman	N	Stanje aranžmana ovisi o broju aktivnih rezervacija i minimalnom/maksimalnom broju putnika. 4 stanja jasno definiraju životni ciklus aranžmana i omogućuju automatsko ažuriranje statusa pri promjeni broja rezervacija.
Facade	PodaciFacade CsvCitacAranzmana, CsvCitacRezervacija, AranzmanDTO, RezervacijaDTO,	N	Pojednostavljuje učitavanje podataka iz CSV datoteka. Facade enkapsulira složenost parsiranja, validacije i obrade grešaka. Klijentski kod koristi jednostavno sučelje bez poznavanja detalja implementacije čitača.
Composite	TuristicaKomponenta, AranzmanKomponenta, RezervacijaKomponenta	N	Omogućuje prikaz hijerarhijske strukture aranžman -> rezervacije. Composite pattern omogućuje uniforman pristup jednostavnim i

¹ N – dodan u 2. zadaći, P – promijenjen u 2. zadaći, S – bez promjena u 2. zadaći

			složenim elementima. Koristi se u ITAC komandi za prikaz aranžmana zajedno sa svim rezervacijama.
Decorator	IzvjestajKomponenta, ObicniIzvjestaj, IzvjestajDekorator, StatistikaDekorator, AnalizaDekorator, FinancijskiDekorator	N	Omogućuje dinamičko dodavanje funkcionalnosti izvještaju o aranžmanu. IEI komanda prima opcije (S-statistika, A-analiza, F-financije) i dinamički "omata" osnovni izvještaj s traženim dekoratorima. Moguće je kombinirati više dekoratora.
Visitor	TuristicaVisitor, StatistikaVisitor, TuristicaKomponenta, AranzmanKomponenta, RezervacijaKomponenta	N	Omogućuje skupljanje statistike o aranžmanima i rezervacijama prolazom kroz Composite strukturu. Visitor obrazac odvaja algoritam obrade od strukture podataka, što omogućuje dodavanje novih operacija bez mijenjanja postojećih klasa.

Dio B.2. Dokumentacija rješenja 3. zadaće

Naziv uzorka dizajna	Klase koje sudjeluju u uzorku dizajna i u kojim ulogama	Status ²	Opis razloga odabira uzorka dizajna
Singleton	TuristicaAgencija	S	Osigurava da postoji samo jedna instanca sustava turističke agencije koja upravlja svim aranžmanima, rezervacijama i njihovim stanjima. Thread-safe implementacija s double-checked locking mehanizmom sprječava istovremeno kreiranje više instanci.
Builder	Aranzman, Builder, AranzmanBuilder, AranzmanDirector	S	Aranzman ima 16 atributa od kojih je 7 obaveznih, a 9 opcionalnih. Builder omogućava postupno kreiranje objekta s različitim kombinacijama atributa.
Factory method	Komanda, KomandaCreator ItakKomanda, ItapKomanda, IrtakKomanda, IroKomanda, OrtaKomanda, DrtaKomanda, IeiKomanda, ItasKomanda, OtaKomanda, UpKomanda, BpKomanda, IpKomanda, PptarKomanda, PstarKomanda, PtarKomanda, UptarKomanda, VstarKomanda, ItakKomandaCreator, ItapKomandaCreator, IrtakKomandaCreator, IroKomandaCreator, OrtaKomandaCreator, DrtaKomandaCreator, IeiKomandaCreator, ItasKomandaCreator, OtaKomandaCreator, UpKomandaCreator, BpKomandaCreator, IpKomandaCreator, PptarKomandaCreator, PstarKomandaCreator, PtarKomandaCreator, UptarKomandaCreator, VstarKomandaCreator, KomandaFactory	P	Omogućuje kreiranje različitih komandi bez poznavanja konkretnih klasa. Svaka komanda ima svoj Creator koji enkapsulira logiku instanciranja i validacije argumenata. Factory klasa mapira tekstualne naredbe na odgovarajuće Creatore.
State	Rezervacija, RezervacijaStanje, NovaRezervacija, PrimljenaRezervacija, AktivnaRezervacija, NaCekanjuRezervacija, OdgodenaRezervacija, OtkazanaRezervacija	S	Rezervacija mijenja ponašanje ovisno o svom stanju. 6 stanja (NOVA, PRIMLJENA, AKTIVNA, NA_ČEKANJU, ODGODENA, OTKAZANA) imaju različita pravila prijelaza. State pattern eliminira složene if-else konstrukcije i čini logiku prijelaza jasnom i proširivom.
State	TuristicaAgencija, AranzmanStanje, UPripremiAranzman, AktivniAranzman, PopunjjenAranzman, OtkazanAranzman	S	Stanje aranžmana ovisi o broju aktivnih rezervacija i minimalnom/maksimalnom broju putnika. 4 stanja jasno definiraju životni ciklus aranžmana i omogućuju automatsko ažuriranje statusa pri promjeni broja rezervacija.
Facade	PodaciFacade CsvCitacAranzmana,	S	Pojednostavljuje učitavanje podataka iz CSV datoteka. Facade enkapsulira složenost

² N – dodan u 2. zadaći, P – promijenjen u 2. zadaći, S – bez promjena u 2. zadaći

	CsvCitacRezervacija, AranzmanDTO, RezervacijaDTO,		parsiranja, validacije i obrade grešaka. Klijentski kod koristi jednostavno sučelje bez poznavanja detalja implementacije čitača.
Composite	TuristickaKomponenta, AranzmanKomponenta, RezervacijaKomponenta	S	Omogućuje prikaz hijerarhijske strukture aranžman -> rezervacije. Composite pattern omogućuje uniforman pristup jednostavnim i složenim elementima. Koristi se u ITAC komandi za prikaz aranžmana zajedno sa svim rezervacijama.
Decorator	IzvjestajKomponenta, ObicniIzvjestaj, IzvjestajDekorator, StatistikaDekorator, AnalizaDekorator, FinancijskiDekorator	S	Omogućuje dinamičko dodavanje funkcionalnosti izvještaju o aranžmanu. IEI komanda prima opcije (S-statistika, A-analiza, F-financije) i dinamički "omata" osnovni izvještaj s traženim dekoratorima. Moguće je kombinirati više dekoratora.
Visitor	TuristickaVisitor, StatistikaVisitor, PretrazivanjeVisitor, TuristickaKomponenta, AranzmanKomponenta, RezervacijaKomponenta	P	Omogućuje skupljanje statistike o aranžmanima i rezervacijama prolazom kroz Composite strukturu. Visitor obrazac odvaja algoritam obrade od strukture podataka, što omogućuje dodavanje novih operacija bez mijenjanja postojećih klasa.
Strategy	RezervacijaStrategija JednaRezervacijaStrategija ViseRezervacijaStrategija NullRezervacijaStrategija	N	Upravljanje dozvoljenim brojem rezervacija prema zadaći. Strategy omogućava promjenu ponašanja u runtime-u ovisno o argumentu komandne linije. Null Object pattern eliminira provjere za null i osigurava siguran rad kada nijedna opcija nije zadana.
Observer	TuristickaPromatrac OsobaPromatrac PromatracSubjekt AranzmanSubjekt	N	Implementira preplatu na obavijesti o promjenama aranžmana i rezervacija (PTAR i UPTAR komande). Observer omogućava loose coupling između aranžmana i zainteresiranih osoba.
Memento	AranzmanMemento MementoCuvar AranzmanKomponenta	N	Spremanje i vraćanje stanja aranžmana i njegovih rezervacija (PSTAR i VSTAR komande). Memento enkapsulira interno stanje bez narušavanja enkapsulacije. Caretaker upravlja stack-om memento objekata.
Chain of Responsibility	ArgumentHandler ApstraktniArgumentHandler TaArgumentHandler, RtaArgumentHandler, JdrArgumentHandler, VdrArgumentHandler KonfiguracijaAplikacije	N	Parsiranje argumenata komandne linije (--ta, --rta, --jdr, --vdr). Svaki handler obrađuje svoj argument i proslijeđuje request sljedećem handleru u lancu. Omogućava lako dodavanje novih argumenata bez izmjene postojećeg koda.

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

Dodani su novi uzorci dizajna Strategy, Chain of Responsibility, Memento, Observer i dodatni visitor uz postojeći.

Implementirane su nove komande koristeći gore navedene uzorke.

Strategy uzorak implementiran je za upravljanje dozvoljenim brojem rezervacija.

Observer uzorak omogućava pretplatu na obavijesti o promjenama.

Memento uzorak koristi se za spremanje i vraćanje stanja aranžmana. Komande PSTAR i VSTAR pohranjuju i vraćaju stanje.

Chain of Responsibility implementiran je za parsiranje argumenata komandne linije. Četiri handler-a (TaArgumentHandler, RtaArgumentHandler, JdrArgumentHandler, VdrArgumentHandler) formiraju lanac.

Dio C.2. Opis funkcionalnosti za uzorak dizajna Decorator (kopirano i nepromijenjeno iz 2. zadaće)

Decorator uzorak implementiran je u IEI komandi koja generira proširene izvještaje o aranžmanima. Komanda prima oznaku aranžmana i opcische parametre S, A i F koji određuju koje sekcije izvještaja će biti uključene.

BasicniIzvjestaj generira osnovne informacije: naziv, period, cijenu i popis rezervacija. IzvjestajDekorator je apstraktna klasa koja omata IzvjestajKomponenta objekt i omogućuje dodavanje dodatnog sadržaja.

StatistikaDekorator dodaje postotak popunjenošti i stopu otkazivanja. AnalizaDekorator dodaje prvu i zadnju rezervaciju te prosječno vrijeme prije polaska. FinancijskiDekorator dodaje ostvareni prihod, potencijalni prihod i maksimalni prihod.

Korisnik kombinira dekoratore prema potrebama. Naredba "IEI 1 S" generira izvještaj sa statistikom, "IEI 1 SAF" generira kompletan izvještaj sa svim sekcijama. Redoslijed slova nije bitan jer se dekoratori primjenjuju dinamički. Novi dekoratori mogu se dodati bez izmjene postojećih klasa što je u skladu s Open-Closed principom

Dio C.3. Opis funkcionalnosti za uzorak dizajna Chain-of-Responsibility

Chain of Responsibility uzorak implementiran je za parsiranje argumenata komandne linije prilikom pokretanja programa. Lanac handler-a obrađuje argumente --ta, --rta, --jdr i --vdr. ArgumentHandler je interface koji deklarira metode obradiArgument() i postaviSljedeceg(). ApstraktniArgumentHandler je apstraktna klasa koja implementira chain logiku - ako handler ne može obraditi argument, prosljeđuje ga sljedećem handleru u lancu.

Četiri konkretna handler-a implementiraju specifičnu logiku:

- TaArgumentHandler - provjerava argument --ta i učitava naziv datoteke s aranžmanima
- RtaArgumentHandler - provjerava argument --rta i učitava naziv datoteke s rezervacijama
- JdrArgumentHandler - provjerava argument --jdr i postavlja JednaRezervacijaStrategija
- VdrArgumentHandler - provjerava argument --vdr i postavlja ViseRezervacijaStrategija

Glavni program (Glavna.java) kreira lanac handler-a povezujući ih metodom postaviSljedeceg(). Prolazi kroz args[] array i poziva prvi handler u lancu za svaki element. Handler koji prepozna svoj argument obrađuje ga i sprema vrijednost u KonfiguracijaAplikacije objekt. Neprepoznati argumenti se preskače uz upozorenje.

Ako ni --jdr ni --vdr nisu zadani, automatski se postavlja NullRezervacijaStrategija koja dozvoljava neograničen broj rezervacija (Null Object pattern).

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

