

Obrazac za zadaću na predmetu "Uzorci dizajna" ak. god. 2022./2023.

Ime i prezime studenta/ice: Petar Mihalić

Matični broj: 0016136283

Dio A. Osnovni podaci o zadaći

R.br.	Pitanje	Odgovor	
1.	Grupa na seminaru:	G1	
2.	Broj i naziv zadaće:	3	Zadaća 3 – Završena brodska luka
3.	Procjena vremena za realizaciju bez decimala):	<u>12</u> sati	
4.	Procjena % završenosti (bez decimala):	<u>75</u> / 100%	
5.	Procjena bodova za izradu zadaće (1 decimala):	<u>11</u> / (DZ3 - 15)	
6.	Želim prezentirati zadaću:	NEMA PREZENTACIJE	
7.	Koji dijelovi iz opisa zadaće nisu realizirani:	MVC / VT100 dio zadaće	
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:	Ne postoje	
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	

Dio B.1. Dokumentacija rješenja 1. zadataće (kopirano i nepromijenjeno)

Naziv uzorka dizajna	Klase koje sudjeluju u uzorku dizajna	Opis razloga odabira uzorka dizajna
Singleton	BrodskoLuka	Odabrao sam klasu BrodskoLuka za implementaciju Singleton uzorka jer je nju potrebno jednom instancirati te koristim metodu getBrodskoLuka za dohvaćanje te jedne instance u drugim klasama za korištenje varijabli iz BrodskoLuka, npr. brojač greški
Factory Method	UcitavanjeSudjelje BrodLoader LukaLoader RasporedLoader ZahtjevRezervacijeLoader	Factory Method sam odabrao za učitavanje datoteka jer omogućuje implementaciju sučelja UcitavanjeSudjelje kojeg nasljeđuju ostale klase za učitavanje pojedinih datoteka tako su u sučelje izdvojene metode koje koriste svaka od ostalih klasa jer su istog formata ali učitavaju različite podatke
Builder	ZahtjevRezervacije ZahtjevRezervacijeLoader BrodskoLuka	Uzorak dizajna Builder sam prepoznao kod klase ZahtjevRezervacije jer object ZahtjevRezervacije je potrebno u nekim slučajevima kreirati uz različite parametre. Builder omogućuje da se po potrebi na objekt dodaju podaci za trajanje veza u satima i status zahtjeva dok su ostali uvijek obavezni.

Dio B.2. Dokumentacija rješenja 2. zadatka

Naziv uzorka dizajna	Klase koje sudjeluju u uzorku dizajna	Status ¹	Opis razloga odabira uzorka dizajna
Singleton	BrodskoLuka	S	Odabrao sam klasu BrodskoLuka za implementaciju Singleton uzorka jer je nju potrebno jednom instancirati te koristim metodu getBrodskoLuka za dohvaćanje te jedne instance u drugim klasama za korištenje varijabli iz BrodskeLuka, npr. brojač greški
Factory Method	UcitavanjeSudjelje BrodLoader LukaLoader RasporedLoader ZahtjevRezervacijeLoader KanalLoader MolLoader MolVezLoader	P	Factory Method sam odabrao za učitavanje datoteka jer omogućuje implementaciju sučelja UcitavanjeSudjelje kojeg nasljeđuju ostale klase za učitavanje pojedinih datoteka tako su u sučelje izdvojene metode koje koriste svaka od ostalih klasa jer su istog formata ali učitavaju različite podatke. U drugoj zadatci dodane klase KanalLoader, MolLoader i MolVezLoader.
Builder	ZahtjevRezervacije ZahtjevRezervacijeLoader BrodskoLuka	S	Uzorak dizajna Builder sam prepoznao kod klase ZahtjevRezervacije jer objekt ZahtjevRezervacije je potrebno u nekim slučajevima kreirati uz različite parametre. Builder omogućuje da se po potrebi na objekt dodaju podaci za trajanje veza u satima i status zahtjeva dok su ostali uvijek obavezni.
Visitor	VrstaVezaVisitor BrojVrstaVezaVisitor ElementVrstaVeza VezPO VezPU VezOS	N	Visitor sam odabrao za zbrajanje ponavljanja svake od vrsta vezova u listi zauzetih vezova. Budući da Visitor predstavlja operaciju koja se izvršava nad elementima objekte strukture kreirao sam klase VezPO, VezPU i VezOS kojima se dodaje operacija brojača za svako ponavljanje zauzetog veza iz pojedine vrste.
Observer	SpojeniBrodoviObserver Brod ObavijestSubject Kanal	N	Uzorak Observer sam prepoznao kod ispisa poruka svim brodovima spojenim na određeni kanal budući da Observer omogućuje pretplate/povezivanja i izlazak iz liste pretplaćenih na obavijesti što je prikladno sa spajanje i odjavljivanje brodova s kanala. Obavijesti su u ovom slučaju poruke koje se generiraju u pojedinom kanalu tokom komunikacije spojenog broda s lukom te se koristi Observer da se poruke šalju svima u kanalu.
Chain of Responsibility	Lanac LanacRaspodjelaPutnika RaspodjelaBrod1 RaspodjelaBrod2 RaspodjelaBrod3 RaspodjelaBrod4 RaspodjelaBrod5	N	Koristio sam uzorak za ponašanje kod nove aktivnosti koja raspoređuje putnike po maksimalno 5 brodova. Koristim Chain za različite mogućnosti raspodjele te ukoliko je raspodjela moguća Handleri prosljeđuju broj putnika tako dugo dok nisu svi raspoređeni.

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

Dio B.3. Dokumentacija rješenja 3. zadatke

Naziv uzorka dizajna	Klase koje sudjeluju u uzorku dizajna	Status ²	Opis razloga odabira uzorka dizajna
Singleton	BrodskoLuka	S	Odabrao sam klasu BrodskoLuka za implementaciju Singleton uzorka jer je nju potrebno jednom instancirati te koristim metodu getBrodskoLuka za dohvaćanje te jedne instance u drugim klasama za korištenje varijabli iz BrodskoLuka, npr. brojač greški
Factory Method	UcitavanjeSudjelje BrodLoader LukaLoader RasporedLoader ZahtjevRezervacijeLoader KanalLoader MolLoader MolVezLoader	S	Factory Method sam odabrao za učitavanje datoteka jer omogućuje implementaciju sučelja UcitavanjeSudjelje kojeg nasljeđuju ostale klase za učitavanje pojedinih datoteka tako su u sučelje izdvojene metode koje koriste svaka od ostalih klasa jer su istog formata ali učitavaju različite podatke. U drugoj zadatci dodane klase KanalLoader, MolLoader i MolVezLoader.
Builder	ZahtjevRezervacije ZahtjevRezervacijeLoader BrodskoLuka	S	Uzorak dizajna Builder sam prepoznao kod klase ZahtjevRezervacije jer objekt ZahtjevRezervacije je potrebno u nekim slučajevima kreirati uz različite parametre. Builder omogućuje da se po potrebi na objekt dodaju podaci za trajanje veza u satima i status zahtjeva dok su ostali uvijek obavezni.
Visitor	VrstaVezaVisitor BrojVrstaVezaVisitor ElementVrstaVeza VezPO VezPU VezOS	S	Visitor sam odabrao za zbrajanje ponavljanja svake od vrsta vezova u listi zauzetih vezova. Budući da Visitor predstavlja operaciju koja se izvršava nad elementima objekte strukture kreirao sam klase VezPO, VezPU i VezOS kojima se dodaje operacija brojača za svako ponavljanje zauzetog veza iz pojedine vrste.
Observer	SpojeniBrodoviObserver Brod ObavijestSubject Kanal	S	Uzorak Observer sam prepoznao kod ispisa poruka svim brodovima spojenim na određeni kanal budući da Observer omogućuje pretplate/povezivanja i izlazak iz liste pretplaćenih na obavijesti što je prikladno sa spajanje i odjavljivanje brodova s kanala. Obavijesti su u ovom slučaju poruke koje se generiraju u pojedinom kanalu tokom komunikacije spojenog broda s lukom te se koristi Observer da se poruke šalju svima u kanalu.
Chain of Responsibility	Lanac LanacRaspodjelaPutnika RaspodjelaBrod1 RaspodjelaBrod2 RaspodjelaBrod3 RaspodjelaBrod4 RaspodjelaBrod5	S	Koristio sam uzorak za ponašanje kod nove aktivnosti koja raspoređuje putnike po maksimalno 5 brodova. Koristim Chain za različite mogućnosti raspodjele te ukoliko je raspodjela moguća Handleri prosljeđuju broj putnika tako dugo dok nisu svi raspoređeni.
Composite	IBrodskoLukaComponent Luka Mol Vez	N	Koristio sam Composite za izradu strukture brodske luke tako da luka, molovi i vezovi čine strukturu stabla. Luka i molovi su komponente koje imaju djecu te sadrže liste elementima ispod njih dok su vezovi rubni elementi bez djece.

² N – dodan u 3. zadatku, P – promijenjen u 3. zadatku, S – bez promjena u 3. zadatku

Iterator	ICollection Collection IEnumerator IEnumerator	N	Iterator sam iskoristio za kretanje po elementima brodske luke, molovima i vezovima. Umjesto u liste podaci se spremaju u Collection te se po njima iterira ovim uzorkom.
Memento	Caretaker Memento Originator	N	Memento sam iskoristio za nove funkcionalnosti pohrane i vraćanja stanja vezova i vremena. Memento omogućuje upravo ono što se od funkcionalnosti traži i olakšava spremanje bez serijalizacije.

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

Implementiran Composite i Iterator uzorak po uputama. Implementirane dodatne funkcionalnosti SPS i VPS kod kojih je korišten uzorak Memento.

Promijenjena klasa Vez.cs tako da sadrži atribut stanja umjesto da se kod naredbe „I“ ispisuje samo dodatni string.

Dio C.2. Opis funkcionalnosti za uzorak dizajna Chain of Responsibility

Raspoređivanje određenog broja putnika tako ali luka omogućava korištenje do 5 brodova odjednom. Potrebno provjeriti stane li broj putnika ako su raspoređeni na 5 brodova s najvećim kapacitetom, ukoliko ne stanu raspoređivanje se ne obavlja. Ukoliko putnici stanu na 5 ili manje brodova uvijek se popunjavaju najveći brodovi a za ostatak putnika se gleda najekonomičniji brod (brod s najmanjim zadovoljivim kapacitetom). Za svaki brod na koji se putnici raspoređuju potrebno je ispisati broj putnika, oznaku i naziv broda. Za zadnji brod na koji su putnici raspoređeni ispisuje se i njegov kapacitet da se vidi koliko je iskorišten.

- Sintaksa:
 - RP [broj_putnika]
- Primjer:
 - RP 7830
- Opis primjera:
 - Traži se raspoređivanje 7830 putnika na maksimalno 5 brodova

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

