

Obrazac za zadaću na predmetu "**Uzorci dizajna**"

Ime i prezime studenta/ice: Marin Mačinković

Matični broj: 00161160219

Dio A. Osnovni podaci o zadaći

R.br.	Pitanje	Odgovor	
1.	Grupa na seminaru:	G2	
2.	Broj i naziv zadaće:	3	mmacinkov_zadaca_3
3.	Procjena vremena za realizaciju bez decimala):	45 sati	
4.	Procjena % završenosti (bez decimala):	80 / 100%	
5.	Procjena bodova za izradu zadaće (1 decimala):	12 / 15	
6.	Želim prezentirati zadaću:	NE	
7.	Koji dijelovi iz opisa zadaće nisu realizirani:	<p>Nije implementiran uzorak dizajna MVC. Kod aktivnosti 7 i 8 obavlja se samo ispis strukture, bez najma i zarade, odnosno računa. Nije realizirana aktivnost 11.</p>	
8.	Postoji li dio zadaće koji vrijedi posebno istaknuti i zašto:	<p>Napomenuo bih par sitnica prije testiranja, primjer za testiranje kod skupnog načina: za aktivnost 1: odabir aktivnosti: 1 → prikaz sintakse → unos aktivnosti → 1; "2020-10-10 00:00:00"; 1; 1: 1. Navedeno se odnosi na sve aktivnosti. Također, pri testiranju je primjereno (radi ljepšeg ispisa i vizualizacije na ekranu) da se koriste sljedeći parametri: tekst=20, cijeli=5, decimali=2.</p>	
9.	Postoje li dijelovi zadaće koji imaju pogrešku u radu i koje:	<p>Nisam znao baratati sa „“ navedenim navodnicima, pa je preporučljivo da se kroz testiranja koriste "" navodnici.</p>	
10.	Da li ste koristili posebne biblioteke u realizaciji zadaće izvan spomenutih na nastavi:	NE	
11.	Da li ste koristili programska rješenja ili dijelove programskog koda od drugih kolega:	NE	

Dio B. Dokumentacija rješenja zadatke

Naziv uzorka dizajna	Klase koje sudjeluju u uzorku dizajna	Opis razloga odabira uzorka dizajna
Singleton	TvrtkaSingleton	Za ime tvrtka umjesto ime vozilo sam se odlučio zbog toga što se u zadaci radi o tvrtci koja upravlja iznajmljivanjem električnih automobila. Smatram naziv tvrtka prikladnijim. Singleton klasa je kreirana da bi se osigurala jedna instanca vozila, odnosno da može postojati samo jedno vozilo, a ne više njih, tijekom izvršavanja programa.
Builder	ICjenikBuilder CjenikConcreteBuilder CjenikBuildDirector IlokacijeVozilaBuilder LokacijeVozilaConcreteBuilder LokacijeVozilaBuildDirector	Budući da se klase Cjenik i LokacijeVozila sastoje od većeg broja atributa s kojima je potrebno baratati na više načina, i povezivati više klasa, korištenje Buildera uvelike olakšava navedeni problem. Korištenjem builder olakšana je konstrukcija samih podataka te se lako postiže čišći kod. Još jedan kandidat je bila klasa Aktivnost, ali nju ipak nisam koristio kao Builder zbog broja atributa koji variraju. Korištenje Buildera mi je uveliko pomoglo pri rješavanju zadatka.
Composite	IComponentTvrtka CompositeTvrtka	Uzorak koji omogućuje hijerarhiju složenih i primitivnih objekata. Koristi se za prikaz hijerarhije odnosno stabla tvrtke, njenih podređenih lokacija te podređenih drugih tvrtki.
Builder	ITvrtkaBuilder TvrtkaConcreteBuilder TvrtkaBuildDirector	Budući da sam koristio Composite za prikaz hijerarhije tvrtke, odlučio sam se i na uzorak dizajna Builder kod tvrtke jer je Composite često ono što Builder gradi, odnosno fino surađuju ova dva uzorka.
Iterator	IIteratorComposite IIteratorLokacija IIteratorTvrtka IteratorTvrtka IteratorLokacija	Također, korišten zbog dobre suradnje sa Compositeom, a koristi se za agregiranje, odnosno iteriranje kroz djecu tvrtke i pregledava dali tvrtka ima potrebnu djecu i kako je ista složena.
Decorator	IDecoratorRedakTablice DecoratorBroj DecoratorText DecoratorRedakTablice DecoratorKonkretniRedak	Decorator je uzorak koji sam koristio jer mi omogućava lako ispisivanje podataka i njihovo formatiranje u tabličnom obliku.
State	IStateOsnovno StateNaPunjenju StateNeispravno StateSlobodno StateUnajmljeno StateUnajmljeno2	State je uzorak koji sam koristio ponajviše jer je bio obavezan za implementaciju u ovoj zadaci, ali uz to omogućio mi je lako prolaženje kroz stanja vozila i ispis i prikaz konkretnih stanja vozila.
ChainOfResponsibility	AbstractHandler EvidencijaHandler ObradaHandler PretragaHandler	ChainOfResponsibility je zadan za korištenje zbog baratanja računima, i idejno bilo ga je potrebno implementirati da obnaša tu funkciju.

Dio C. Opis promjena u odnosu na prethodnu zadaću (osim kod 1. zadaće)

1. Refaktoriran veliki dio koda da bi se prilagodilo novim uzorcima, kao i npr. kod korisnika je dodan novi redak za praćenje ugovora.
2. Riješeni su buggovi koji su se događali pri iznajmljivanju i vraćanju vozila.
3. U tvrtci su lokacije promjenjene u listu da bi se lakše mogli koristiti uzorci Iterator i Composite
4. Uspješno je implementiran Decorator s kojim je bilo problema u prošloj zadaći
5. Implementirane su funkcionalnosti 6, 7 i 8 koje su nedostajale u prošloj zadaći.
6. Dodane su razne provjere za elemente aktivnosti, kao npr. za provjeru formata datuma.

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

