

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

Ime i prezime studenta/ice: _____

Matični broj: _____

Dio A. Osnovni podaci o zadaći

R.br.	Pitanje	Odgovor	
1.	Grupa na seminaru:	G1	
2.	Broj i naziv zadaće:	3	Napredno upravljanje programima TV kuće
3.	Procjena vremena za realizaciju bez decimala):	30 sati	
4.	Procjena % završenosti (bez decimala):	95 / 100%	
5.	Procjena bodova za izradu zadaće (maks 10, 1 decimala):	9.5 / 10	
6.	Želim prezentirati zadaću:	NE	
7.	Koji dijelovi iz opisa zadaće nisu realizirani:	Popunjavanje rasporeda prema prioritetu. Ne dodaju se emisije bez definiranih dana i vremena.	
8.	Postoji li dio zadaće koji vrijedi posebno istaknuti i zašto:	NE	
9.	Postoje li dijelovi zadaće koji imaju pogrešku u radu i koje:	NE	
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 zadaće

Naziv uzorka dizajna	Klase koje sudjeluju u uzorku dizajna	Opis razloga odabira uzorka dizajna
Singleton	IspisOsobaUlogaSingleton TjedniRasporedSingleton	Kroz zadatak se više puta ispisuju osoba i njegova/njezina uloga, stoga je ispis realiziran preko Singleton klase kako bi se instanca stvorila samo prilikom prvog poziva. Pošto se u tjedni raspored dodaju programi, dani i emisije koji se preuzimaju u različitim klasama, najlakša solucija je napraviti raspored kao singleton. Svakim idućim pozivom vraća se postojeća instanca koja je stvorena prvi put i moguće su operacije sa postojećom instancom.
Factory Method	PreuzimanjeFactory PreuzmiOsobe PreuzmiUloge PreuzmiTvKuce PreuzmiPrograme PreuzmiEmisije PreuzmiVrste Preuzmi	Retci koji su učitani iz datoteka mogu predstavljati osobe, uloge, emisije, vrste, TV kuće ili programe. Učitane retke iz datoteka potrebno je interpretirati kao liste osoba, uloga, emisija, vrste, TV kuća ili programa. PreuzimanjeFactory pomoću definiranih podklasa radi tu interpretaciju. Podklase definiraju o kojoj je vrsti objekta riječ.
Builder	EmisijaBuilderImpl EmisijaBuilderDirector Emisija	Iako se Builder uzorak koristi za konstrukciju složenih objekata korak po korak, gdje posljednji korak vraća objekt, ovdje je Builder implementiran za relativno jednostavan objekt Emisija kako bi se demonstrirao način na koji on funkcionira.
Composite	DaniUTjednu EmisijaUProgramu RasporedComposite TjedniRaspored TvKuca	Svaka TvKuca sadrži program koji sadrži dane emitiranja, a dani emitiranja sadrže emisije. Composite nam omogućuje vezu dio-cjelina.
Decorator	BasicDecorator Decorator DecoratorEmisijaUProgramu DecoratorOsobaUloga DecoratorReklame StringBasicDecorator StringDecorator StringDecoratorEmisija StringDecoratorPohranjivanje	Decorator se koristi za kreiranje tabličnih ispisa za emisije u programima i prikaz osoba sa ulogama u emisija. Decorator sa prefiksom String prima string koji se splita i zatim ispisuje u tabličnom obliku, dok Decorator bez prefiksa prima EmisijaUProgramu i ispisuje podatke u tabličnom obliku.
Iterator	AbstractEmisijeIterator EmisijeIterator SveEmisijeIterator VrsteEmisijaIterator	Iterator je odabrani za iteriranje kroz programe u kojima se emisija nalazi. Razlog odabira Iteratora je zato što pruža mogućnost pamćenja trenutne pozicije obrade, jednostavan dohvat trenutnog elementa i iteracije na sljedeći.
Observer	Observer Subject Osoba EmisijaUProgramu	Observer određuje komunikaciju između Osobe(Subject) i EmisijeUProgramu(Observer). Osoba je objekt koji obavještava EmisijeUProgramu o promjenama u njegovom stanju.
Visitor	Element ElementVisitor ElementVisitorImpl	Visitor je odabrani za izračun broja minuta reklama zato što omogućuje definiranje nove operacije bez da mijenjamo klasu elemenata na kojima djeluje.
Memento	Caretaker Memento Originator	Memento je odabran kako bismo mogli pamtit stanja i kasnije se vraćati na ista putem jednoznačnih identifikatora.

Chain Of Responsibility	Chain KratkaEmisija SrednjeDugaEmisija DugaEmisija	Chain Of Responsibility je odabrani za kategoriziranje emisija prema njihovom vremenu trajanja. Objekti iz liste emisija primaju i prenose zahtjeve (u koju kategoriju svrstati emisiju) duž lanca sve dok ga jedan od njih ne preuzme
Model-View-Controller	Controler DefaultPogled FormatPogled	MVC se koristi kako bismo mogli definirati poglede koji prikazuju drugačije formate ispisa.

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

- kreirani Decorator za tablični prikaz prihoda reklama
- kreirani Decoratori koji kao parametar uzimaju String koji se kasnije razdvaja i ispisuje u pravilnom formatu
- izračun broja minuta reklama implementiran je pomoću Visitor-a
- dodani jednoznačni redni broj za svaku emisiju u programu
- dodana funkcionalnost brisanja emisije iz programa
- dodana funkcionalnost spremanja trenutnog stanja programa u TV kući pomoću Memento uzorka
- dodana funkcionalnost ispisa svih dosadašnjih spremanja stanja
- dodana funkcionalnost povratka podataka na neko od prethodnih spremljenih stanja putem jednoznačnog identifikatora pohrane stanja pomoću Memento uzorka
- dodana funkcionalnost promjene pogleda putem Model-View-Controller uzorka

Vlastita funkcionalnost:

Kategoriziranje emisija prema njihovom trajanju pomoću Chain Of Responsibility uzorka. Emisije su podijeljene u tri kategorije: kratka emisija (0-30 min), srednje duga emisija (30-60 min) i duga emisija (60+ min).

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

