

*Izjavljujem da tijekom izrade ove zadaće neću od drugoga primiti niti drugome pružiti pomoć, te da se neću koristiti nedopuštenim sredstvima. Ove su radnje teška povreda Kodeksa ponašanja te mogu uzrokovati i trajno isključenje s Fakulteta. Također izjavljujem da mi zdravstveno stanje dozvoljava pisanje ove zadaće.*

---

JMBAG

Ime i prezime

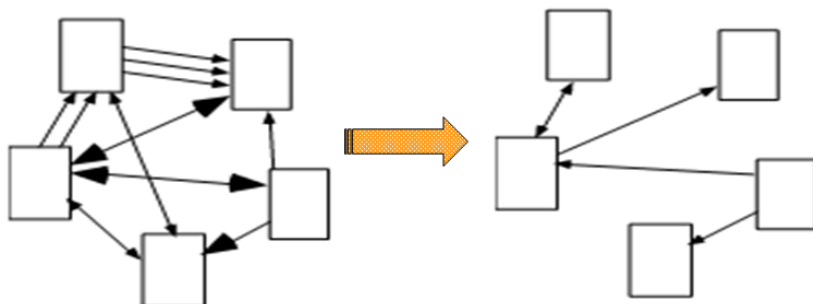
Vlastoručni potpis

---

### GRUPA A

1. (1 bod) Složenost, usklađenost, promjenjivost i nevidljivost su suštinski (inherentni, nezaobilazni) problemi razvoja programske potpore.
2. (1 bod) Arhitektura programske podrške mora rezultirati dokumentacijom koja ima tri svojstva: \_\_dobra\_\_, \_\_ispravna\_\_ i \_\_uspješna\_\_.
3. (1 bod) Problemi zahtjeva domene primjene su: \_\_razumljivost i implicitnost\_\_.
4. (1 bod) Model ubrzanog razvoja (engl. *Agile modeling*) programske potpore posebno je pogodan za \_\_brzi razvoj manjih i srednjih\_\_ projekata a karakterizira ga \_\_stalna\_\_ interakcija s klijentima.
5. (1 bod) Kod UML dijagrama obrazaca uporabe, višestrukosti se mogu pisati na kojoj vrsti veza? \_\_Na vezi pridruživanja (asocijacije)\_\_
6. (1 bod) Pravokutnikom na životnoj liniji (engl. *lifeline*) objekta prikazuje se \_\_aktivnost objekta\_\_.
7. (1 bod) Kod unificiranog procesa razvoja programske potpore, jezgrena aktivnost analize i oblikovanja programske potpore provodi se u najvećoj mjeri tijekom faze \_\_razrade (engl. elaboration)\_\_.
8. (1 bod) Projektni dnevnik (engl. *product backlog*), dnevnik sprinta (engl. *sprint backlog*) i \_\_inkrement\_\_ su Scrum artefakti.
9. (1 bod) Konceptijska arhitektura (engl. *Conceptual Architecture*) programske potpore usmjerena je na pogodnu dekompoziciju sustava i komunikaciju s \_\_netehničkim dionicima (uprava, prodaja, korisnici)\_\_.

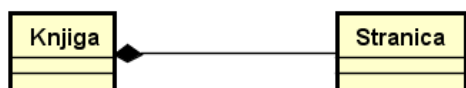
10. (1 bod) Primjena kojeg principa oblikovanja objektno usmjerene arhitekture programske potpore vodi do transformacije arhitekture prema slici?



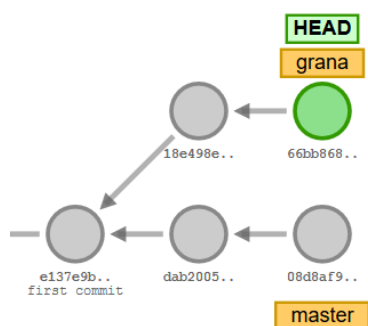
Smanjivanje međuovisnosti (engl. Reduce coupling where possible)

11. (1 bod) Dobri dokumenti oblikovanja arhitekture, kao pomoć izradi bolje programske potpore, pišu se s gledišta čitatelja i služe kao sredstvo komunikacije tima za implementaciju/budućih osoba zaduženih za održavanje, razvoj ili promjene/osobama uključenih u oblikovanje povezanih sustava ili podsustava.

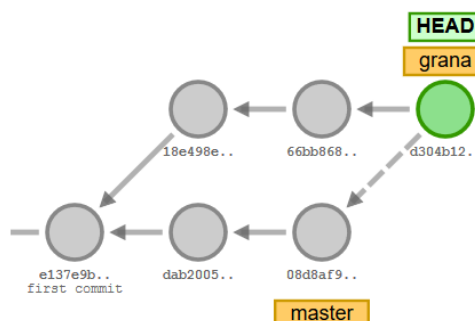
12. (1 bod) Nacrtajte grafičku oznaku sadržavanja UML dijagrama razreda kojom se naglašava ovisnost razreda o životnom ciklusu drugog razreda i prikažite primjer dva razreda u takvoj vezi.



13. (1 bod) Kojom Git naredbom repozitorij pohranjuje promjene iz grane „master“ u granu „grana“, tj. prelazi iz stanja sa slike a) u stanje sa slike b)?



slika a)



slika b)

git merge master

## Problemski dio - Aparat za izradu proteinskih napitaka

Potrebno je modelirati aparat za automatsku pripremu proteinskih napitaka u teretanama s automatskom centralnom nabavom. Aparat se izvorna sastoji od ekrana osjetljivog na dodir na kojem korisnici odabiru željeni napitak, utora za ubacivanje gotovine, prostora za povrat gotovine i prostora za predaju izrađenog napitka. Iznutra, aparat ima spremnik gotovine, upravljački modul koji provodi sve potrebne operacije i ostvareno sučelje za povezivanje s centralnim sustavom nabave.

Korisnik najprije odabire jedan od 10 ponuđenih proteinskih napitaka. Zatim odabire bazu za pripremu napitka koja može biti voda ili mlijeko. Aparat putem osjetnika očitava količinu sastojaka za pripremu napitka i ako nema dovoljne količine sastojaka, korisniku se treba ispisati odgovarajuća poruka na zaslonu i prekinuti kupnja. Ako je sve u redu, aparat ispisuje korisniku cijenu napitka. Iznos ubačene gotovine se uvećava na ekranu svaki put kada korisnik ubaci gotovinu. Odabirom potvrde kupnje, aparat izračunava je li ubačeni iznos veći ili jednak cijeni napitka. Ako je ubačeni iznos gotovine manji od cijene napitka korisniku se treba ispisati odgovarajuća poruka na zaslonu te vratiti ubačenu gotovinu. Ovisno o prethodnom odabiru sastava, proces izrade napitka je zasnovan na recepturi na bazi vode ili na bazi mlijeka. Kada je uređaj gotov s izradom napitka, korisnik može preuzeti napitak, te mu se, ako je ubačen iznos veći od potrebnog, vraća višak gotovine.

Korisnik može odustati od kupnje prije potvrde kupnje, pri čemu mu se vraća sva ubačena gotovina. Po završetku interakcije s korisnikom, zaslon se vraća u početno stanje.

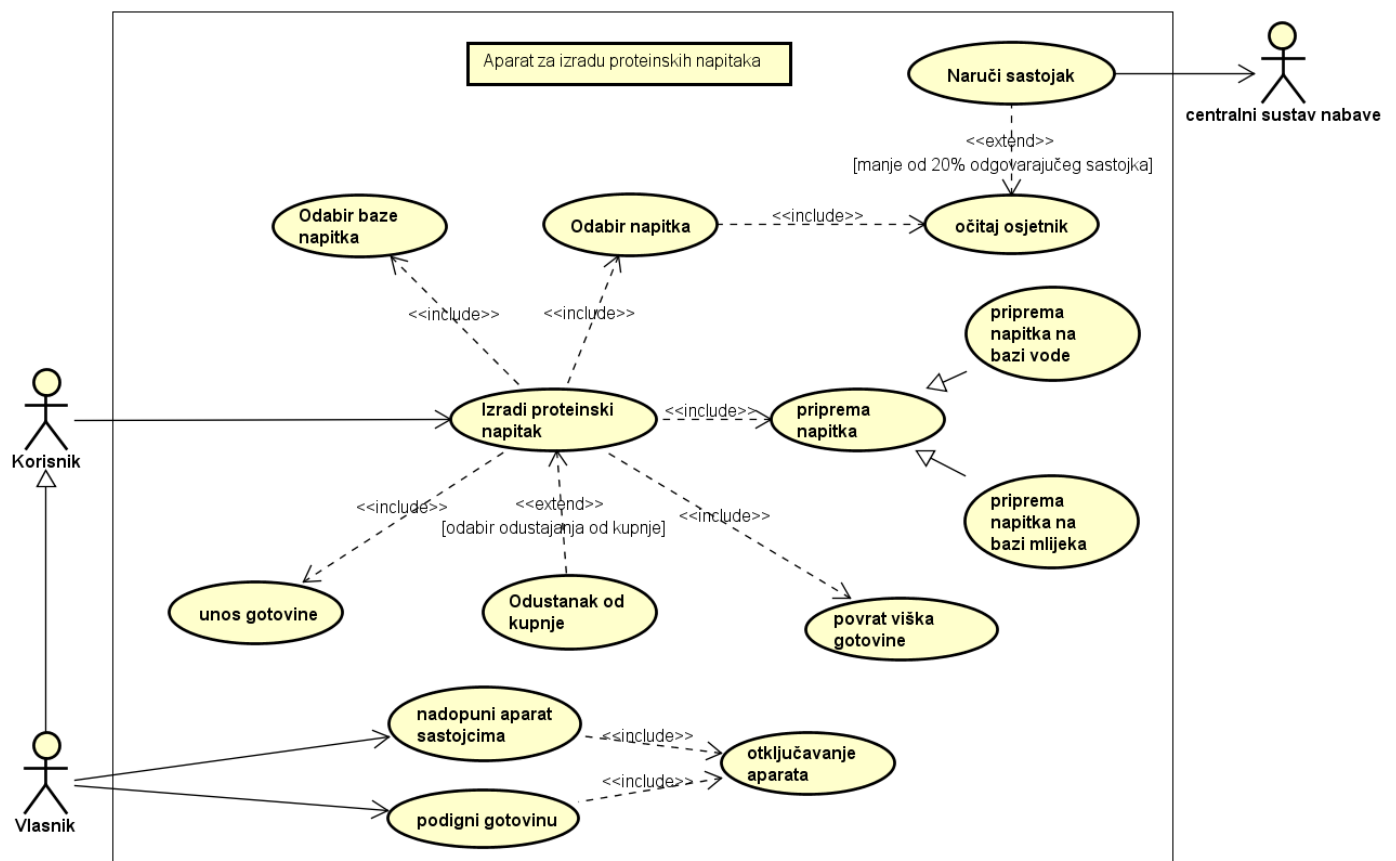
Uređaj automatski, kada razina nekog od 5 sastojaka padne ispod 20%, šalje narudžbu za taj sastojak centralnom sustavu nabave.

Vlasnik aparata može pomoću RFID čipa otključati aparat te zatim nadopuniti aparat sa sastojcima za izradu napitaka ili podići gotovinu od kupnje napitaka.

*NAPOMENA: Prilikom izrade pojedinih dijagrama uz opći opis uzmite u obzir te pošaljite i sljedeće eksplicitno navedene zahtjeve uz pojedine zadatke.*

### 14. (4 boda) Dijagram obrazaca uporabe

UML dijagramom obrazaca uporabe modelirajte aparat za izradu proteinskih napitaka.



### 1. Uočiti aktore kao imenice koje su povezane s nekom akcijom u sustavu:

- korisnik,
- vlasnik,
- centralni sustav nabave

Imenice koje su dio sustava i ne modeliraju se kao aktor:

- utor za ubacivanje gotovine, prostor za povrat gotovine, Ekran, osjetnik

### 2. Uočiti glagole ili glagolske imenice koje opisuju specifične akcije aktora:

- odabir željenog napitka
- odabir baze napitka
- odabir potvrde kupnje
- očitavanje osjetnika
- vraćanje gotovine za 2 slučaja (odustanak od kupnje, previše ubačeno gotovine)
- odustajanje od kupnje
- slanje narudžbe centralnom sustavu
- otključavanje aparata
- nadopunjavanje aparata sastojcima
- podizanje gotovine

Temeljem toga utvrđeni su obrasci uporabe (engl. *use case*) koji opisuju sekvence akcija (uključujući varijante) koje sustav obavlja u interakciji s aktorima:

- Izrada proteinskog napitka što uključuje:
  - o Include - Odabir željenog napitka koji uključuje
    - Očitavanje osjetnika, te ako ima manje od 20% onda
      - Extend - Naručivanje sastojka
  - o Include - Odabir baze napitka (s specifikacijama za mlijeko i vodu)
  - o Include - Unos gotovine
  - o Include - Povrat viška gotovine
  - o Extend - Odustanak od kupnje
  - o Include - Pripravljivanje napitka(uz uvjet da je dovoljno novaca ubačeno i postoji odabrani napitak)

- Nadopunjavanje aparata sastojcima koji uključuje(include) otključavanje aparata
- Podizanje gotovine koji uključuje(include) otključavanje aparata

Neke od akcija mogu se, ali i ne moraju prikazati kao zaseban obrazac uporabe:

- Dojava poruke za nedostatak gotovine

### 3. Povezivanje aktora s obrascima uporabe i određivanje aktivnih i pasivnih aktora.

- Pasivni aktor je centralni sustav nabave ,
- Aktivni aktori su korisnik i vlasnik.

Vlasnik je specifikacija korisnika.

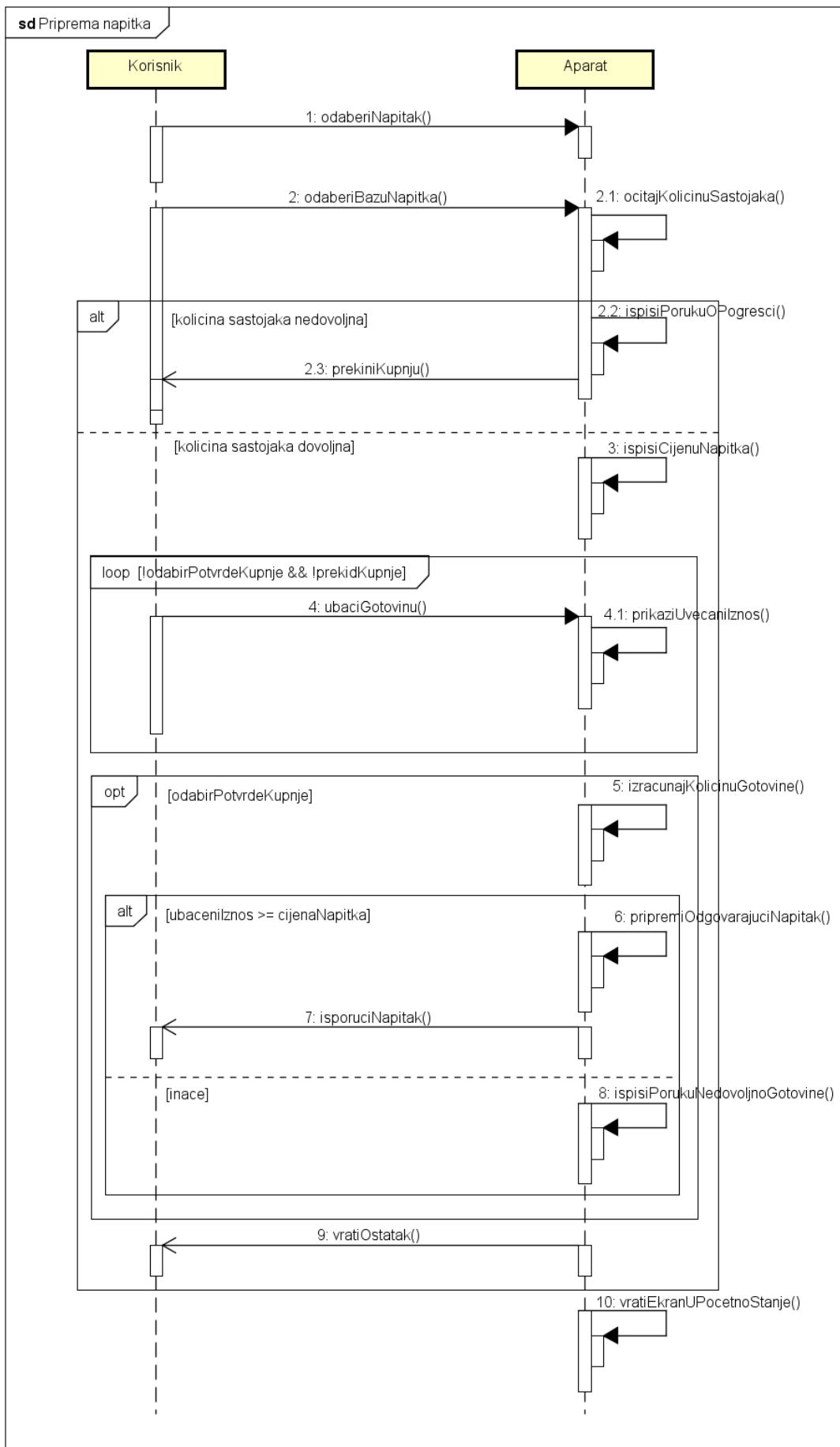
### 4. Prikazati granice sustava i imenovati sustav.

#### Bodovanje:

- Aktori – 1 bod
- Obrasci uporabe - 1 bod
- Veze između obrazaca i aktora – 0.5 boda
- Veze između obrazaca uporabe – 1 bod
- Granica sustava i imenovanje dijagrama – 0.5 bodova

### 15. (4 boda) Sekvencijski dijagram

Sekvencijskim UML-dijagramom prikažite postupak izrade proteinskog napitka počevši od odabira napitka.



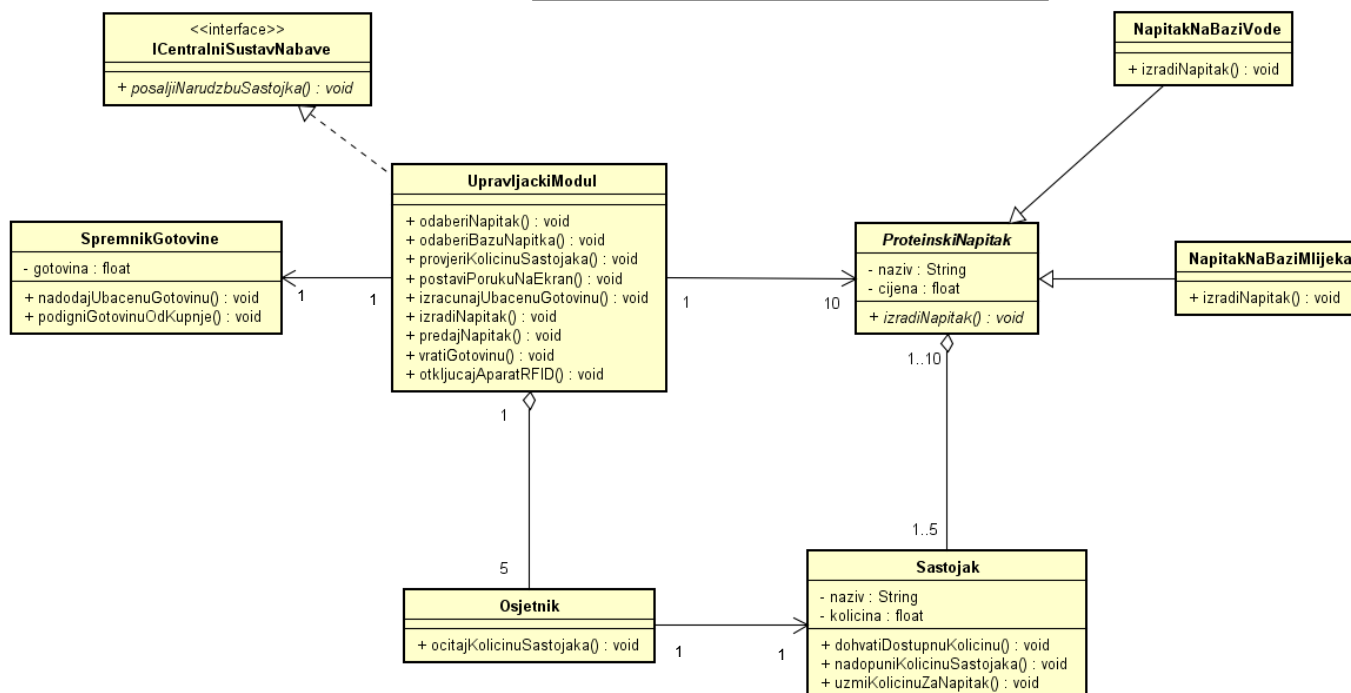
### Postupak rješavanja sekvencijskog dijagrama:

1. Za životne linije izabiru se oni aktori iz dijagrama obrazaca uporabe koji imaju veze s izradom proteinskog napitka:
  - To je samo "Korisnik".
  - Dodatno, uvodi se aktor "Aparat" kako bi se opisala komunikacija korisnika s modeliranim sustavom.
  - Vanjske dijelove aparata nije potrebno modelirati kao zasebne aktore (npr. utor za ubacivanje gotovine).
2. Izrada proteinskog napitka opisana je u dijelu teksta koji počinje s rečenicom:  
"Korisnik najprije odabire jedan od 10 ponuđenih proteinskih napitaka. ..."  
a završava s rečenicom:  
"... Po završetku interakcije s korisnikom, zaslon se vraća u početno stanje."
3. Potrebno je uočiti redoslijed odvijanja akcija i uvjete pod kojima se pojedine akcije odvijaju.  
Redoslijed je:
  - 3.1. Odabir napitka
  - 3.2. Odabir baze napitka
  - 3.3. Očitavanje sastojaka
  - 3.4. Unos gotovine (provodi se u petlji)
  - 3.5. Potvrda kupnje ili odustajanje od kupnje
  - 3.6. Preuzimanje napitka
  - 3.7. Povrat viška gotovine
  - 3.8. Postavljanje zaslona u početno stanje
4. Također, ako dođe do bilo kakve pogreške, ispisuje se na zaslonu odgovarajuća poruka i prekida se proces izrade napitka.
5. Važno je modelirati petlju (okvir "loop") kod unosa gotovine koja se provodi sve dok korisnik ne odabere potvrdu kupnje ili prekid kupnje.
6. Uvjeti se rješavaju odgovarajućim okvirima alt (za slučaj dviju mogućnosti - if/else) ili opt (za slučaj jedne ili više opcija), pri čemu se uvjeti pišu za svaki fragment okvira.
7. Gotovina se, prema tekstu zadatka, vraća u tri situacije: ako je kupnja prekinuta, ako je kupnja potvrđena, ali ubačene gotovine nema dovoljno ili ako je korisnik uplatio više od potrebnog iznosa gotovine.
8. Povrat gotovine se može modelirati na više načina u dijagramu, ali idealno je, zbog jednostavnosti modela, koristiti samo jednu poruku povrata ostatka gotovine, koja se navodi na kraju fragmenta kada je sastojaka za izradu napitka bilo dovoljno, jer su time pokrivena sva tri slučaja.

## 16. (4 boda) Dijagram razreda

UML dijagramom razreda modelirajte programsku potporu aparata za izradu proteinskih napitaka. Programski sustav aparata ima upravljački modul koji je odgovoran za komunikaciju s korisnicima i otkučavanje aparata za vlasnika. Pretpostavite da aparat ima po jedan osjetnik za svaki od sastojaka, a da se napitak izrađuje od određene količine pojedinih sastojaka. Korisnike, vlasnika i vanjsko sučelje aparata (zaslon, utor za gotovinu ...) ne treba modelirati. Također, pri modeliranju nije potrebno navoditi parametre metoda kao niti nazive veza ili uloga..

Model programske potpore aparata za izradu proteinskih napitaka



Postupak rješavanja dijagrama razreda:

### 1. Identificiranje razreda

Uočiti bitne imenice iz teksta zadatka koje opisuju unutrašnjost aparata:

- spremnik gotovine
- upravljački modul
- sučelje prema centralnom sustavu nabave
- proteinski napitak
- osjetnik
- sastojak

Imenice koje ne opisuju unutrašnjost aparata se ne prikazuju kao razredi (korisnik, vlasnik, RFID čip, utor za unos gotovine...)

Sučelje prema centralnom sustavu nabave se modelira kao sučelje (stereotip <<interface>>)

Potrebno je uočiti da postoje dvije baze napitka, mliječna baza i vodena baza te se one također modeliraju kao razredi.

### 2. Određivanje odnosa među razredima

Upravljački modul, budući da treba upravljati svim operacijama unutar aparata, treba biti povezan sa spremnikom gotovine, proteinskim napitkom i osjetnicima.

Pritom je razumno pretpostaviti da će upravljački modul sadržavati osjetnike (agregacija), a da je s ostalim razredima povezan običnom asocijacijom.

Upravljački modul realizira (implementira) sučelje prema centralnom sustavu nabave kako bi ostvario mogućnost automatske nabave sastojaka.

Generalizacijom modeliramo odnos između apstraktnog napitka (jer mu se još ne zna baza) i specifične, konkretne baze napitka (mlijeko ili voda).

Osjetnik je povezan asocijacijom s odgovarajućim sastojkom.

Proteinski napitak sadržava sastojke te je stoga povezan agregacijom sa sastojkom.

### 3. Određivanje višestrukosti pridruživanja

Svugdje gdje ima više od jednog objekta na nekoj strani pridruživanja potrebno je to označiti s odgovarajućim brojem primjeraka objekata ili oznakom.

To je slučaj s napitcima, kojih ima točno 10 te sastojcima, kojih ima točno 5, tako da bilo koje veze povezane s tim razredima trebaju imati takve višestrukosti.

#### **4. Definiranje operacija i atributa za pojedine razrede**

U zadatku je jasno rečeno da upravljački modul ostvaruje sve metode za komunikaciju s korisnikom te otključavanje aparata od strane vlasnika.

Stoga taj razred ima i najviše metoda koje se mogu većinom iščitati iz točno riješenog sekvencijskog dijagrama (ima ih ukupno 9).

Apstraktnu metodu izrade napitka u razredu proteinski napitak implementiraju konkretni razredi baze napitka.

Osjetnici trebaju očitati količinu sastojaka.

Sastojak ostvaruje metode nadopunjavanja količine, očitavanja količine te uzimanja količine (za određeni napitak).

U spremnik gotovine nadodaje se ubačena gotovina, a vlasnik može i podići skupljenu gotovinu.

Od atributa, potrebno je modelirati količinu sastojaka, cijenu i naziv napitka i gotovinu u spremniku.

Ostalo je opcionalno (npr. naziv sastojka).

Načelno, sve metode su javne, a svi atributi privatni.

Privatnim atributima pristupa se preko get i set metoda, što nije potrebno prikazivati, kao niti konstruktore.

#### **5. Imenovanje dijagrama**

Dijagram je potrebno imenovati (npr. „Model programske potpore aparata za izradu proteinskih napitaka“).

#### **Bodovanje**

- Dobro identificirani razredi – 1.25 bodova
- Veze između razreda – 1.25 bodova
- atributi i metode – 1 bod
- višestrukost – 0.5 bodova