



Oblikovanje programske potpore Završni ispit

9. 2. 2016.



JMBAG

Ime i prezime

Vlastoručni potpis

Minimum za prolaz je 12 bodova, maksimum 36 bodova

GRUPA A

1. (1 bod) Navedite generičke aktivnosti u svim procesima programskog inženjerstva.

Rješenje: Specifikacija, oblikovanje i implementacija, validacija i verifikacija, evolucija

2. (1 bod) Kako se nazivaju programski alati koji podupiru aktivnosti procesa programskog inženjerstva?

Rješenje: CASE , alati Računalom podržano programsko inženjerstvo ...

3. (1 bod) Navedite barem dvije značajke metodologije ubrzanog razvoja (engl. *agile methodology*) programske potpore.

Rješenje: Iterativni razvoj, mali inkrementi, kontinuirano poboljšanje PP, naglasak na ljude i suradnju, uključenost korisnika u proces razvoja...

4. (1 bod) Navedite na koje ste sve načine u projektnoj dokumentaciji izrazili korisničke zahtjeve.

Rješenje: UC-ovi s pripadajućim dijagramima (funkcionalni zahtjevi), sekv. dijagrami, lista nefunkcionalnih zahtjeva

5. (1 bod) Što su zahtjevi domene primjene?

Rješenje: Funkc. i nefunkc. zahtjevi koji proizlaze iz domene primjene/specifični su za domenu primjene/karakteriziraju domenu primjene

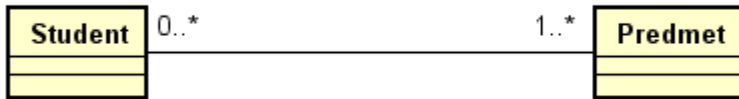
6. (1 bod) Koji se modifikatori vidljivosti operacija koriste u OO paradigmi kako bi se postigla enkapsulacija? Prikažite kako se označavaju na UML-dijagramima razreda.

Rješenje: Public +, private - , protected #... mogu navesti i package (ali ne moraju)

7. (1 bod) Koji minimalni uvjet mora ispuniti neki razred da bi bio apstraktni razred?
Može li se instancirati takav razred?

Rješenje: Razred koji ima barem jednu apstraktnu metodu. Ne može.

8. (1 bod) Prema donjem dijagramu razreda odgovorite koliko najviše predmeta može svaki student upisati te mora li uopće upisati predmet?



Rješenje: Student može upisati neograničeni (nedefinirani) broj predmeta, ali mora upisati barem 1.

9. (1 bod) Nacrtajte dijagram komponenti s dvije komponente koje su povezane jednim sučeljem. Navedite i objasnite vrste sučelja komponenti.

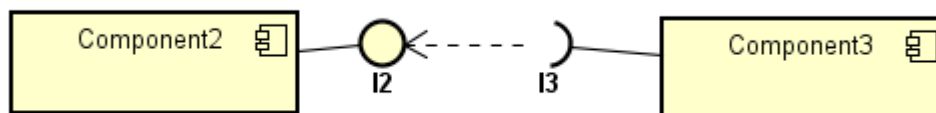
Rješenje:

Ponudeno sučelje (eksportirano sučelje) – skup usluga/metoda koje komponenta realizira (nudi).

Zahtijevano sučelje (importirano sučelje) – skup usluga/metoda koje komponenta koristi/poziva od druge komponente.



Može i ovako



10. (1 bod) Navedite i kratko opišite elemente UML-dijagrama razmještaja.

Rješenje: Čvorovi - uređaji i okolina izvođenja; Komponente (programski artefakti); Spojevi (veze) između čvorova i između komponenti

11. (1 bod) Kako je organiziran primjenski program u uslužnoj arhitekturi?

Rješenje: Uslužno usmjerena arhitektura organizira primjenski program (cjelovitu aplikaciju) kao kolekciju usluga koje međusobno komuniciraju uporabom dobro definiranih sučelja

12. (1 bod) Koji su minimalni elementi kojima se opisuju oblikovni obrasci (engl. *design pattern*)?

Rješenje:

<i>Naziv</i>	<i>Razumljivo ime</i>
<i>Opis</i>	<i>Opis problema</i>
<i>Rješenje</i>	<i>Opis predložka koji može biti upotrijebljen na različite načine</i>
<i>Posljedice</i>	<i>Rezultati i pogodnosti primjene obrasca uporabe</i>

13. (1 bod) Navedite kojoj skupini ili kojim skupinama pripadaju metode u razredu AbstractClient OCSF arhitekture koje se ne redefiniraju.

Rješenje: <<control>> i <<accessor>>

AbstractClient
<pre><<control>> openConnection () sendToServer () closeConnection () <<hook>> connectionEstablished () connectionClosed () connectionException () <<slot>> handleMessageFromServer () <<accessor>> isConnected () getPort () setPort () getHost () setHost () getInetAddress ()</pre>

14. (1 bod) Za slučaj arhitekture klijent-poslužitelj kada postoji n spojenih klijenata izračunajte minimalan broj dretvi pri radu poslužitelja implementiranog objektno usmjerenim radnim okvirom OCSF (NAPOMENA: zanemarite administracijske dretve, dretve OS-a, VM,...).

Rješenje: $n+1$

15. (1 bod) Koji je cilj provođenja aktivnosti ispitivanja u procesu oblikovanja programske potpore?

Rješenje: Otkrivanje informacija o ispravnosti i kvaliteti, te poboljšanja pronalaženjem kvarova i problema ispitivane programske podrške.

Priznaje se i kraće rješenje: pronalaženje pogrešaka.

16. (1 bod) Na koji način možemo odrediti broj mogućih putova koji minimalno jednom pokrivaju izvođenje svih naredbi i uvjeta, uz pretpostavku dostupnosti izvornog koda?

Rješenje: Korištenjem teorije grafova – izračun broja linearno neovisnih putova - ciklomatska složenost $CV(G) = \text{Lukovi} - \text{Čvorovi} + 2 \cdot P$

17. (1 bod) Primjenom tehnike kombinacijskog ispitivanja (engl. *Combination testing*) potrebno je ispitati funkciju koja kao ulaze prima tri dvoznamenkasta cijela broja koji se unose tipkovnicom. Izračunajte broj potrebnih ispitnih slučajeva.

Rješenje: ulazne vrijednosti -99..99

$199 \cdot 199 \cdot 199 = 7\,880\,599$

bilo bi izvrsno da studenti prokomentiraju i neispravne vrijednosti ulaza.

18. (1 bod) Navedite elemente dokumentacije ispitnih slučajeva (engl. *test case*).

Rješenje: Minimalno: Ulaz, Očekivani izlaz, Stvarni rezultat

Dodatno: ime (naziv) ispitnog slučaja, status (anomalije), kako ispitivati modul ili funkciju, opis stanja prije ispitivanja, funkcije koja se ispituje...

19. (1 bod) U timskom razvoju programske potpore pojedine dijelove istodobno razvija više članova tima. Tim koristi GIT sustav za upravljanje inačicama datoteka (engl. *Version control, revision control, source control*). Gdje su smješteni podaci koje mijenjaju pojedini članovi tima? Obrazložite.

Rješenje: u lokalnom repozitoriju. Članovi tima mijenjaju podatke u svojim lokalnim repozitorijima.

20. (1 bod) Formalan logički sustav je uređeni par (Γ, L) . Kako definiramo model formalnog logičkog sustava?

Rješenje: Neka interpretacija je model FLS ako evaluira sve njegove formule Γ u istinito.

21. (2 boda) Definirajte potrebne predikate i konstante te preslikajte rečenice u dobro definirane formule predikatne logike prvoga reda:

"Neki ispravan i kompletan formalan sustav može biti neodlučljiv."

Propozicijska logika je formalan sustav koji je ispravan, kompletan i odlučljiv."

Rješenje:

formalan_sustav(x) - x je formalan sustav

ispravan(x) - x je ispravan

kompletan(x) - x je kompletan

odlučljiv(x) - x je odlučljiv

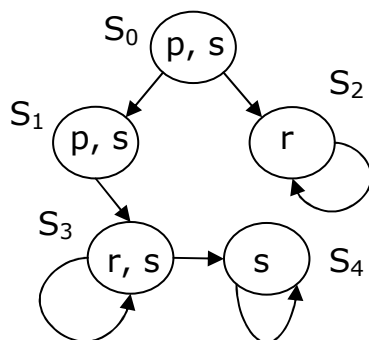
Propozicijska_logika = konst.

$\exists x (\text{formalan_sustav}(x) \wedge \text{ispravan}(x) \wedge \text{kompletan}(x) \wedge \neg \text{odlučljiv}(x))$

*$\text{formalan_sustav}(\text{Propozicijska_logika}) \wedge \text{ispravan}(\text{Propozicijska_logika}) \wedge$
 $\text{kompletan}(\text{Propozicijska_logika}) \wedge \text{odlučljiv}(\text{Propozicijska_logika})$*

22. (2 boda) Za zadani model implementacije M Kripke strukturom prema slici potrebno je:

- Odrediti S (skup stanja), R (relaciju prijelaza), L (funkciju označavanja).
- Odrediti sva stanja koja zadovoljavaju formulu **A** ($p \vee (r \wedge s)$).



Rješenje:

a)

$S = \{S_0, S_1, S_2, S_3, S_4\}$

$R = \{(S_0, S_1), (S_0, S_2), (S_1, S_3), (S_3, S_3), (S_3, S_4), (S_4, S_2), (S_4, S_4)\}$.

$L(S_0) = \{p, s\}; L(S_1) = \{p, s\}; L(S_2) = \{r\}; L(S_3) = \{r, s\}; L(S_4) = \{s\}$

b) S_1 i S_3

Problemški dio - Opis zadatka

Kolodvor Zagreb uvodi aparate za prodaju željezničkih karata kako bi ubrzali poslovanje i poboljšali razinu usluge.

Aparat na zaslonu osjetljivom na dodir prikazuje sve potrebne sadržaje o tijeku kupnje karte i poruku o stanju aparata. Osim zaslona, aparat još ima:

1. utor za prihvrat bankovnih kartica
2. utor za prihvrat papirnatih novčanica
3. utor za prihvrat kovanica
4. pisač karata
5. prostor za povrat kovanica
6. gumb za odustajanje od kupnje

Početno stanje zaslona omogućava odabir jezika. Nakon odabira jezika korisnik zadaje odredište i datum polaska. Prema odabranom odredištu i datumu polaska, na zaslonu se prikazuju moguća vremena polaska vlaka te detaljne informacije o svakom vlaku: broj vlaka, vrsta vlaka, peron i kolosijek. Korisnik bira željeni vlak. Nakon toga, korisnik još izabire razred smještaja u vlaku. Korisniku se zatim ispisuju sve prethodno odabrane informacije na zaslonu, uključujući i cijenu te mu se nudi izbor sredstva plaćanja: bankovna kartica ili gotovina.

Po izboru sredstva plaćanja, automat otvara odgovarajući utor za unos sredstva. Ako je korisnik izabrao gotovinu, trenutni iznos gotovine se uvećava na ekranu svaki put kad ubaci gotovinu. Kupnja bankovnom karticom podrazumijeva uspostavu veze s bazom podataka banke te nakon uspješne uspostave veze, aparat od baze zahtijeva provedbu plaćanja. Ako veza ne može biti uspostavljena ili na računu nema dovoljno sredstava, na zaslonu se ispisuje odgovarajuća poruka, prekida se kupnja i vraća se kartica.

Odabirom potvrde kupnje provodi se kupnja. Ako je kupnja uspješno provedena, klijentu se ispisuje karta sa svim informacijama o vlaku, vraća višak gotovine ili bankovna kartica te se kupnja evidentira u bazi podataka željeznice. U slučaju plaćanja gotovinom, ako ubačeni iznos gotovine nije dosegao potrebni iznos, klijentu se treba ispisati odgovarajuća poruka na zaslonu te vratiti ubačenu gotovinu.

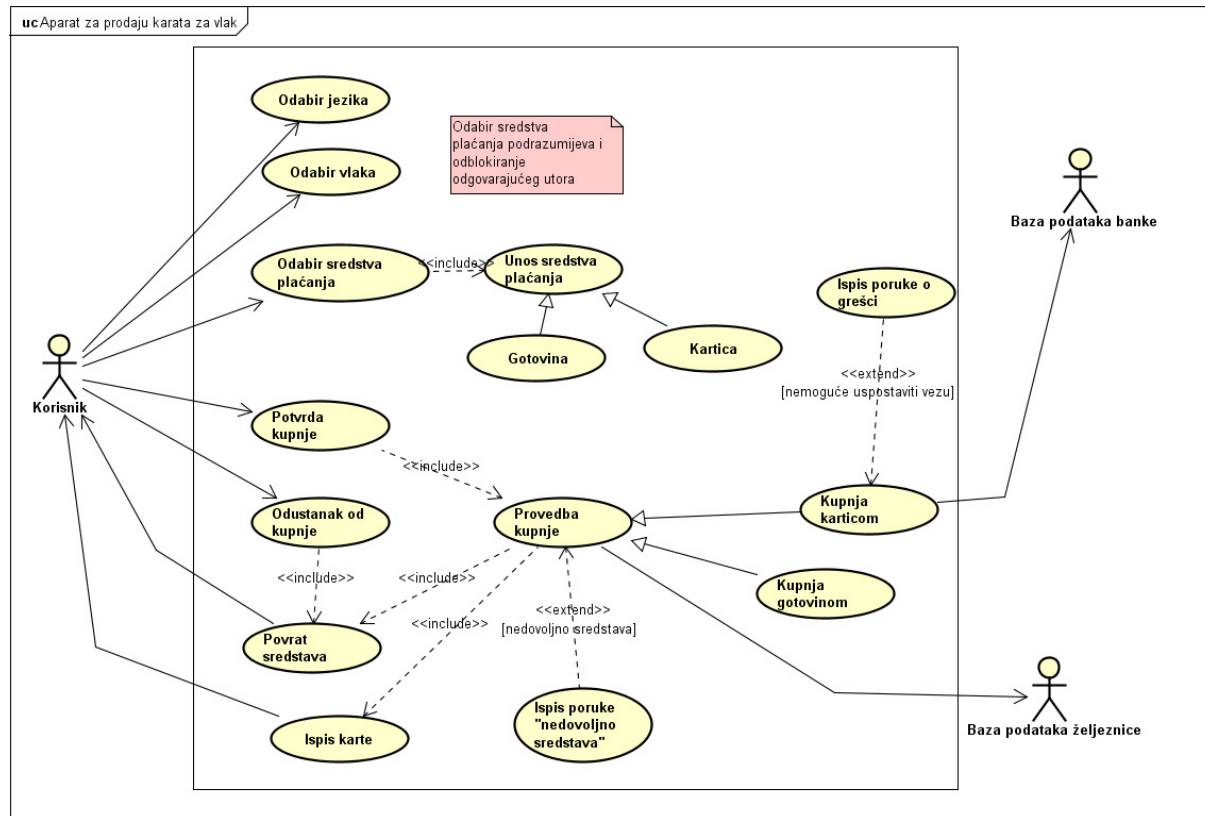
Korisnik može odustati od kupnje u svakom trenutku prije potvrde kupnje pritiskom odgovarajućeg gumba, pri čemu mu se vraćaju sva ubačena sredstva. Po završetku interakcije s korisnikom, svi utori se zatvaraju i zaslon se vraća u početno stanje.

23. (4 boda) Dijagramom obrazaca uporabe modelirajte sustav za prodaju željezničkih karata.

24. (4 boda) Dijagramom stanja modelirajte prikaz zaslona osjetljivog na dodir prilikom izbora i kupnje karte **gotovinom**. Napomena: pazite na elemente prijelaza između stanja.

25. (4 boda) Dijagramom aktivnosti modelirajte aktivnosti **nakon** potvrđivanja kupnje uporabom **bankovne kartice**. Moguće pogreške pri evidentiranju kupnje u bazi podataka željeznice nije potrebno modelirati. Napomena: plivačke staze obavezno koristite pri modeliranju.

Rješenje (23.):

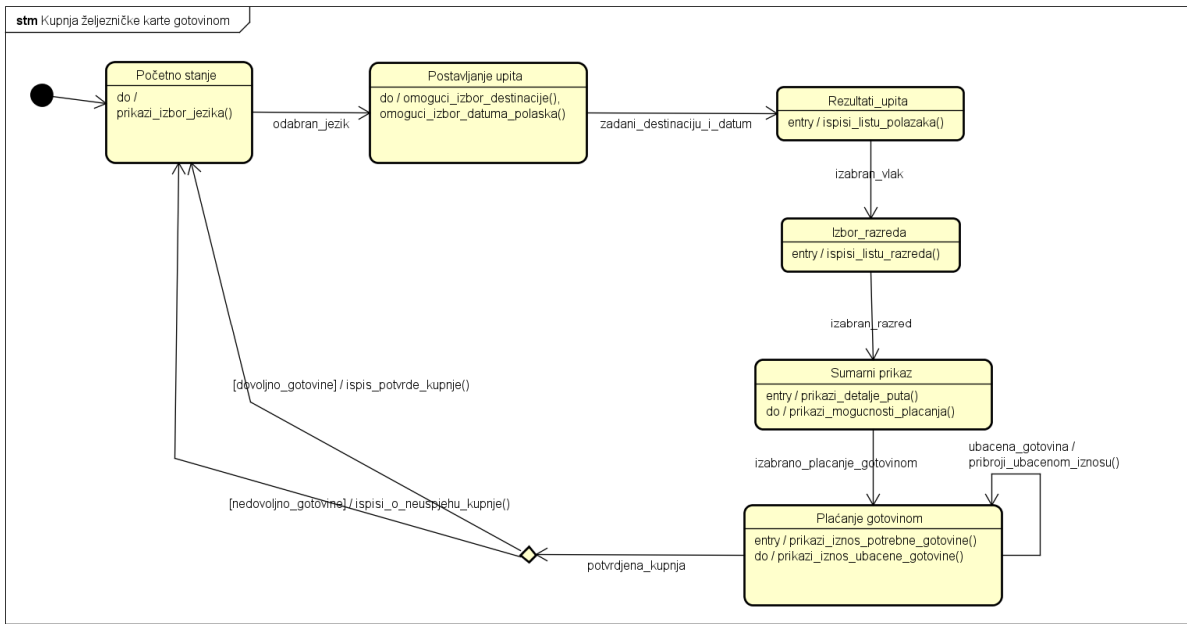


Kriteriji bodovanja:

Umjesto UCova „Odabir sredstva plaćanja“ i „Unos sredstva plaćanja“ priznaje se i jedinstveni UC.

0.5 bodova za prepoznavanje aktora. 2 boda za prepoznavanje UCova i ispravne veze s aktorima: „Odabir jezika“, „Unos sredstva plaćanja“, „Odabir vlaka“, „Potvrda kupnje“, „Odustanak od kupnje“, „Povrat sredstava“, „Ispis karte“, „Provedba kupnje“. 1,5 boda za točne veze među UC-ovima (include, extend, generalizacija – svaka po pola boda).

Rješenje (24.):



powered by Astah

Kriteriji bodovanja:

Navod glavnine potrebnih stanja:

- min. 4 stanja: 0,5 bod
- min. 5 stanja: 1 bod

Ispravna notacija stanja (naziv, aktivnosti entry/do): 1 bod

Prijelazi vođeni događajima: 1 bod

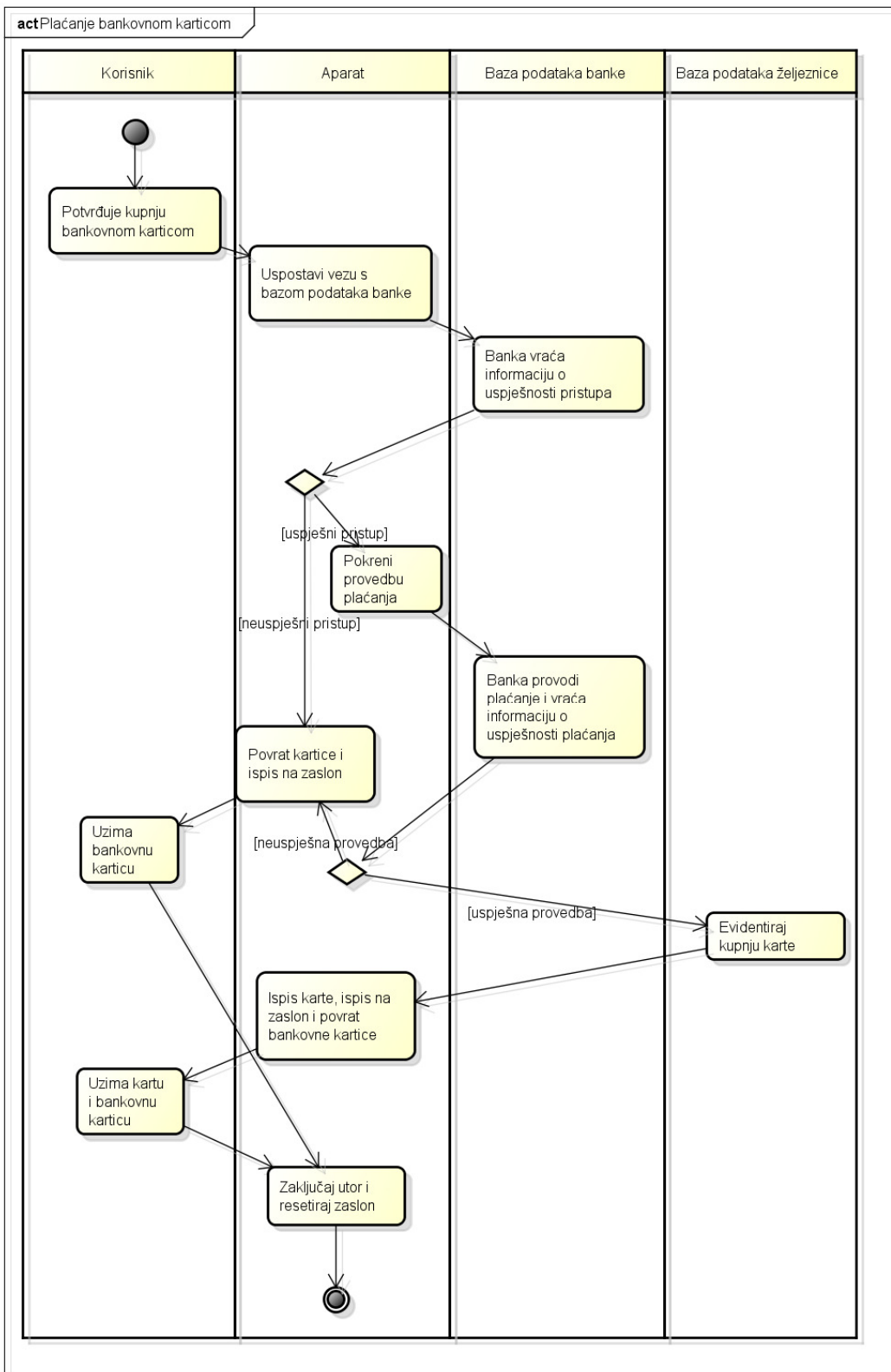
Ispravna notacija i semantika grananja i uvjeta: 1 bod

Nema početnog stanja: -0,5 boda

Nepostojanje prijelaza ubacivanja gotovine u stanju „Plaćanje gotovinom”: -1 bod

Bilo koji dodatni detalj koji bitno narušava rješenje, tj. nepoznavanje temeljne notacije i semantike dijagrama stanja: -0,5 boda

Rješenje (25.):



powered by Astah

Kriteriji bodovanja:

Ispravno organizirane plivačke staze s ispravnim nacrtanim elementima dijagrama aktivnosti - 1 bod

Provjera uspostave veze - 0.5 boda

Provjera provedbe plaćanja - 0.5 boda

Baza podataka željeznice + evidentiranje kupnje karte u bazi željeznice - 0.5 boda

Početno stanje + uvijek na kraju dolazak u završno stanje - 0.5 boda

Zaključavanje utora i resetiranje zaslona - 0.5 boda

Ispis karte, ispis na zaslon i povrat bankovne kartice - 0.5 boda