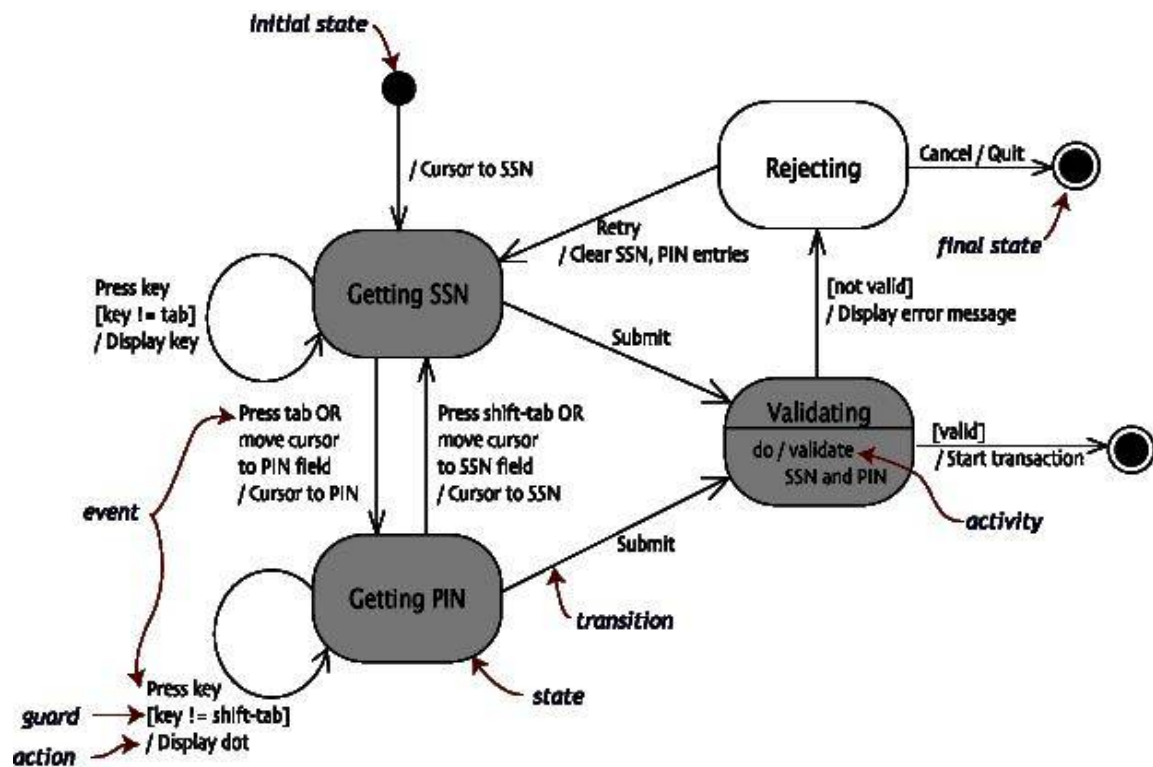


OPP - Samoprovjera znanja 3

Na pitanja odgovorio Telefunken.

1. UML dijagrami koji se primjenjuju za opis modela toka upravljanja (engl. *control flow*) ili toka podataka su:
 - a. Sekvencijski dijagrami
 - b. Dijagrami stanja
 - c. Komunikacijski dijagrami
 - d. Dijagrami aktivnosti**
2. UML dijagrami koji se temelje na prikazu fizičkih modularnih i zamjenjivih jedinki kojima se može pristupati samo kroz sučelja su:
 - a. Dijagrami komponenti**
 - b. Dijagrami razreda
 - c. Dijagrami paketa
 - d. Dijagrami objekata
3. Za objekt u UML dijagramu objekata vrijede tvrdnje:
 - a. Simbol objekta je pravokutnik s dva pretinca.**
 - b. Objekti imaju definiciju metoda.
 - c. Simbol objekta je pravokutnik s tri pretinca.
 - d. Objekti imaju vrijednosti atributa.**
 - e. Objekti nemaju definiciju atributa.
4. Moguće razine vidljivosti unutar UML dijagrama paketa su:
 - a. javno (engl. *public*)**
 - b. paket (engl. *package*)
 - c. privatno (engl. *private*)**
 - d. zaštićeno (engl. *protected*)**
5. Topologiju sustava te odnose njegovih sklopovskih i programskih dijelova prikazuju UML dijagrami:
 - a. komponenti
 - b. komunikacije
 - c. paketa
 - d. objekata
 - e. razmještaja**

6. Prikazani UML diagram je:



- dijagram razreda
- dijagram obrazaca uporabe
- sekvencijski diagram
- dijagram stanja**
- dijagram objekata
- dijagram aktivnosti

7. U skupinu UML dijagrama međudjelovanja (engl. interaction diagrams) spadaju:

- komunikacijski diagram**
- sekvencijski diagram**
- dijagram obrazaca uporabe
- dijagram stanja
- dijagram aktivnosti

8. Klijent-poslužitelj arhitektura ima nekoliko dobrih strana. Koje od slijedećih NISU dobre strane ove arhitekture (koje izjave nisu točne)?

a. konzistencija podataka

b. raspodjela posla na više računala

c. jednostavno upravljanje sigurnošću podataka

d. različiti poslužitelji

e. lakoća programiranja takvih sustava (ne samo za profesionalce)

f. mogućnost odvojenog oblikovanja klijenta i poslužitelja

g. različiti klijenti

h. jednostavnost sinkronizacije

9. Za razred `ConnectionToClient` vrijedi tvrdnja da je za komunikaciju sa svim klijentima potrebna jedna jedina instanca tog razreda.

Točno

Netočno

10. U SOA (*Software as a Service*) arhitekturi aplikacija se organizira kao kolekcija **usluga** koje međusobno komuniciraju uporabom dobro definiranih javnih **sučelja**.

11. Za kombinacijsko ispitivanje vrijede tvrdnje:

a. na broj ispitnih slučajeva ne utječe broj grananja u programskom kodu

b. broj ispitnih slučajeva ne ovisi o strukturi podataka

c. Pogodno za ispitivanje međutjecaja varijabli

d. Otkriva nedokumentirane odzive

12. Uporabom se programska potpora kviri jer:

a. postaje nekompatibilno s okolinom

b. sklopovlje na kojem se izvodi podložno je starenju

c. pogreške se uočavaju tek kod intenzivne i dugotrajne uporabe

d. rezervni dijelovi zastarijevaju

e. krajnji korisnici se mijenju

f. je izložena teškim radnim uvjetima

13. Programskom potporom nazivamo:

- a. razvojnu okolinu
- b. skup povezanih programa
- c. program, pripadajuću dokumentaciju, korisničke upute**
- d. program i sklopovsku dokumentaciju
- e. skup naredbi i procedura

14. Pri ispitivanju programske potpore upotrebljavamo termin ispitni slučaj. Odredi ispravne tvrdnje.

- a. bilježimo status/uspješnost provođenja ispitivanja
- b. prije provođenja ispitivanja moraju biti definirani ulazni podaci i očekivani izlaz
- c. nakon provođenja ispitni slučajevi se ne ponavljaju
- d. na očekivane rezultate ispitnih slučajeva utječe opis stanja prije izvođenja
- e. ispitne slučajeve jedinstveno obilježavamo

15. Za metriku pokrivanja stupnja gotovosti ispitivanja upotrebljavamo:

Odaberite jedan ili više odgovora:

- a. Pokrivenost putova**
- b. Vrijeme izvođenja
- c. Pokrivenost grana**
- d. Broj ispitnih slučajeva
- e. Postotak programskih elemenata koji su izvedeni**
- f. Pokrivenost linija koda**

16. Koji tipovi UML dijagrama prikazuju i jasno razlikuju aktivnosti (engl. *activity*) i akcije (engl. *action*) sudionika?

Odaberite jedan ili više odgovora:

- a. Komunikacijski dijagram (engl. communication diagram)
- b. Sekvencijski dijagram (engl. sequence diagram)
- c. Dijagram obrazaca uporabe (engl. use-case diagram)
- d. Dijagram stanja (engl. statechart diagram)**
- e. Dijagram aktivnosti (engl. activity diagram)**

17.Artefakti dijagrama komponenata mogu biti:

Odaberite jedan ili više odgovora:

- a. sklopovske komponente
- b. izvorni kodovi**
- c. poslužitelji
- d. izvršne datoteke**
- e. statičke i dinamičke biblioteke**

18.Prednosti uporabe arhitektura s n razina su:

Odaberite jedan ili više odgovora:

- a. Lako je odrediti optimalno preslikavanje odgovornosti na razine.
- b. Olakšana potpora zahtjevima povećanja i poboljšanja sustava.
- c. Postizanje poboljšanja performansi izričitim podržavanjem definicije n -razinske arhitekture pri oblikovanju.
- d. Ograničeni utjecaj promjena jedne razine na cijeli sustav.**
- e. Pojednostavljeno oblikovanje temeljem viših razina apstrakcije.**
- f. Dobra potpora ponovnom korištenju (engl. *reuse*), prenosivost i sl.**

19.Koje metode korisnik OCSF radne klupe mora obavezno implementirati za klijentsku stranu?

Odaberite jedan ili više odgovora:

- a. closeConnection()
- b. handleMessageFromServer()**
- c. sendToServer()
- d. isConnected()
- e. openConnection()**



20. Za tankog klijenta (*thin client*) vrijedi tvrdnja da se izvršni kod preuzima preko mreže.

Odaberite jedan odgovor:

Točno

Netočno

21. UML dijagrami koji se temelje na pojedincima nekog dijagrama razreda se nazivaju:

Odaberite jedan odgovor:

a. Dijagrami komponenti (engl. component diagram)

b. Dijagrami objekata (engl. object diagram)

c. Dijagrami paketa (engl. package diagram)

d. Dijagrami razreda (engl. class diagram)

22. Koje su osnovne komponente svakog komunikacijskog protokola?

Odaberite jedan odgovor:

a. Dva jezika i pravila konverzacije

b. Jedan jezik i pravila konverzacije

c. Pravila konverzacije

23. Od navedenih koji razred(i) pripada(ju) poslužiteljskoj strani?

Odaberite jedan ili više odgovora:

a. AbstractServer

b. AbstractClient

c. ConnectionToClient

24. UML dijagram koji prikazuje uređen redoslijed razmjene poruka između sudionika bez izrazitog prikaza preciznijih vremenskih odnosa je:

Odaberite jedan odgovor:

a. Komunikacijski dijagram (engl. communication diagram)

b. Objektni dijagram (engl. object diagram)

c. Dijagram aktivnosti (engl. activity diagram)

d. Sekvencijski dijagram (engl. sequence diagram)

25. Kada želimo napraviti liniju proizvoda koja će se imati zajedničku osnovnu funkcionalnost, a razlikovati se u pojedinim mogućnostima koristiti ćemo mehanizam koji (u OO arhitekturi) nazivamo:

Odaberite jedan odgovor:

- a. Statični UML dijagrami
- b. Scenarij
- c. Radni okvir**
- d. Predefinirana arhitektura sustava

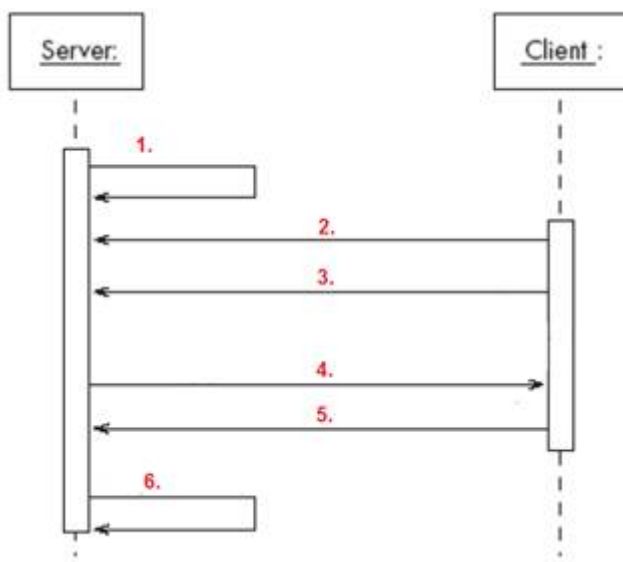
26. Koja od navedenih NISU pravila uporabe OCSF-a (*moguće više točnih odgovora*)?

Odaberite jedan ili više odgovora:

- a. Redefinirati metode u apstraktnim razredima.**
- b. Kreirati podrazrede.
- c. Modificirati apstraktne razrede tako da se dobije puna funkcionalnost.**
- d. Konkretizirati metode u podrazredima.

27. Na kojoj se web tehnologiji baziraju standardni komunikacijski protokoli u SOA-i? **XML**

28. Na donjoj slici prikazan je jednostavan sekvencijski dijagram koji prikazuje komunikaciju u arhitekturi klijent - poslužitelj. Povežite redni broj aktivnosti prikazan na strelicama s nekim od ponuđenih opisa aktivnosti.



- Aktivnost 1. **Početak rada i slusanje**
- Aktivnost 2. **Spajanje**
- Aktivnost 3. **Slanje poruke (zahtjeva)**
- Aktivnost 4. **Slanje odgovora**
- Aktivnost 5. **Odspajanje**
- Aktivnost 6. **Prestanak slusanja**