

581359 Ohjelmistoprosessit ja ohjelmistojen laatu

Mahdolliset koekysymykset – näistä arvotaan kuhunkin kokeeseen tulevat kysymykset. Kysymysten pistemäärä on 4 tai 6 pistettä (ilmoitetaan kysymyspaperissa). Vastauksen odotettu pituus on noin ½ - 1 sivua. Vastauksen pitää koostua kokonaisista lauseista, jotka muodostavat loogisen ja johdonmukaisen kokonaisuuden (pelkkä bullettlista ei riitä).

Kysymykset luetellaan alla aihealueittain

Laadun määritelmä ja laatumalli

1. Miten ISO/IEC 25010 standardissa määritellään *ohjelmiston laatu*? Anna määritelmä ja selitä se lyhyesti.
2. Selitä ISO/IEC 25010 *laatumallin* rakenteen perusidea ja luettele mallin päätason laatupiirteet.
3. Varaudu selittämään, mitä jokin seuraavista *tuotelaadun piirteistä* tarkoittaa ja luettele ja selitä lyhyesti sen alipiirteet: *Toiminnallinen sopivuus, Tehokkuus, Ylläpidettävyys*
4. Selitä, miten ISO/IEC 25010 *laatumallia* on tarkoitus käyttää ohjelmiston laadun määrittämiseen.
5. Mitkä ovat ISO/IEC 25010 tunnistamat *käyttäjäryhmät*; pohdi ja selosta, miten jokin itse valitsemasi tuotelaadun piirre voi vaikuttaa jonkin käyttäjäryhmän kokemaan laatuun.

Laadun vaikutusketju

6. Selitä, mitä *laadun vaikutusketjulla* tarkoitetaan ja miten se toimii.
7. Anna (kuvitteellinen tai todellinen) konkreettinen esimerkki siitä, miten ohjelmiston kehitysprosessi (menetelmät, resurssit, työskentelytavat jne.) voi vaikuttaa loppukäyttäjän kokemaan laatuun positiivisesti tai negatiivisesti.

Ohjelmistoviat ja laadun taloudelliset vaikutukset

8. Mitä tarkoittavat *virhe*, *vika*, ja *häiriö*? Miten ne liittyvät toisiinsa?
9. Ohjelmistoprojekteissa on mahdollista tehdä monen tyyppisiä virheitä. Anna jokaisesta näistä projektin osa-alueista kaksi esimerkkiä mahdollisista virheistä (yhteensä neljä/kuusi esimerkkiä)
 - Vaatimusten määrittely
 - Asiakkaan ja kehittäjän välinen kommunikaatio,
 - Ohjelmistosuunnittelu ja koodaus
10. Mitä taloudellisia vaikutuksia huonolla ja hyvällä laadulla on ohjelmiston kehitykseen ja ylläpitoon (kaksi/kolme vaikutusta kummastakin)?

Ohjelmistojen mittaaminen

11. Määrittele *mitta* ja *mittaus*.
12. Mitä tarkoittavat *suora mitta*, *johdettu mitta* ja *sijaismitta*?
13. Kaner ja Bond määrittelevät mittauksen (measurement) seuraavasti: "*Measurement is the empirical, objective assignment of numbers, according to a rule derived from a model or theory, to attributes of objects or events with the intent of describing them*". Miksi alleviivattu osa määritelmästä on tärkeä? Anna esimerkki (huonosta) mittauksesta, joka ei täytä tätä vaatimusta.
14. Selitä, mikä on *laatumittari* ja mihin tarkoituksiin laatumittareita käytetään.
15. Selitä *Goal-Question-Metric –mallin* perusidea ja kerro, miten sitä käytetään ohjelmiston laatumittareiden valitsemiseen.
16. Mitä *riskejä* liittyy ohjelmistomittareiden käyttöön *tuotteen laadun* ja *työntekijöiden suoriutumisen* arvioimiseen?
17. Mitä etuja ja haittoja on *koodirivien määrän* (LOC, KLOC) käytöllä *ohjelmiston koon* mittarina?
18. Mitä etuja ja haittoja on *toimintopisteiden* (FP) käytöllä *ohjelmiston koon* mittarina?
19. Anna esimerkki jostain mittarista, jolla voidaan mitata ohjelmiston *rakennetta*. Mitä mittarin tuottamat arvot kertovat ohjelmiston laadusta? Miten arvoja pitää tulkita?

Projektinhallinta

20. Piirrä ja selitä *projektinhallintakolmio* (Project Management Triangle) ja kerro, mitä sen muoto kertoo projektin kustannuksista ja tuottavuudesta. Mitkä ovat *kaksi tärkeintä mittaria*, jotka voidaan kolmiosta laskea?
21. Selosta, miten kehitystiimin *kyvyt* (developer team capability) ja *tuotteen omistajan kyvyt* (product owner capability) vaikuttavat projektinhallintakolmion muotoon. Mitkä ovat kummankin ryhmän tärkeimmät taidot (eli mitä pitää osata)?

Kehitysprosessit

22. Selitä *iteratiivisen ja lisäävän ohjelmistojen kehitystavan* (iterative and incremental development, eli IID) perusideat ja selitä, mitä hyötyjä ne tuovat.
23. Vertaile *riskilähtöistä* ja *asiakaslähtöistä* tapaa päättää (IID-kehitystavan) iteraation tavoitteista (eli mitä tuotetaan iteraation aikana). Mitä näillä tavoilla tarkoitetaan, ja minkälainen vaikutus niillä voi olla tuotteen laatuun?
24. Miten ketterän XP (eXtreme Programming) kehitysmenetelmän piirteet tukevat laadukkaan ohjelmiston tuottamista?
25. Mikä on *refaktoroinnin* rooli ketterässä kehittämisessä (esim XP)? Mitä riskejä refaktorointiin liittyy?
26. Selitä, mikä on Cumulative Flow Diagram –kaavio (CFD) ja mihin sitä käytetään ketterässä ohjelmistokehityksessä. Anna jokin esimerkki mahdollisesta ongelmasta, joka voidaan kaavion avulla havaita.