

Harjoituskoe – Vastaukset

Harjoituskoe C

Käännösversio 1.0

Perustuu alkuperäiseen koeversioon 1.2

ISTQB® Sertifioitu testaaja Perustaso

Yhteensopiva sertifikaattisisällön version 4.0 kanssa

International Software Testing Qualifications Board



Tekijänoikeushuomautus

Tekijänoikeus © International Software Testing Qualifications Board (jäljempänä ISTQB®). ISTQB® on International Software Testing Qualifications Boardin rekisteröity tavaramerkki.

Kaikki oikeudet pidätetään.

Kirjoittajat siirtävät tekijänoikeudet täten ISTQB®:lle. Kirjoittajat (nykyisinä tekijänoikeuksien haltijoina) ja ISTQB® (tulevana tekijänoikeuden haltijana) ovat hyväksyneet seuraavat käyttöehdot:

Tästä asiakirjasta saa kopioida ei-kaupalliseen käyttöön tarkoitettuja otteita, jos lähde mainitaan

Kuka tahansa akkreditoitu koulutustarjoaja voi käyttää tätä harjoituskoea koulutuksessaan, jos tekijät ja ISTQB® mainitaan lähteinä ja harjoituskokeen tekijänoikeuksien omistajina, ja edellyttäen, että tällaisen kurssin mainonnassa saa mainita sertifikaattisisällön vasta sen jälkeen, kun ISTQB®:n tunnustamalta jäsenyhdistykseltä on saatu virallinen hyväksyntä koulutusmateriaalille.

Kuka tahansa yksilö tai ryhmä voi käyttää tätä harjoituskoea artikkeleiden ja kirjojen pohjana, jos kirjoittajat ja ISTQB® mainitaan lähteinä ja harjoituskokeen tekijänoikeuksien omistajiksi.

Tämän harjoituskokeen muu käyttö on kielletty ilman ISTQB®:n kirjallista hyväksyntää.

Jokainen ISTQB®:n tunnustama jäsenyhdistys voi kääntää tämän harjoituskokeen edellyttäen, että se sisällyttää yllä olevan tekijänoikeusilmoituksen harjoituskokeen käännettyyn versioon.

Vastuu asiakirjasta

Tästä asiakirjasta vastaa ISTQB® Examination Working Group.

Tätä asiakirjaa ylläpitää ISTQB® Syllabus Working Groupista ja Exam Working Groupista koostuva ydintyöryhmä.

Kiitokset

ISTQB®:n ydintiimi: Stuart Reid and Adam Roman

Ydintiimi kiittää koetyöryhmän arviointiryhmää, opetussuunnitelmatyöryhmää ja jäsenhallituksia ehdotuksista ja panoksesta.

Versiohistoria

Versio	Alkup. versio	Päivämäärä	Huomautukset
1.0	1.2	15.3.2024	Ensimmäinen käännösversio

Vastaukset

Kysymys numero (#)	Oikea vastaus	LO	K-taso	Pisteet
1	b	FL-1.1.1	K1	1
2	c	FL-1.1.2	K2	1
3	b	FL-1.3.1	K2	1
4	b, e	FL-1.4.1	K2	1
5	a	FL-1.4.3	K2	1
6	c	FL-1.4.5	K2	1
7	b	FL-1.5.2	K1	1
8	a	FL-1.5.3	K2	1
9	d	FL-2.1.2	K1	1
10	d	FL-2.1.3	K1	1
11	b	FL-2.1.5	K2	1
12	c	FL-2.1.6	K2	1
13	d	FL-2.2.1	K2	1
14	b	FL-2.2.3	K2	1
15	d	FL-3.1.3	K2	1
16	a	FL-3.2.1	K1	1
17	b	FL-3.2.4	K2	1
18	b	FL-3.2.5	K1	1
19	c	FL-4.1.1	K2	1
20	c	FL-4.2.1	K3	1

Kysymys numero (#)	Oikea vastaus	LO	K-taso	Pisteet
21	d	FL-4.2.2	K3	1
22	d	FL-4.2.3	K3	1
23	a	FL-4.2.4	K3	1
24	c	FL-4.3.2	K2	1
25	a	FL-4.3.3	K2	1
26	b	FL-4.4.1	K2	1
27	d	FL-4.4.3	K2	1
28	b	FL-4.5.2	K2	1
29	d	FL-4.5.3	K3	1
30	a	FL-5.1.1	K2	1
31	c	FL-5.1.4	K3	1
32	a	FL-5.1.5	K3	1
33	b	FL-5.1.6	K1	1
34	d	FL-5.1.7	K2	1
35	c	FL-5.2.3	K2	1
36	b	FL-5.3.2	K2	1
37	d	FL-5.4.1	K2	1
38	b	FL-5.5.1	K3	1
39	d	FL-6.1.1	K2	1
40	d	FL-6.2.1	K1	1

Vastaukset

Kysymys numero (#)	Oikea vastaus	Selitys/perustelut	Oppimistavoite (LO)	K-taso	Pistemäärä
1	b	<p>a) Väärin. Dokumentoitujen vaatimusten täyttymisen kelpuutus on väärin, koska kelpuutus liittyy käyttäjien vaatimusten ja odotusten täyttymiseen, kun taas todentaminen koskee määritettyjen vaatimusten täyttämistä, joten tämä olisi oikein, jos sana "kelpuutus" korvattaisiin sanalla "todentaminen".</p> <p>b) Oikein. Häiriöiden aiheuttaminen ja vikojen tunnistaminen on luultavasti dynaamisen testauksen yleisin tavoite</p> <p>c) Väärin. Virheiden aiheuttaminen ja juurisyiden tunnistaminen on väärin, koska testaajat eivät aiheuta virheitä, he yrittävät aiheuttaa häiriöitä. Virheet ovat tyypillisesti toteuttajien tekemiä, ja ne johtavat vikoihin, joita testaajat yrittävät tunnistaa joko suoraan staattisen testauksen avulla tai epäsuorasti dynaamisen testauksen avulla. Juurisyiden tunnistaminen on hyödyllistä, mutta se on osa vikojenjäljitystä, joka on testauksen ulkopuolinen toiminto</p> <p>d) Väärin. Sen todentaminen, että testauksen kohde täyttää käyttäjien odotukset, on väärin, koska todentaminen koskee tiettyjen (dokumentoitujen) vaatimusten täyttymisen tarkistamista, kun taas kelpuutus koskee käyttäjien vaatimusten ja odotusten täyttämistä, joten tämä olisi oikein, jos sana "todentaminen" korvataan sanalla "kelpuutus".</p>	FL-1.1.1	K1	1

Kysymys numero (#)	Oikea vastaus	Selitys/perustelut	Oppimistavoite (LO)	K-taso	Pistemäärä
2	c	<p>a) Väärin. Dynaaminen testaus aiheuttaa häiriöitä (joiden pohjalta viat voidaan sitten paikantaa ja korjata). Vikojen jäljitys liittyy kuitenkin vikojen paikantamiseen ja korjaamiseen. Siksi vikojen jäljitys ei korjaa häiriöitä</p> <p>b) Väärin. Sekä testaus että vikojen jäljitys myötävaikuttavat testauksen kohteen laadun parantamiseen, joten molempia pitäisi todellakin pitää hyvänä asiana. Vikojen jäljitystä pidetään yleensä positiivisena toimintana, koska se korjaa jotain. Dynaamiseen testaukseen kuuluu testauksen kohteen epäonnistumisen tahallinen aiheuttaminen, minkä vuoksi jotkut pitävät sitä negatiivisena toimintana, mutta se on hyvin kapea näkemys (eikä suinkaan testaa- jien tyypillinen näkemys). Sekä positiiviset että negatiiviset testitapaukset ovat mahdollisia. Positiiviset testitapaukset tarkistavat, että testauksen kohde tekee oikein sen, mitä sen pitäisi tehdä, kun taas negatiivinen testaus tarkistaa, että testauksen kohde ei tee sitä, mitä sen ei pitäisi tehdä</p> <p>c) Oikein. Testaus osoittaa, että vikoja esiintyy, joko suoraan havaitsemalla vika katselmoinneissa (tai staattisen analyysin työkalulla) tai epäsuorasti aiheuttamalla häiriö dynaamisessa testauksessa. Vikojen jäljitys on testauksesta erillinen toimenpide (jonka yleensä suorittavat toteuttajat), ja se koskee vikojen paikantamista (vain dynaamisessa testauksessa) ja vikojen korjaamista.</p> <p>d) Väärin. Vikojen syyt ovat tyypillisesti inhimillisiä virheitä. Testaus löytää vikoja joko suoraan staattisen testauksen avulla tai epäsuorasti aiheuttamalla häiriöitä dynaamisessa testauksessa, ja vikojen jäljitys korjaa viat. Testaus ei siis löydä vikojen syytä ja vikojen jäljitys ei korjaa vikojen syytä</p>	FL-1.1.2	K2	1

Kysymys numero (#)	Oikea vastaus	Selitys/perustelut	Oppimistavoite (LO)	K-taso	Pistemäärä
3	b	<p>Vikojen puuttumisen harhaluulo liittyy ajatukseen, että vaatimustenmukaisuuden varmistaminen (eli toteutusvikojen puuttumisen varmistaminen) ei takaa käyttäjien tyytyväisyyttä järjestelmään. Tämän huomioon ottamiseksi on myös tarpeen kelpuuttaa, että järjestelmä vastaa käyttäjien tarpeita ja odotuksia, täyttää liiketoiminnan tavoitteet ja ylittää kilpailevat järjestelmät.</p> <p>a) Väärin. "Testaus osoittaa vikojen olemassaolon, ei niiden puuttumista" -periaate selittää, että vaikka testaus voi osoittaa vikojen olemassaolon testauksen kohteessa, ei ole mahdollista osoittaa, että vikoja ei ole, ja siten taata kohteen oikeellisuutta. Siksi sen selittäminen, että testauksella ei ole mahdollista osoittaa vikojen puuttumista, liittyy osittain siihen periaatteen, ei vikojen puuttumisen harhaluuloon.</p> <p>b) Oikein. Tukemalla loppukäyttäjää hyväksymistestauksen tekemisessä pitäisi olla mahdollista varmistaa, että järjestelmä vastaa käyttäjien tarpeita ja odotuksia</p> <p>c) Väärin. Ei ole mahdollista varmistaa, että toimitetussa järjestelmässä ei ole toteutusvikoja, koska "testaus osoittaa vikojen olemassaolon, ei niiden puuttumista" -periaate selittää, että vaikka testaus voi osoittaa vikojen olemassaolon testauksen kohteessa, ei ole mahdollista osoittaa, että vikoja ei ole, ja siten taata kohteen oikeellisuutta.</p> <p>d) Väärin. Sellaisten testien muokkaaminen, jotka eivät aiheuta häiriöitä, sen varmistamiseksi, että vain vähän vikoja jää jäljelle, on yksi tapa käsitellä "testit menettävät tehonsa" -periaatetta. Tämä periaate liittyy ajatukseen, että identtisten testien toistaminen muuttumattomalla koodilla ei todennäköisesti paljasta uusia vikoja, ja siksi testien muuttaminen voi olla välttämätöntä. Tämä ei varmista sitä, että järjestelmä vastaa käyttäjien tarpeita ja odotuksia</p>	FL-1.3.1	K2	1

Kysymys numero (#)	Oikea vastaus	Selitys/perustelut	Oppimistavoite (LO)	K-taso	Pistemäärä
4	b, e	<p>Tarkastellaan testianalyysin kuvausta: Testaamista vaativien ominaisuuksien tunnistamiseksi testauksen pohjamateriaali analysoidaan ja sen pohjalta määritellään testattavat tilanteet, jotka sitten priorisoidaan niihin liittyvien riskien kanssa. Testattavien tilanteiden systemaattiseen tunnistamiseen kattavuuskohteiksi liittyy usein testaustekniikoiden käyttö sekä testianalyysin aikana että osana testien suunnittelutoimenpiteitä.</p> <p>Yllä olevasta kuvauksesta voidaan nähdä, että testaustekniikoita käytetään usein testianalyysissä ja testien suunnittelussa. Raja-arvoanalyysi ja ekvivalenssiositus ovat testaustekniikoita.</p> <p>a) Väärin. Testien valmisteluun ei todennäköisesti liity testaustekniikoiden käyttöä, koska siinä keskitytään enimmäkseen testitapausten kokoamiseen testiprocedureiksi, kun taas testaustekniikoilla luodaan testitapauksia</p> <p>b) Oikein. Testien suunnitteluun liittyy todennäköisesti testaustekniikoiden käyttö testitapausten luomiseksi testattavista tilanteiden ja kattavuuskohteiden pohjalta.</p> <p>c) Väärin. Testien suorittamiseen ei todennäköisesti liity testaustekniikoiden käyttöä, koska siinä keskitytään enimmäkseen testiprocedureiden (ja siten testitapausten) suorittamiseen, kun taas testaustekniikoilla luodaan testitapauksia</p> <p>d) Väärin. Testauksen seurantaan ei todennäköisesti liity testaustekniikoiden käyttöä. Testauksen seurannassa keskitytään enimmäkseen jatkuviin tarkistuksiin suunnitelman noudattamisen varmistamiseksi, kun taas testaustekniikoilla luodaan testitapauksia</p> <p>e) Oikein. Testianalyysiin liittyy todennäköisesti testaustekniikoiden käyttö testattavien tilanteiden tunnistamiseksi</p>	FL-1.4.1	K2	1

Kysymys numero (#)	Oikea vastaus	Selitys/perustelut	Oppimistavoite (LO)	K-taso	Pistemäärä
5	a	<p>Tarkastellaan listattuja testaustoimenpiteitä ja niiden tuottamia materiaaleja:</p> <ul style="list-style-type: none">A. Testianalyysi - priorisoidut testattavat tilanteet (4) (esim. hyväksymiskriteerit) ja testauksen pohjamateriaalista tunnistettujen vikojen vika-raportitB. Testien suunnittelu – priorisoidut testitapaukset, testausohjeet, kattavuuskohteet (1), testiaineiston ja testiympäristön vaatimuksetC. Testien valmistelu – testiproceduurit, automatisoidut testiskriptit, testi-joukot, testiaineisto, testien suoritusajataulu (3) ja testiympäristön elementit, kuten tyngät, ajurit, simulaattorit ja palvelujen virtualisoinnitD. Testauksen päättäminen – testauksen loppuraportti, dokumentoidut opitut kokemukset, parannustoimenpiteet ja muutospyyntöt (2) (tuotteen kehitysjonon kohtina) <p>Näin ollen:</p> <ul style="list-style-type: none">a) Oikein. Oikeat vastaavuudet ovat: 1B, 2D, 3C, 4Ab) Väärinc) Väärind) Väärin	FL-1.4.3	K2	1

Kysymys numero (#)	Oikea vastaus	Selitys/perustelut	Oppimistavoite (LO)	K-taso	Pistemäärä
6	c	<p>a) Väärin. Vaikka on oikein sanoa, että ketterässä ohjelmistokehityksessä ketterä tiimi voi itse hoitaa osan testauksen hallinnan tehtävistä, testausrooli ei ole ensisijaisesti tiimin ulkopuolisen yksittäisen henkilön vastuulla. Sen sijaan testauksen suorittavat todennäköisemmin useat tiimin jäsenet noudattaen tiimiperustaista lähestymistapaa</p> <p>b) Väärin. Testauksen hallinnan rooliin sisältyy ensisijaisesti testauksen suunnitteluun, testauksen seurantaan ja hallintaan sekä testauksen päättämiseen liittyviä toimenpiteitä. Näin ollen, vaikka väite on osittain oikein, on väärin sanoa, että testausrooli on ensisijaisesti vastuussa testauksen seurannasta ja hallinnasta</p> <p>c) Oikein. Ketterässä ohjelmistokehityksessä osan testauksen hallinnan tehtävistä voi hoitaa ketterä tiimi itse. Toteutustiimin ulkopuoliset testauspäälliköt voivat kuitenkin hoitaa sellaiset testausmenpiteet, jotka koskevat useita organisaation tiimejä</p> <p>d) Väärin. Testauksen hallinnan rooli sisältää ensisijaisesti testauksen suunnitteluun, testauksen seurantaan ja hallintaan sekä testauksen päättämiseen liittyviä toimenpiteitä, kun taas testausrooli vastaa ensisijaisesti testauksen teknisistä näkökohdista, kuten testianalyysistä, testien suunnittelusta, testien valmistelusta ja testien suorituksesta. Testauksen hallinnan rooli ei siis normaalisti vastaa testianalyysistä ja testien suunnittelusta, vaikka on oikein sanoa, että testausrooli on ensisijaisesti vastuussa testien valmistelusta ja suorituksesta.</p>	FL-1.4.5	K2	1

Kysymys numero (#)	Oikea vastaus	Selitys/perustelut	Oppimistavoite (LO)	K-taso	Pistemäärä
7	b	<p>a) Väärin. Tiimiperustaisessa lähestymistavassa testaaajilla on tärkeä rooli, kun he jakavat testausosaamistaan tiimin kanssa ja ohjaavat tuotekehitystä. He tekevät yhteistyötä muiden tiimin jäsenten kanssa haluttujen laatutasojen saavuttamiseksi ja työskentelevät liiketoiminnan edustajien kanssa hyväksymistestien luomiseksi. Testaajat tekevät yhteistyötä myös toteuttajien kanssa optimaalisen testausstrategian ja automaation lähestymistapojen määrittämiseksi</p> <p>b) Oikein. Hyödyntämällä kunkin tiimin jäsenen monipuolisia taitoja tehokkaimmin, tiimiperustainen lähestymistapa myötävaikuttaa parempaan tiimidynamiikkaan, edistää tiivistä kommunikaatiota ja yhteistyötä sekä luo synergistisen vaikutuksen, joka hyödyttää koko projektia.</p> <p>c) Väärin. Tiimiperustaisen lähestymistavan avulla jokainen tiimin jäsen, jolla on tarvittavat taidot ja tiedot, voi suorittaa minkä tahansa tehtävän, joten tiimin asiantuntijajäsenet eivät ole tämän lähestymistavan etu.</p> <p>d) Väärin. Tiimiperustaista lähestymistapaa käyttävien tiimien optimaaliseen kokoon ei ole erityisiä ohjeita, eikä ole viitteitä siitä, että isommat tiimit olisivat parempia</p>	FL-1.5.2	K1	1

8	a	<p>a) Oikein. Testauksen riippumattomuuden ensisijainen hyöty on se, että testaajat tunnistavat todennäköisemmin erityyppisiä häiriöitä ja vikoja kuin toteuttajat heidän eri tyyppisten taustojensa ja teknisten näkökulmien sekä mahdollisten harhakäsitysten, kuten kognitiivisen harhan vuoksi. Testauksen riippumattomuuden suurin haittapuoli on kuitenkin se, että testaajat voivat joutua eristykseen toteutustiestä, mikä saattaa johtaa viestintäongelmiin, yhteistyön puutteeseen ja mahdollisiin vastakkainasettelutilanteisiin, joissa testaajia syytetään julkaisuprosessin viivästyksistä ja pullonkauloista.</p> <p>b) Väärin. Toteuttajan koodin tuntemus ei tarkoita sitä, että hän löytää siitä vain harvoin vikoja, vaan tuttuus tarkoittaa sitä, että hän pystyy löytämään tehokkaasti monia vikoja omasta koodistaan. Ja sen sijaan, että toteuttajilla ja testaajilla olisi yhteinen tausta, toteuttajien ja testaajien erilainen tausta mainitaan yleensä syynä siihen, että testaajat ja toteuttajat löytävät erilaisia vikoja.</p> <p>c) Väärin. Testausta voidaan tehdä eri riippumattomuuden tasoilla, lähtien toteuttajasta, joka ei ole riippumaton, organisaation ulkopuolisten testaajien erittäin korkeaan riippumattomuuteen. Useimmissa projekteissa käytetään useita riippumattomuuden tasoja, jolloin toteuttajat suorittavat yksikkö- ja komponentti-integraatiotestauksen, testaustiimi suorittaa järjestelmä- ja järjestelmäintegraatiotestauksen ja liiketoiminnan edustajat tekevät hyväksymistestauksen. Testaajat voivat siis kuulua toteutustiestiin eikä heidän tarvitse tulla organisaation ulkopuolelta. Sovellusalueen tuntemus muuttuu tapauskohtaisesti, eikä se ole riippuvainen riippumattomuuden tasosta.</p> <p>d) Väärin. Testausta voidaan tehdä eri riippumattomuuden tasoilla, lähtien toteuttajasta, joka ei ole riippumaton, organisaation ulkopuolisten testaajien erittäin korkeaan riippumattomuuteen, jolloin toteutustiestiin ulkopuoliset testaajat ovat yleensä riippumattomampia kuin testaajat tiimin sisällä. On kuitenkin enemmän syytä uskoa, että tiimin ulkopuoliset testaajat ovat todennäköisesti enemmän eristyksissä toteuttajista ja siksi heitä todennäköisemmin syytetään tuotteen julkaisun viivästyisestä.</p>	FL-1.5.3	K2	1
---	---	--	----------	----	---

Kysymys numero (#)	Oikea vastaus	Selitys/perustelut	Oppimistavoite (LO)	K-taso	Pistemäärä
9	d	<p>a) Väärin. Laadunvalvonta koskee kaikkia kehitystoimenpiteitä, mikä tarkoittaa, että jokaista ohjelmiston kehitystoimenpidettä vastaa vastaava testaustoimenpide. Tässä yritetään kuitenkin rinnastaa testautasot kehitystasoihin, ja vaikka tiedetään, mitä "testautasoiilla" tarkoitetaan, termistä "kehitystaso" ei ole yhteistä käsitystä.</p> <p>b) Väärin. Jokaisella ohjelmiston kehitystoimenpiteellä on vastaava testaustoimenpide; testauksen tavoitteet ovat kuitenkin aivan erilaiset. Testauksen tavoitteena voi olla esimerkiksi varmistaa, että testauksen kohde täyttää sopimuksessa asetetun vaatimuksen, jonka mukaan ennen toimitusta sille on suoritettava tietyntyyppinen testaus. Tässä tapauksessa ei ole mitään syytä olla vastaavaa kehitystavoitetta</p> <p>c) Väärin. Laadunvalvonta koskee kaikkia kehitystoimenpiteitä, mikä tarkoittaa, että jokaista ohjelmiston kehitystoimenpidettä vastaa vastaava testaustoimenpide. Sama symmetria ei kuitenkaan koske testausta ja käyttäjien toimintaa. Esimerkiksi joidenkin järjestelmien osalta on vaikea edes tunnistaa loppukäyttäjiä. Lisäksi jotkin testautustoimenpiteet kohdistuvat toteuttajiin (esim. ylläpidettävyyden testaus), eikä niissä ole käyttäjänäkökulmaa.</p> <p>d) Oikein. Laadunvalvonta koskee kaikkia kehitystoimenpiteitä, mikä tarkoittaa, että jokaista ohjelmiston kehitystoimenpidettä vastaa vastaava testaustoimenpide.</p>	FL-2.1.2	K1	1
10	d	<p>a) Väärin. Komponenttitestiohjattu kehitys ei ole oikea esimerkki testit ensin -ohjelmistokehitystavasta</p> <p>b) Väärin. Integraatiotestiohjattu kehitys ei ole oikea esimerkki testit ensin -ohjelmistokehitystavasta</p> <p>c) Väärin. Järjestelmätestiohjattu kehitys ei ole oikea esimerkki testit ensin -ohjelmistokehitystavasta</p> <p>d) Oikein. Hyväksymistestiohjattu kehitys (ATDD) on hyvin tunnettu esimerkki testit ensin -ohjelmistokehitystavasta</p>	FL-2.1.3	K1	1

Kysymys numero (#)	Oikea vastaus	Selitys/perustelut	Oppimistavoite (LO)	K-taso	Pistemäärä
11	b	<p>a) Väärin. Shift left -testaukseen liittyvillä käytännöillä pyritään toteuttamaan enemmän testaustoimenpiteitä ohjelmistokehityksen elinkaaren alkuvaiheissa ja ajatuksena on, että ohjelmistokehityksen elinkaari etenee vasemmalta oikealle. Testausprosessin vasenta puolta ei ole olemassa</p> <p>b) Oikein. Shift left korostaa testauksen aikaisemman aloituksen merkitystä ohjelmistokehityksen elinkaareissa. Shift leftin toteuttaminen vaatii lisäkoulutusta ja se kasvattaa työmäärää ja kustannuksia ohjelmistokehityksen elinkaaren alkuvaiheissa, mutta kokonaissäästöjen pitäisi olla korkeammat</p> <p>c) Väärin. Vaikka automatisoidut yksikkötestit ja komponentti-integraatiotestit regressiotestausta varten ovat yleensä arvokkaita, näiden testien luominen on yleensä toteuttajien vastuulla, ja jos noudatetaan jatkuvan integraation/jatkuvan toimituksen (CI/CD) lähestymistapaa, nämä testit on toimitettu koodin kanssa. Joissakin tilanteissa testaaaja voi automatisoida testejä regressiotestausta varten ja joskus jopa yksikkö- ja komponentti-integraatiotestejä varten, mutta tämä ei ole osa Shift left -lähestymistapaa, joka siirtää testausta aiemmaksi ohjelmistokehityksen elinkaareissa</p> <p>d) Väärin. Testaajien kouluttaminen suorittamaan tehtäviä varhaisessa ohjelmistokehityksen elinkaaren vaiheessa tukisi Shift left -lähestymistapaa korostamalla testauksen aloittamisen tärkeyttä aikaisemmin elinkaareissa. Shift left -lähestymistapaan ei kuitenkaan kuulu se, että automatisoidaan lisää testaustoimenpiteitä suoritettavaksi myöhemmin ohjelmistokehityksen elinkaareissa.</p>	FL-2.1.5	K2	1

Kysymys numero (#)	Oikea vastaus	Selitys/perustelut	Oppimistavoite (LO)	K-taso	Pistemäärä
12	c	<p>a) Väärin. Yksi jälkipalaverien tarkoituksista on tunnistaa mahdollisia prosessiparannuksia, jotka toteutettuina parantavat kehitysprosessin tulevien tuotosten (testauksen kohteiden) laatua. Näin ollen tämä todennäköisesti tapahtuu jälkipalaverin tuloksena.</p> <p>b) Väärin. Jälkipalaverien etuihin testauksen kannalta kuuluu testaustehokkuuden paraneminen prosessien parannusten myötä. Näin ollen tämä todennäköisesti tapahtuu jälkipalaverin tuloksena.</p> <p>c) Oikein. Jälkipalaverien osallistujina on yleensä testaaajia, toteuttajia, arkkitehteja, tuoteomistajia ja liiketoiminta-analytikoita, mutta loppukäyttäjiä kutsutaan tai he osallistuvat harvoin näihin kokouksiin – eivätkä he myöskään todennäköisesti saa raportteja näistä kokouksista. Näin ollen on hyvin epätodennäköistä, että he oppivat ja ymmärtävät enemmän kehitys- ja testausprosesseista jälkipalaverien kautta</p> <p>d) Väärin. Jälkipalaverien etuihin testauksen kannalta kuuluu testimateriaalin (mukaan lukien automatisoidut testiskriptit) laadun paraneminen toteuttajien kanssa suoritettujen yhteisten katselmointien kautta. Näin ollen tämä todennäköisesti tapahtuu jälkipalaverin tuloksena.</p>	FL-2.1.6	K2	1

Kysymys numero (#)	Oikea vastaus	Selitys/perustelut	Oppimistavoite (LO)	K-taso	Pistemäärä
13	d	<p>a) Väärin. Komponenttitestaus (kutsutaan myös yksikkötestaukseksi) sisältää yksittäisten komponenttien testaamisen erillään, ja se on enimmäkseen todentamista määrittelyitä vastaan eikä kelpuutusta käyttäjien tarpeiden perusteella. Testaajat eivät kuitenkaan yleensä tee tätä testausta, vaan yleensä toteuttajat suorittavat tämän testauksen kehitysympäristössään.</p> <p>b) Väärin. Komponentti-integraatiotestaukseen kuuluu komponenttien välisten rajapintojen ja vuorovaikutuksen testaaminen, ja se on enimmäkseen todentamista määrittelyitä vastaan eikä kelpuutusta käyttäjien tarpeiden perusteella. Testaajat eivät kuitenkaan yleensä suorita tätä testausta, vaan yleensä toteuttajat suorittavat tämän testauksen</p> <p>c) Väärin. Järjestelmäintegraatiotestaus tutkii rajapintoja muihin järjestelmiin ja ulkoisiin palveluihin, ja se on enimmäkseen todentamista määrittelyitä vastaan eikä kelpuutusta käyttäjien tarpeiden perusteella. Tämän tyyppisen testauksen suorittavat useimmiten testaajat</p> <p>d) Oikein. Hyväksymistestaus keskittyy varmistamaan (kelpuuttamaan), että järjestelmä täyttää käyttäjän liiketoimintatarpeet ja on valmis käyttöön otettavaksi. Ihannetapauksessa tämän testauksen suorittavat loppukäyttäjät</p>	FL-2.2.1	K2	1

Kysymys numero (#)	Oikea vastaus	Selitys/perustelut	Oppimistavoite (LO)	K-taso	Pistemäärä
14	b	<p>a) Väärin. Varmistustestaus on tarpeen sen tarkistamiseksi, että päivitykset ovat johtaneet oikeaan toteutukseen, mutta olisi järkevää suorittaa myös sen jälkeen regressiotestausta sen varmistamiseksi, ettei järjestelmän muuttumattomille alueille ole syntynyt tai niiltä ole paljastunut vikoja.</p> <p>b) Oikein. Varmistustestauksessa tarkistetaan, että päivitykset ovat johtaneet oikeaan toteutukseen, ja sitten regressiotestausta käytetään sen varmistamiseksi, ettei järjestelmän muuttumattomille alueille ole syntynyt tai niiltä ole paljastunut vikoja.</p> <p>c) Väärin. Regressiotestausta tulee tehdä sen varmistamiseksi, ettei järjestelmän muuttumattomille alueille ole syntynyt tai niiltä ole paljastunut vikoja päivitystä tehtäessä, mutta on myös tarpeen suorittaa varmistustestaus, joka tarkistaa, että päivitykset ovat johtaneet oikeaan toteutukseen.</p> <p>d) Väärin. Varmistustestauksella tarkistetaan, että päivitykset ovat johtaneet oikeaan toteutukseen, ja regressiotestausta käytetään sen varmistamiseksi, ettei järjestelmän muuttumattomille alueille ole syntynyt tai niillä ole paljastunut vikoja. Tehtäessä (eli kun päivitys on testattava), varmistustestaus kuitenkin edeltää regressiotestausta</p>	FL-2.2.3	K2	1

Kysymys numero (#)	Oikea vastaus	Selitys/perustelut	Oppimistavoite (LO)	K-taso	Pistemäärä
15	d	<p>Tarkastellaan jokaista lueteltu esimerkvikaa:</p> <ul style="list-style-type: none"> i. Suunnittelukuvausten kaksi eri osaa poikkeavat toisistaan suunnittelun monimutkaisuuden vuoksi – tämä on esimerkki määrittelyviasta, joihin kuuluu epäjohdonmukaisuuksia, epäselvyyksiä, ristiriitaisuuksia, puutteita, epätarkkuuksia ja päällekkäisyyksiä, jotka löytyvät helpoimmin staattisen testauksen avulla. ii. Vasteaika on liian pitkä, joten käyttäjät menettävät kärsivällisyytensä – tämä on esimerkki vasteaikaviasta, joka voidaan havaita käytännössä vain suorittamalla ohjelma ja mittaamalla vasteaika, ja joka on siis helpoimmin löydettävissä dynaamisella testauksella. iii. Koodissa olevaa polkua ei voida saavuttaa suorituksen aikana - tämä on esimerkki koodausviasta, joihin kuuluu muuttujia, joille ei ole määritetty arvoja, määrittelemättömät muuttujat, monistettu tai saavuttamaton koodi sekä liiallinen koodin monimutkaisuus, ja joka on helpoimmin löydettävissä staattisella testauksella iv. Muuttuja määritellään, mutta sitä ei käytetä myöhemmin ohjelmassa - tämä on esimerkki koodausviasta, joihin kuuluu muuttujia, joille ei ole määritetty arvoja, määrittelemättömät muuttujat, monistettu tai saavuttamaton koodi sekä liiallinen koodin monimutkaisuus, ja joka on helpoimmin löydettävissä staattisella testauksella v. Ohjelman raportin luomiseen tarvitseman muistin määrä on liian suuri – tämä on esimerkki suorituskykyviasta, joka voidaan käytännössä havaita vain suorittamalla ohjelma ja mittaamalla käytetty muisti, ja joka löytyy siis helpoimmin dynaamisella testauksella <p>Näin ollen: a) Väärin b) Väärin c) Väärin d) Oikein. Parhaiten staattisella testauksella löytyvät i, iii ja iv</p>	FL-3.1.3	K2	1

Kysymys numero (#)	Oikea vastaus	Selitys/perustelut	Oppimistavoite (LO)	K-taso	Pistemäärä
16	a	<p>a) Oikein. Palautteen saaminen sidosryhmiltä aikaisin ja usein ohjelmistokehitysprosessissa voi olla erittäin hyödyllistä. Se helpottaa aikaista viestintää mahdollisista laatuongelmista, voi estää vaatimuksiin liittyviä väärinkäsityksiä ja varmistaa, että kaikki sidosryhmien vaatimusten muutokset ymmärretään ja toteutetaan aikaisemmin</p> <p>b) Väärin. Palaute tulee sidosryhmiltä, ja on epätodennäköistä, että palautteen antaminen parantaisi heidän ymmärrystään liittyen heidän omiin käyttäjävaatimuksiinsa</p> <p>c) Väärin. Palautteen saaminen sidosryhmiltä aikaisin ja usein ohjelmistokehitysprosessissa voi olla erittäin hyödyllistä. Se helpottaa aikaista viestimistä mahdollisista laatuongelmista, voi estää vaatimuksiin liittyviä väärinkäsityksiä ja varmistaa, että kaikki sidosryhmien vaatimusten muutokset ymmärretään ja toteutetaan aikaisemmin. Se ei kuitenkaan tarkoita, että vaikka vaatimusten muutokset voidaan ymmärtää ja toteuttaa aikaisemmin, vaatimukseen kannustettaisiin tekemään rajattomasti muutoksia.</p> <p>d) Väärin. Palaute tulee sidosryhmiltä, eikä se koske viestintää heille. Viestintä loppukäyttäjille voisi sisältää tiedottamisen siitä, mitä vaatimuksia ei toteuteta ennen julkaisua, mutta ihannetilanteessa näin ei pitäisi tapahtua ollenkaan</p>	FL-3.2.1	K1	1

17	b	<p>Tarkastellaan jokaista lueteltua katselmointityyppiä:</p> <ol style="list-style-type: none"> 1. Tekninen katselmointi – Tämän tyyppisen katselmoinnin suorittavat teknisesti pätevät katselmoijat ja sitä johtaa puheenjohtaja. Tavoitteena on saavuttaa yhteisymmärrys ja tehdä teknisiin ongelmiin liittyviä päätöksiä sekä arvioida laatua ja rakentaa luottamusta tuotokseen, luoda uusia ideoita, tekijöiden motivointi ja kehittymisen mahdollistaminen sekä poikkeamien havaitseminen. 2. Epämuodollinen katselmointi – Päättävänä on löytää poikkeamia. Prosessia ei ole määritelty, eikä tulosten muodollista dokumentointia vaadita 3. Tarkastus – Tämä on muodollisin katselmointityyppi, ja siinä noudatetaan koko yleistä katselmointiprosessia. Ensisijaisena tavoitteena on löytää suurin osa poikkeavuuksista, ja muita tavoitteita ovat laadun arviointi ja luottamuksen kasvattaminen tuotokseen, tekijöiden motivointi ja kehittymisen mahdollistaminen. Lisäksi kerätään mittaritietoja, joiden avulla voidaan parantaa ohjelmistokehityksen elinkaarta (SDLC), katselmointiprosessi mukaan luettuna. Tekijä ei voi toimia katselmoinnin vastuuhenkilönä tai kirjuriina 4. Läpikäynti - Tämäntyyppinen katselmointi on tekijän vetämä ja se palvelee erilaisia tavoitteita, kuten laadun arviointi ja luottamuksen rakentaminen tuotokseen, katselmoijien kouluttaminen, yhteisymmärryksen saavuttaminen, uusien ideoiden luominen, tekijöiden motivointi ja kehittymisen mahdollistaminen sekä poikkeavuuksien löytäminen. Katselmoijat voivat tehdä yksilöllisen katselmoinnin ennen läpikäyntiä, mutta se ei ole pakollista <ol style="list-style-type: none"> A. Tavoitteisiin kuuluvat yhteisymmärryksen saavuttaminen, uusien ideoiden luominen ja tekijöiden motivoiminen kehittymään (A) B. Päättävänä on löytää mahdollisia vikoja, eikä muodollista dokumentoitua tulosta tuoteta (D) C. Päättävänä on löytää mahdollisia vikoja, ja mittaritietojen keräämistä edellytetään prosessin parantamisen tueksi (C) 	FL-3.2.4	K2	1
----	---	--	----------	----	---

Kysymys numero (#)	Oikea vastaus	Selitys/perustelut	Oppimistavoite (LO)	K-taso	Pistemäärä
		<p>D. Tavoitteisiin kuuluvat katselmoijien kouluttaminen, yhteisymmärryksen saavuttaminen, uusien ideoiden luominen ja mahdollisten vikojen havaitseminen (B)</p> <p>Näin ollen: a) Väärin b) Oikein. Oikeat vastaavuudet ovat: 1A, 2D, 3C, 4B c) Väärin d) Väärin</p>			
18	b	<p>a) Väärin. Onnistuneiden katselmointien varmistamiseksi on tärkeää varmistaa johdon tuki katselmointiprosessille, mutta se ei kuitenkaan tarkoita, että heidän pitäisi osallistua katselmointeihin katselmoijina</p> <p>b) Oikein. Onnistuneiden katselmointien varmistamiseksi on tärkeää jakaa katselmoitava tuotos osiin, jotka ovat riittävän pieniä, jotta ne voidaan katselmoida kohtuullisessa ajassa. Näin varmistetaan, että katselmoijat eivät menetä keskittymiskykyään yksilöllisen katselmoinnin tai katselmointipalaverin aikana.</p> <p>c) Väärin. Onnistuneiden katselmointien varmistamiseksi on tärkeää määritellä selkeästi tavoitteet ja mitattavissa olevat lopetuskriteerit, ilman osallistujien arviointia.</p> <p>d) Väärin. Onnistuneiden katselmointien varmistamiseksi on tärkeää jakaa katselmointi pienempiin osiin, jotta katselmoijat eivät menetä keskittymiskykyään yksilöllisen katselmoinnin tai katselmointipalaverin aikana. Suunnittelun ei siis pitäisi lähteä siitä, että yhdessä katselmoinnissa katselmoidaan yksi asiakirja.</p>	FL-3.2.5	K1	1

Kysymys numero (#)	Oikea vastaus	Selitys/perustelut	Oppimistavoite (LO)	K-taso	Pistemäärä
19	c	<p>a) Väärin. Useimmissa tapauksissa samoihin testauksen kohteisiin voidaan käyttää sekä mustalaatikkotekniikoita että kokemuspohjaisia tekniikoita</p> <p>b) Väärin. Sekä mustalaatikkotekniikoita että kokemuspohjaisia testaustekniikoita voidaan käyttää kaikilla testaustasoilla</p> <p>c) Oikein. Mustalaatikkotekniikat (tunnetaan myös määrittelypohjaisina tekniikoina) perustuvat testauksen kohteen määrittelyyn käyttäytymisen analyysiin kiinnittämättä huomiota sen sisäiseen rakenteeseen. Näin ollen testauksen pohjamateriaalina on yleensä määrittelykuvaus. Kokemuspohjaiset testaustekniikat käyttävät tehokkaasti testaaajien tietoja ja kokemusta testitapausten suunnittelussa ja valmistelussa. Tämä tarkoittaa, että testaaaja ei välttämättä testejä suunnitellessaan käytä lainkaan määrittelykuvauksia.</p> <p>d) Väärin. Kokemuspohjaisilla testaustekniikoilla voidaan havaita vikoja, jotka saattavat jäädä huomiotta mustalaatikko- (ja lasilaatikko) -testaustekniikoita käytettäessä. Näin ollen kokemuspohjaiset tekniikat täydentävät mustalaatikkotekniikoita ja lasilaatikkotekniikoita, ja sekä mustalaatikkotekniikoita että kokemuspohjaisia testaustekniikoita voidaan käyttää kaikissa ohjelmistokehityksen elinkaarimalleissa.</p>	FL-4.1.1	K2	1
20	c	<p>a) Väärin. Näillä kolmella arvolla saavutetaan ekvivalenssiluokkien täysi kattavuus, mutta se voidaan saavuttaa vain kahdella arvollakin</p> <p>b) Väärin. Näillä kolmella arvolla saavutetaan ekvivalenssiluokkien täysi kattavuus, mutta se voidaan saavuttaa vain kahdella arvollakin</p> <p>c) Oikein. Arvo "1" kattaa luokat "pituus väärä" ja "eri numeroiden määrä virheellinen". Arvo "1234" kattaa luokat "pituus oikea" ja "eri numeroiden määrän oikein". Nämä kaksi arvoa kattavat kaikki neljä tunnistettua ekvivalenssiluokkaa</p> <p>d) Väärin. Tämä joukko ei kata "eri numeroiden määrää virheellinen" -ekvivalenssiluokkaa</p>	FL-4.2.1	K3	1

Kysymys numero (#)	Oikea vastaus	Selitys/perustelut	Oppimistavoite (LO)	K-taso	Pistemäärä
21	d	<p>Ekvivalenssiluokat ovat: {..., 99, 100}, {101, 102, ..., 198, 199}, {200, 201, ...}. Näin ollen on olemassa 4 raja-arvoa, jotka ovat: 100, 101, 199 ja 200.</p> <p>2-arvoraja-arvoanalyysissä jokaiselle raja-arvolle on kaksi kattavuuskohdetta (raja-arvo ja sen lähin naapuri, joka kuuluu viereiseen osioon). Koska lähimmät naapurit ovat myös viereisen osion raja-arvoja, kattavuuskohteita on vain neljä .</p> <p>Näin ollen:</p> <p>a) Väärin. Vain 100 ja 200 ovat 2-arvoraja-arvoanalyysin kelvollisia kattavuuskohteita, joten näillä arvoilla saavutetaan 50 % kattavuus</p> <p>b) Väärin. Vain 100 ja 200 ovat 2-arvoraja-arvoanalyysin kelvollisia kattavuuskohteita, joten näillä arvoilla saavutetaan 50 % kattavuus</p> <p>c) Väärin. Vain 100 ja 101 ovat 2-arvoraja-arvoanalyysin kelvollisia kattavuuskohteita, joten näillä arvoilla saavutetaan 50 % kattavuus</p> <p>d) Oikein. 101, 199 ja 200 ovat 2-arvoraja-arvoanalyysin kelvollisia kattavuuskohteita, joten näillä arvoilla saavutetaan 75 % kattavuus</p>	FL-4.2.2	K3	1
22	d	<p>a) Väärin. Yhdistelmä (K, K, E) ei vastaa mitään sääntöä. Tämä on esimerkki puutteesta, ei ristiriita</p> <p>b) Väärin. Yhdistelmä (K, E, K) vastaa vain yhtä saraketta, R2, joten ristiriitaa ei ole</p> <p>c) Väärin. Molemmat yhdistelmät (K, K, K) ja (E, K, K) vastaavat vain yhtä saraketta, R1, joten ristiriitaa ei ole</p> <p>d) Oikein. Yhdistelmä (E, E, E) vastaa sekä R2:ta että R3:a, mutta R2:lla ja R3:lla on eri toimenpiteet, joten tässä on ristiriita R2:n ja R3:n välillä.</p>	FL-4.2.3	K3	1

Kysymys numero (#)	Oikea vastaus	Selitys/perustelut	Oppimistavoite (LO)	K-taso	Pistemäärä
23	a	<p>Seuraavat kolme siirtymää "VARAUSPYYNTÖ → VAHVISTUS" "ODOTUSLISTA → VAHVISTUS" "ODOTUSLISTA → LOPPU"</p> <p>eivät voi esiintyä samassa testitapauksessa, mikä viittaa siihen, että tarvitaan vähintään kolme testitapausta. Kaikki muut siirtymät voivat esiintyä yhdessä yhden tai useamman näiden kolmen siirtymän kanssa, joten tarvitaan vähintään kolme testitapausta. Itse asiassa on vain kolme mahdollista siirtymäketjua: TT1: ALKU (Huonetiedustelu) → VARAUSPYYNTÖ (saatavilla) → VAHVISTUS (Maksu) → LOPPU TT2: ALKU (Huonetiedustelu) → VARAUSPYYNTÖ (Ei saatavilla) → ODOTUSLISTA (Saatavilla) → VAHVISTUS (Maksu) → LOPPU TT3: ALKU (Huonetiedustelu) → VARAUSPYYNTÖ (Ei saatavilla) → ODOTUSLISTA (Peruutus) → LOPPU</p> <p>Näin ollen: a) Oikein b) Väärin c) Väärin d) Väärin</p>	FL-4.2.4	K3	1
24	c	<p>Haaratestauksessa kattavuuskohteita ovat haarat, joita edustavat kontrollivuokaavion siirtymänuolet. Kontrollivuokaaviossa on 8 nuolta.</p> <p>Näin ollen: a) Väärin b) Väärin c) Oikein d) Väärin</p>	FL-4.3.2	K2	1

Kysymys numero (#)	Oikea vastaus	Selitys/perustelut	Oppimistavoite (LO)	K-taso	Pistemäärä
25	a	<p>a) Oikein. Pelkän mustalaatikkotestauksen suorittaminen ei tuota tietoja todellisesta koodikattavuudesta. Lasilaatikkokattavuusmittarit tuottavat objektiivisen kattavuuden mittauksen ja tarjoavat tarvittavaa tietoa, jotta kattavuuden lisäämiseksi voidaan luoda lisää testejä ja sitä kautta lisätä luottamusta koodiin</p> <p>b) Väärin. Tämä väite on väärin, eikä se myöskään vastaa kysymykseen (sillä ei ole mitään tekemistä mustalaatikkotestauksen kanssa)</p> <p>c) Väärin. Yleisesti ottaen lasilaatikko- ja mustalaatikkotekniikoiden välillä ei ole alisteisuussuhteita</p> <p>d) Väärin. Lasilaatikkotekniikoilla suunnitellaan testejä itse testauksen kohteen perusteella, kun taas mustalaatikkotekniikoita käytetään testien suunnitteluun määrittelyiden perusteella. Siksi näillä kahdella tekniikalla johdettujen kattavuuskohteiden välillä ei ole yhteyttä</p>	FL-4.3.3	K2	1

Kysymys numero (#)	Oikea vastaus	Selitys/perustelut	Oppimistavoite (LO)	K-taso	Pistemäärä
26	b	<p>a) Väärin. Tutkivassa testauksessa käytetään testausohjeita, ei luetteloa mahdollisista vioista/häiriöistä. Vaikka tutkivassa testauksessa voidaan käyttää muita testaustekniikoita, tässä tapauksessa vikahyökkäys on todennäköisin vaihtoehto</p> <p>b) Oikein. Tämä on luettelo mahdollisista häiriöistä. Vikahyökkäykset ovat järjestelmällinen lähestymistapa virheenarvaukseen ja ne edellyttävät, että testaaaja luo tai hankkii luettelon mahdollisista virheistä, vioista ja häiriöistä, ja suunnittelee testejä, jotka tunnistavat virheisiin liittyviä vikoja, paljastavat vikoja tai aiheuttavat häiriöitä</p> <p>c) Väärin. Testaaaja käyttää tarkistuslistaa testauksensa tueksi. Sekä virheenarvaus että tarkistuslistoihin perustuva testaus käyttävät tällaisia listoja, mutta tässä lista sisältää mahdollisia häiriöitä, ei testattavia tilanteita, joten TODENNÄKÖISIN testaustekniikka on vikahyökkäys, joka keskittyy virheisiin, vioihin ja häiriöihin.</p> <p>d) Väärin. Raja-arvoanalyysi perustuu ekvivalenssiosioiden raja-arvojen analyysiin. Yllä oleva luettelo ei mainitse ekvivalenssiosioita tai niiden raja-arvoja</p>	FL-4.4.1	K2	1

Kysymys numero (#)	Oikea vastaus	Selitys/perustelut	Oppimistavoite (LO)	K-taso	Pistemäärä
27	d	<p>a) Väärin. Vaikka on totta, että testaaaja voi valmistella ja suorittaa yksityiskohdaisia testitapauksia tarkistuslistan perusteella, se ei selitä, kuinka tämä johdaisi kattavuuden lisääntymiseen</p> <p>b) Väärin. Tarkistuslistan kohtia ei pidä automatisoida . Mutta vaikka niitä automatisoitaisiinkin, automatisoidut testiskriptit suorittavat testit aina samalla tavalla, mikä ei yleensä johda kattavuuden lisääntymiseen</p> <p>c) Väärin. On totta, että jokainen tarkistuslistan kohta tulisi testata erikseen ja itsenäisesti. Tämä vaikuttaa kuitenkin testien suoritusjärjestykseen eikä vaikuta saavutettuun kattavuuteen, eikä siten johda kattavuuden lisääntymiseen</p> <p>d) Oikein. Jos tarkistuslistat ovat ylätasoinen listoja, varsinaisessa testauksessa esiintyy todennäköisesti vaihtelua, mikä voi johtaa <u>suurempaan kattavuuteen</u> mutta heikompaan toistettavuuteen. Jos kaksi testaaajaa käyttää ylätasoinen tarkistuslistaa, kumpikin heistä voi käyttää erilaista testiaineistoa, testiaskelia jne. Tällä tavalla yksi testaaaja todennäköisesti kattaa joitain alueita, joita toinen testaaaja ei kata, ja tämä lisää kattavuutta</p>	FL-4.4.3	K2	1

Kysymys numero (#)	Oikea vastaus	Selitys/perustelut	Oppimistavoite (LO)	K-taso	Pistemäärä
28	b	<p>a) Väärin. Tämä hyväksymiskriteeri kuvaa, mitä sääntöjä tai määräyksiä järjestelmän tulee noudattaa (tässä tapauksessa oikeus tulla poistetuksi). Tämä on esimerkki sääntösuuntautuneesta hyväksymiskriteeristä</p> <p>b) Oikein. Tämä hyväksymiskriteeri kuvaa esimerkkiskenaarion, joka järjestelmällä on pystyttävä toteuttamaan. Tämä on esimerkki skenaariosuuntautuneesta hyväksymiskriteeristä</p> <p>c) Väärin. Tämä lause näyttää enemmän koodiriviltä, joka toteuttaa jonkin liiketoimintasäännön. Hyväksymiskriteerit tulee kirjoittaa yhteistyössä liiketoiminnan edustajien kanssa, ja siksi ne tulee kirjoittaa heidän ymmärtämälleen kielellä. Tämä lause on mitä todennäköisimmin näille sidosryhmille käsitteämätön</p> <p>d) Väärin. Tämä hyväksymiskriteeri kuvaa, mitä sääntöjä tai määräyksiä järjestelmän on noudatettava ja kuinka niiden noudattaminen varmistetaan. Siksi tämä on esimerkki sääntösuuntautuneesta hyväksymiskriteeristä, ei skenaariosuuntautuneesta hyväksymiskriteeristä</p>	FL-4.5.2	K2	1
29	d	<p>a) Väärin. Haluamme tarkistaa, että erikoiskäyttäjillä on tavallisten käyttäjien oikeudet, joten meidän on testattava erikoiskäyttäjän käyttöoikeudet, ei tavallisen käyttäjän oikeuksia</p> <p>b) Väärin. Haluamme tarkistaa, että erikoiskäyttäjillä on tavallisten käyttäjien oikeudet, joten meidän on testattava erikoiskäyttäjän käyttöoikeudet, ei tavallisen käyttäjän oikeuksia</p> <p>c) Väärin. Hyväksymiskriteereissä ei ole mainittu kerrosta 5. Testitapaukset eivät saa laajentaa käyttäjätarinaa. Mutta vaikka haluaisimme tehdä negatiivista testausta, tämä testi ei liity suoraan AC3:een</p> <p>d) Oikein. Tällä tavalla voimme tarkistaa, onko erikoiskäyttäjällä pääsy kerroksiin, joihin tavallisella käyttäjällä on pääsy</p>	FL-4.5.3	K3	1

Kysymys numero (#)	Oikea vastaus	Selitys/perustelut	Oppimistavoite (LO)	K-taso	Pistemäärä
30	a	<p>a) Oikein. Testaussuunnitelma voi sisältää testiaineiston <i>vaatimukset</i> (osana testauksen lähestymistapaa), mutta ei testitapausten yksityiskohtaista testiaineistoa. Testiaineisto on osa testitapauksia, ei testaussuunnitelmaa. Lisäksi on yleensä mahdotonta määrittellä testiaineistoa testaussuunnitelmaa laadittaessa, koska ei ole tarkkaan tiedossa, miltä komponentit näyttävät</p> <p>b) Väärin. Yksi testaussuunnitelman tavoitteista on auttaa varmistamaan, että suoritettujen testausmenpiteet täyttävät määritellyt vaatimukset sisällyttämällä suunnitelmaan aloitus- ja lopetuskriteerit. Koodikattavuusvaatimukset ovat esimerkki tällaisesta kriteeristä yksikkötestaustasolla.</p> <p>c) Väärin. Dokumentaation mallipohjat kuuluvat tyypillisesti testaussuunnitelman sisältöön. Vakiomuotoisen viestintä- tai raportointitavan määrittely helpottaa sidosryhmien välistä viestintää</p> <p>d) Väärin. Yksi testaussuunnitelman tavoitteista on osoittaa, että testaus noudattaa olemassa olevaa testauspolitiikkaa ja -strategiaa, tai selittää, miksi testaus poikkeaa niistä. Tämä on esimerkki testaustasojen noudattamisesta (tai noudattamatta jättämisestä) koskevan poikkeaman selittämisestä.</p>	FL-5.1.1	K2	1
31	c	<p>Kaavion perusteella: $A(4)=6$ ja $A(3)=8$ (kaksi viimeistä harmaata ruutua).</p> <p>Kaavan perusteella: $E(5) = (3 \cdot A(4) + A(3)) / 4 = (3 \cdot 6 + 8) / 4 = 26 / 4 = 6,5$ henkilötyöpäivää.</p> <p>Näin ollen: a) Väärin b) Väärin c) Oikein d) Väärin</p>	FL-5.1.4	K3	1

Kysymys numero (#)	Oikea vastaus	Selitys/perustelut	Oppimistavoite (LO)	K-taso	Pistemäärä
32	a	<p>Testitapaukset halutaan suorittaa niiden prioriteettien mukaan, mutta huomioon on otettava myös riippuvuudet.</p> <p>Jos otetaan huomioon vain prioriteetit, ensin halutaan suorittaa TT 5 ja TT 7 (korkein prioriteetti), sitten TT 1, TT 3 ja TT 4 ja lopuksi TT 2 ja TT 6 (matalin prioriteetti).</p> <p>Kuitenkin, jotta TT 7 voidaan suorittaa, on ensin suoritettava TT 4.</p> <p>Jotta voidaan suorittaa TT 5, on suoritettava TT 4 ja TT 2, mutta TT 2:n suorituksen estää TT 1, joka tulisi suorittaa ennen TT 2:ta.</p> <p>Näin ollen, jotta prioriteetin 1 testitapaukset voidaan suorittaa mahdollisimman aikaisin, viiden ensimmäisen testitapauksen tulee olla: TT 4 - TT 7 - TT 1 - TT 2 - TT 5.</p> <p>Seuraavaksi on suoritettava TT 3, koska sillä on korkeampi prioriteetti kuin TT 6:lla.</p> <p>Näin ollen koko järjestys on TT 4 – TT 7 – TT 1 – TT 2 – TT 5 – TT 3 – TT 6.</p> <p>Kuudes testitapaus on siis TT 3.</p> <p>Näin ollen: a) Oikein b) Väärin c) Väärin d) Väärin</p>	FL-5.1.5	K3	1
33	b	<p>a) Väärin. Testipyramidimalli ei anna tietoa testien prioriteeteista</p> <p>b) Oikein. Testipyramidimalli osoittaa, että eri testeillä on erilainen yksityiskoh- taisuuden taso</p> <p>c) Väärin. Testipyramidimalli on riippumaton kattavuuskriteereistä</p> <p>d) Väärin. Testipyramidimalli ei kuvaa mitään eri testien välisiä suhteita</p>	FL-5.1.6	K1	1

Kysymys numero (#)	Oikea vastaus	Selitys/perustelut	Oppimistavoite (LO)	K-taso	Pistemäärä
34	d	<p>a) Väärin. Testausneljännekset ryhmittelevät testaustasot ja testaustyyppit erikseen useiden kriteerien mukaan. Ne eivät kuvaa mitään testaustasojen ja testaustyyppien yhdistelmiä, eivätkä ne liity mihinkään kohtaan ohjelmistokehityksen elinkaaren sisällä. Sekä testaustasoja että testautyyppijä käsitellään erikseen testausneljännesmallissa</p> <p>b) Väärin. Testausneljännekset ryhmittelevät testaustasot ja testautyyppit useiden kriteerien mukaan. Ne eivät kuvaa kullakin testitasolla suoritettujen yksittäisten testautyyppien yksityiskohtaisuuden astetta. Tällaista testautasoihin liittyvää mallia kutsutaan testautuspyramidiksi</p> <p>c) Väärin. Väite on väärä, koska yleensä mikä tahansa testautyyppi voidaan suorittaa millä tahansa testautasolla</p> <p>d) Oikein. Testausneljännekset ryhmittelevät testautasot, testautyyppit, toimenpiteet, testautetekniikat ja tuotokset ketterässä ohjelmistokehityksessä. Tässä mallissa testit voivat olla liiketoimintasuuntautuneita tai teknologiasuuntautuneita. Testit voivat tukea tiimiä (eli ohjata kehitystä) tai arvioida tuotetta (eli mitata sen käyttäytymistä odotuksia vastaan). Näiden kahden näkökulman yhdistelmä määrittää neljä neljänestä</p>	FL-5.1.7	K2	1
35	c	<p>a) Väärin. Riskien seuranta on osa riskienhallintaa, ei riskianalyysiä</p> <p>b) Väärin. Riskien tunnistaminen itsessään ei anna mahdollisuutta toteuttaa riskien pienentämistoimia. Pienentämistoimenpiteet määritellään riskien hallintavaiheessa</p> <p>c) Oikein. Tämä on esimerkki siitä, kuinka riskianalyysi vaikuttaa testautuksen perusteellisuuteen ja laajuuteen</p> <p>d) Väärin. Kattavuuskohteet määritellään testautetekniikoita käyttämällä, ei riskianalyysin avulla</p>	FL-5.2.3	K2	1

Kysymys numero (#)	Oikea vastaus	Selitys/perustelut	Oppimistavoite (LO)	K-taso	Pistemäärä
36	b	<p>a) Väärin. Testauksen edistymisraportteja käytetään enimmäkseen testauksen seurannan ja hallinnan sekä testauksen päättämisen aikana, ei testien suunnittelun aikana</p> <p>b) Oikein. Testauksen loppuraportti laaditaan testauksen päättämisen yhteydessä, kun projekti, testaustaso tai testaustyyppi on valmis ja, ihannetilanteessa, sen päätöskriteerit ovat täyttyneet. Tämä raportti käyttää tietoja testauksen edistymisraporteista sekä muita tietoja</p> <p>c) Väärin. Testauksen edistymisraportteja käytetään enimmäkseen testauksen seurannan ja hallinnan sekä testauksen päättämisen aikana, ei testianalyysin aikana</p> <p>d) Väärin. Testauksen edistymisraportteja käytetään eniten testauksen seurannan ja hallinnan sekä testauksen päättämisen aikana, ei testauksen suunnittelun aikana</p>	FL-5.3.2	K2	1
37	d	<p>a) Väärin. Kun käyttäjä ilmoittaa ohjelmistohäiriöstä, on materiaalien lisäysten yksilöllisten tunnisteen ansiosta mahdollista koota uudelleen käyttäjän käyttämän ohjelmistoversion tiedostot (sekä vastaavat testiskriptien versiot) ja siten toistaa häiriö ja paikantaa vika nopeammin</p> <p>b) Väärin. Jos testiympäristön muutos aiheuttaa odottamattomia ongelmia testauksen aikana, kokoonpanonhallinnan avulla testaajat voivat palata ympäristön aiempaan versioon. Tämä varmistaa, että testausta voidaan jatkaa ilman, että muutos vaikuttaa siihen</p> <p>c) Väärin. Kokoonpanonhallinta varmistaa, että kaikkiin yksilöityihin dokumentteihin (esim. vaatimusmäärittelyihin) ja ohjelmiston osiin viitataan yksiselitteisesti testausdokumentaatioissa (esim. testaussuunnitelmissa)</p> <p>d) Oikein. Tämä varmistetaan vikojenhallinnan, ei kokoonpanohallintaprosessin avulla</p>	FL-5.4.1	K2	1

Kysymys numero (#)	Oikea vastaus	Selitys/perustelut	Oppimistavoite (LO)	K-taso	Pistemäärä
38	b	a) Väärin. Tämä on tärkeää, mutta ei yhtä tärkeää kuin testiympäristön elementit b) Oikein. Tärkeä asia, joka puuttuu, on testauksessa käytetyn selaimen ja laitteen tunniste. Selain- ja laitetiedot ovat tärkeitä, koska tällainen vika voi olla selain- tai laitekohtainen. Esimerkiksi kirjautumispainike voi toimia hyvin yhdellä selaimella (tai tietyn selaimen yhdellä versiolla), mutta ei toisella. Siksi selain- ja laitetiedot voivat auttaa toteuttajia toistamaan ongelman ja löytämään ongelman juurisyyn nopeammin c) Väärin. Testauksen kohde on nimetty (WebShop v0.99) d) Väärin. Vaikutus on mainittu – se on vakavuus (korkea)	FL-5.5.1	K3	1
39	d	a) Väärin. Testien suoritus- ja kattavuustyökalut helpottavat testitapausten automatisoitua suoritusta ja nämä testitapaukset suorittamalla saavutetun kattavuuden mittaamista. Nämä työkalut eivät kuitenkaan auta vikojen järjestämisessä ja kokoonpanonhallinnassa b) Väärin. Testien suunnittelu- ja valmistelutyökalut helpottavat testitapausten, testiaineiston ja testiproseduurien luomista, mutta ne eivät auta vikojen organisoinnissa ja kokoonpanonhallinnassa c) Väärin. Vikojenhallintatyökaluja käytetään vikojen hallintaan, mutta ne eivät ole testaustyökaluja, eikä niitä käytetä testitapausten tai kokoonpanonhallinnan järjestämiseen d) Oikein. Testauksenhallintatyökalut lisäävät testausprosessin tehokkuutta helpottamalla ohjelmistokehityksen elinkaaren, vaatimusten, testien, vikojen ja kokoonpanonhallintaa.	FL-6.1.1	K2	1

Kysymys numero (#)	Oikea vastaus	Selitys/perustelut	Oppimistavoite (LO)	K-taso	Pistemäärä
40	d	<p>a) Väärin. "Kyky luoda testitapauksia ilman pääsyä käsiksi testauksen pohjamateriaaliin" ei ole mahdollista. Testitapausten luominen joko testaaajien toimesta tai työkalujen avulla vaatii testauksen pohjamateriaalin saatavilla oloa.</p> <p>b) Väärin. "Lisääntyneen kattavuuden saavuttaminen objektiivisemmän arvioinnin avulla" ei ole testiautomaation suora hyöty. Testiautomaatio tuottaa objektiivisemmän arvion kattavuudesta, mutta objektiivinen arviointi ei lisää kattavuutta. Vain käyttämällä kattavuustuloksia lisätestien kirjoittamiseen voidaan kattavuutta mahdollisesti lisätä</p> <p>c) Väärin. "Testien suoritusajojen pidentyminen käytettävissä olevan suuremman prosessointitehon myötä" on ristiriitainen väite, koska suurempi prosessointiteho normaalisti lyhentäisi suoritusajojen, ja pidentyneet suoritusajat eivät ole hyödyksi, koska testaus kestäisi kauemmin.</p> <p>d) Oikein. Inhimillisten virheiden estäminen suuremman johdonmukaisuuden ja toistettavuuden avulla on testiautomaation etu, koska testiautomaatio ei voi kärsiä inhimillisistä virheistä. Se tarkoittaa esimerkiksi sitä, että testit johdetaan johdonmukaisesti vaatimuksista, testiaineisto luodaan systemaattisesti ja testit suoritetaan työkalulla samassa järjestyksessä samalla tiheydellä.</p>	FL-6.2.1	K1	1