

CoMa - Testaussuunnitelam

Mindmap-ryhmä / Kari Velling

Helsinki 15.12.2005

Ohjelmistotuotantoprojekti

HELSINGIN YLIOPISTO

Tietojenkäsittelytieteen laitos

Kurssi

581260 Ohjelmistotuotantoprojekti (6 ov)

Projektiryhmä

Antti Kavonen

Ilari Nieminen

Tiina Torvinen

Valtteri Tyrsky

Kari Velling

Antti Vähäkotamäki

Asiakas

Anni Rytönen

Johtoryhmä

Juho Teuvo (ohjaaja)

Juha Taina

Kotisivu

<http://www.cs.helsinki.fi/group/mindmap>

Versiohistoria

Versio	Päiväys	Tehdyt muutokset
1.0	15.12.2005	Testausraportin ensimmäinen versio

Sisältö

1	Yleistä	1
1.1	Testauksen tavoitteet	1
1.2	Testausstrategia	1
2	Testausprosessi	1
2.1	Testivaiheet	1
2.2	Testauksen pohjadokumentit	2
2.3	Aloitus- ja lopetuskriteerit	2
2.4	Resurssit	3
2.5	Aikataulu	3
3	Testiaineiston hallinta	3
3.1	Testauksen dokumentointi	3
3.2	Testausaineiston hallinta	3
4	Testiympäristö ja välineet	3
4.1	Testiympäristöt	3
4.2	Testausvälineet	3
5	Virheiden arvionti	3
6	Testauksen seuranta	4
6.1	Palaverit	4
6.2	Raportointi	4
6.3	Mittarit	4
7	Testaajien koulutukset	5
	Lähteet	5

1 Yleistä

1.1 Testauksen tavoitteet

Testauksella varmistetaan sovelluksen laatu ennen toimitusta asiakkaalle niin, että sovellus on määrittelyjen mukainen ja läpäisee hyväksymistestauksen.

1.2 Testausstrategia

Sovelluksen laadun varmistaminen toteutetaan usealla toistaan seuraavalla testitasolla; yksikkö-, integrointi- ja järjestelmätestaus, jossa jokaisen testitason onnistunut läpivienti on riippuvainen edeltävästä testitasosta/testivaiheesta. Näistä riippuvuuksista johtuen käytetään selkeitä entry/exit-kriteeritä siirryttäessä tasolta/vaiheesta toiselle, joita tulee noudattaa. Kunkin testitason loppuraportissa raportoidaan mm. entry/exit-kriteerien täyttyminen.

Käytetään standardoitua geneeristä testiprosessia:

- Testin suunnittelu: testitasojen suunnittelu.
- Testin määrittely: testitasojen testitapaussuunnittelu.
- Testiajo: testitasojen testitapausten ajaminen.
- Analyysi ja defektiraportointi: testitasojen testiajon tulosten vertaaminen odotettuihin tuloksiin ja virheiden raportointi.
- Lopetus: testitasojen testauksen kattavuuden arviointi testisuunnitelmaa vasten ja loppuraportti.

Testiprosessin vaiheiden sisältö on riippuvainen testitasosta. Tämä esitetään tarkemmin kunkin testitason suunnitelmassa.

2 Testausprosessi

2.1 Testivaiheet

2.1.1 Yksikkö- ja integrointitestaus

Yksikkö- ja integrointitestaus tehdään projektiryhmän toimesta. Yksikkötestauksessa testataan jokainen toteutettu yksikkö (metodi tai palvelu tms.) sen toteutusvaatimuksia vasten. Integrointitestauksella varmistetaan, että yksiköiden rajapinnat on toteutettu oikein ja että ne toimivat määrittelyjen mukaisesti.

2.1.2 Järjestelmätestaus

Järjestelmätestauksessa testataan järjestelmän toiminta loppukäyttäjän näkökulmasta määrittelyjä vasten. Tässä testauksessa testitapaukset ovat kokonaisia tehtäviä ja tehtäväketjuja. Järjestelmätestauksessa tulee järjestelmän toimintaa testattaessa ja testitapauksia suunniteltaessa ottaa huomioon ainakin seuraavat näkökulmat:

- toiminnallisuus
- suorituskyky
- käyttäjän ohjeet
- käytettävyys

Järjestelmätestauksessa testauksen vastuu on projektiryhmällä. Asiakas voi toimittaa tarvittavan testidatan. Järjestelmätestaus tehdään yliopiston Linux-ympäristössä ja testidatan tulee sisällöltään ja määrältään vastata mahdollisimman paljon loppukäytön tilannetta.

2.1.3 Hyväksymistestaus

Hyväksymistestauksen tekee asiakas ja se on asiakkaan vastuulla. Hyväksymistestaus tehdään yliopiston Linux-ympäristössä. Projektiryhmällä on valmius osallistua hyväksymistestauksessa testisuunnitteluun, testitapaussuunnitteluun ja/tai testiajoon mikäli asiakas näin toivoo. Tällöin asiasta sovitaan erikseen.

2.2 Testauksen pohjadokumentit

Tässä listataan dokumentit, joiden pohjalta testaussuunnitelma tehdään

Yksikkö- ja integrointitestaus

Järjestelmätestaus

Hyväksymistestaus

2.3 Aloitus- ja lopetuskriteerit

Testauksen aloitus- ja lopetuskriteerit ovat testausvaihekohtaisia ja ne täsmennetään kunkin vaiheen testaussuunnitelmassa. Yhteisinä aloituskriteereinä ovat:

- edellisten vaiheiden testaus on suoritettu, testauksen tulokset on dokumentoitu ja testauksessa löytyneet virheet on korjattu
- alkavan testausvaiheen testauksen suunnittelu (myös testitapaukset) on tehty ja tarkastettu (sisäinen tarkastus/katselmointi)

Järjestelmätestauksen päätyttyä ennen hyväksymistestauksen alkua tehdään hyväksymistestausympäristössä ns. luovutustesti. Tässä testissä varmistetaan muutamalla testitapauksella, että koko testattava versio on siirtynyt oikein hyväksymistestausympäristöön ja että versio on siinä kunnossa, että sen hyväksymistestaus voidaan aloittaa

Järjestelmätestauksen päätyttyä, järjestelmätestauksen ja sitä edeltäneiden testausvaiheiden tulokset tarkastetaan testauskatselmuksessa.

2.4 Resurssit

testausvaiheiden vastuut kuvattuna. Sama henkilö voi hoitaa useita rooleista.

2.5 Aikataulu

3 Testiaineiston hallinta

3.1 Testauksen dokumentointi

3.2 Testausaineiston hallinta

Testiaineiston tulee olla kaikissa testausvaiheissa mahdollisimman paljon tuotantoaineiston kaltaista. Testiaineiston hallinta tulee järjestää siten, että tarvittaessa voidaan palauttaa haluttu lähtötilanne.

4 Testiympäristö ja välineet

4.1 Testiympäristöt

Testausympäristöstä täytyy keskustella.

4.2 Testausvälineet

Yksikkötestaukseen käytetään JUnit-työkalua.

5 Virheiden arvionti

Seuraavaa virheen elinkaarimallia käytetään alkaen järjestelmätestauksesta. Virheen vakavuusluokka ja prioriteetti on seuraava: Vakavuus: Pieni, Korkea ja Kiireellinen. Prioriteetti: Pieni, Korkea ja Kiireellinen.

Virheen tila on seuraava:

- New : uusi virhe raportoitu, ei ole vielä käsiteltävänä.
- Open : virhe otettu käsiteltäväksi.
- Fixed : virhe korjattu, ei ole vielä uudelleen testattu.
- Reopen : korjaus ei toimi, virhe avattu uudelleen, ei ole vielä käsiteltävänä.
- Closed : korjaus uudelleen testattu ja todettu toimivaksi. Virhe suljettu.
- Deferred : virhettä ei oteta käsittelyyn projektin tässä vaiheessa. Voidaan myöhemmin siirtää tarvittavaan tilaan testausvastaavan tai projektipäällikön toimesta.
- Rejected : raportoitu virhe ei ole virhe (virhe on testauksessa itsessään tai testipohjassa, mutta ei softassa). Voidaan myöhemmin siirtää tarvittavaan tilaan testausvastaavan tai projektipäällikön toimesta.

6 Testauksen seuranta

6.1 Palaverit

6.2 Raportointi

Jokaisen testitason loppuvaiheessa annetaan loppuraportti joka sisältää informaatiota testauksen kattavuudesta suhteessa ko. testitason tesisuunnitelmaan. Virheraportit kirjoitetaan sitä mukaa kuin ne löytyvät, virheen tilat päivitetään myös reaaliajassa eli sitä mukaa kuin virheen tila muuttuu.

6.3 Mittarit

Testauksen edistyminen:

- edellisten vaiheiden testaus on suoritettu, testauksen tulokset on dokumentoitu ja testauksessa löytyneet virheet on korjattu
- suoritettut testit vs. suunnitellut testit
- testaukseen käytetty aika (per vaihe ja per resurssi)
- löytyneet virheet vs. korjatut virheet
- löytyneet virheet osa-alueittain
- löytyneet virheet /tunti

Testauksen laatu:

- virhetiheys per. osa-alue
- hyväksymistestauksessa löytyneet virheet vs. järjestelmätestauksen virheet

7 Testaajien koulutukset

Testaajille järjestetään koulutusta ja aikaa perehtyä testaustyökaluihin, järjestelmään sekä suunnittelumenetelmiin.

Lähteet