

Tehtävä 8

Referaatti Kasper Hirvikosken kandidaatintutkielmasta Metriikat käytänteiden tukena ohjelmiston laadun arvioimisessa

Hirvikoski tutkii kandityössään ohjelmiston laadun arviointia keskittyen sisäisiin metriikoihin eli erilaisiin tapoihin mitata ohjelmiston sisäisen rakenteen laatua. Ensimmäinen esitelty menetelmä on koodikirnu, jossa mitataan ja arvioidaan ohjelmiston eri osiin kohdistuneita muutoksia valitulla ajanjaksolla. Mittaaminen kohdistuu mm. käsiteltyjen ja poistettujen koodirivien ja tiedostojen määrään suhteessa näiden kokonaismäärään, ja taustalla on ajatus siitä että tiheämmin muuttuva koodi on alttiimpaa virheille kuin koodi joka muuttuu harvemmin. Tutkimuksissa onkin havaittu ohjelmiston virhealttiuden kasvavan kun koodikirnun mittaamat suhteelliset muutosarvot nousevat.

Verkkoanalyysi tutkii puolestaan ohjelmiston eri komponenttien välisiä riippuvuuksia ja näistä muodostuvia verkostoja. Analyysi voidaan toteuttaa eri skaaloissa keskittyen esim. egoverkoksi kutsuttuun aliverkkoon, jonka avulla voidaan selvittää komponentin paikallisia yhteyksiä, tai globaaliverkkoon joka mahdollistaa yksittäisen solmun merkityksen tutkimisen koko ohjelmiston kannalta. Oletuksena on, että komponentin virheherkkyys kasvaa sen riippuvuuksien lisääntymisen myötä, ja menetelmällä pyritään paikantamaan juuri riippuvuuksiensa takia kriittisiä komponentteja.

Myös testien määrällä ja laadulla on havaittu selviä yhteyksi ohjelmiston laatuun. Testikattavuudella mitataan sitä, kuinka suuren osan koodista testit käsittävät, ja on havaittu että suurempi testikattavuus johtaa yllensä pienempään virheilmoitusten määrään julkaisun jälkeen. Kattavuus ei kuitenkaan kerro vielä testien laadusta. Tähän sopii paremmin välineeksi mutaatiotestaus, jossa luodaan testattavasta koodista mutanteja joilla tutkitaan testien kykyä havaita yleisimpiä käyttäjien tekemiä virheitä.

Hirvikoski painottaa myös kehittäjien olennaista merkitystä laadun kannalta, ja sitä kautta käsittelyyn tulevat myös ohjelmistotuotantomenetelmät. Ketterän kehityksen menetelmissä laadun hallintaa ja valvontaa painotetaan, ja se on hajautettu koko projektin kestolle ja kaikkien kehittäjien vastuulle. Ketteriin menetelmiin liittyvistä käytänteistä etenkin testilähtöisellä kehityksellä ja pariohjelmoinnilla on havaittu olevan positiivisia vaikutuksia ohjelmiston laatuun. Itsestäänselvästi myös hyvät ohjelmointikäytännöt eli laadukkaan koodin tuottamisen periaatteet edesauttavat lopputuotteen toimivuutta. Hirvikosken viittaamien tutkimusten valossa ketterä kehitys vaikuttaisikin olevan laadunäkökulmasta erittäin suositeltava ohjelmistotuotannon malli.