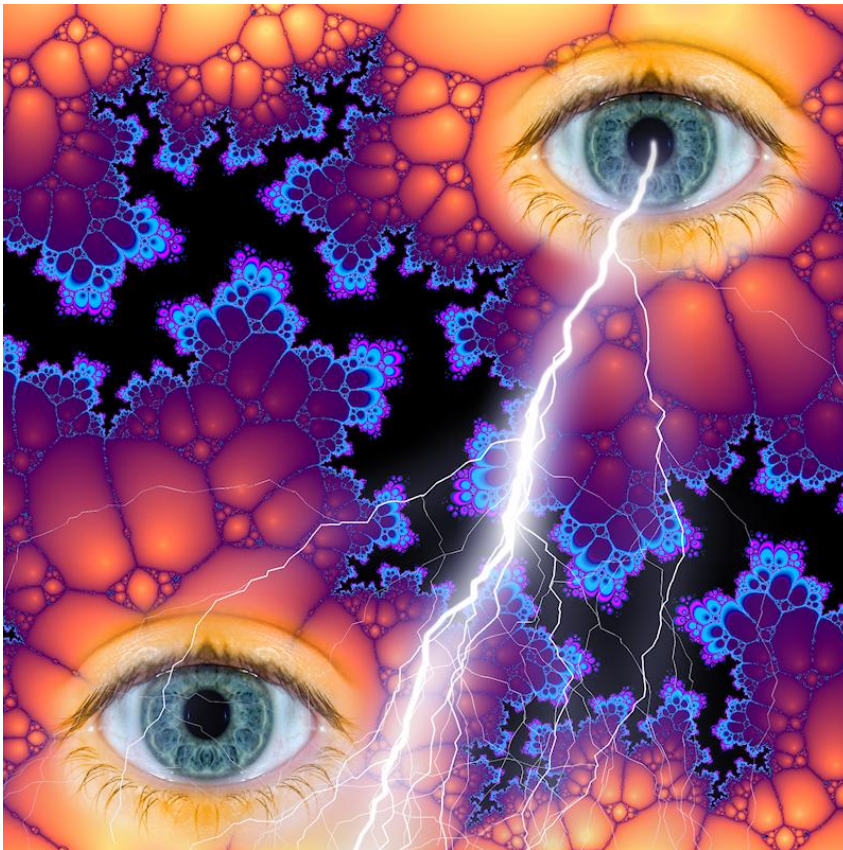


Matti Vuori

Tietopohjainen testaus ja silmien avaaminen kohteen ilmiöille

Testaus on älykästä aivotyötä. Miksiköhän siitä puhutaan niin mekanistisesti pohtimatta testaaajan aivojen käyttöä eri näkökulmista? Useimmiten testausta käsitellään tekniikkalähtöisesti ottamatta huomioon tekniikoiden edellyttämiä psykologisia tekijöitä – joihin voidaan vaikuttaa hyvällä työn ja prosessien suunnittelulla. Tässä paperissa on näkökulmana testaaajan tietoisuuden taso ja miten sen huomioon ottamisella voidaan tehostaa testausta.

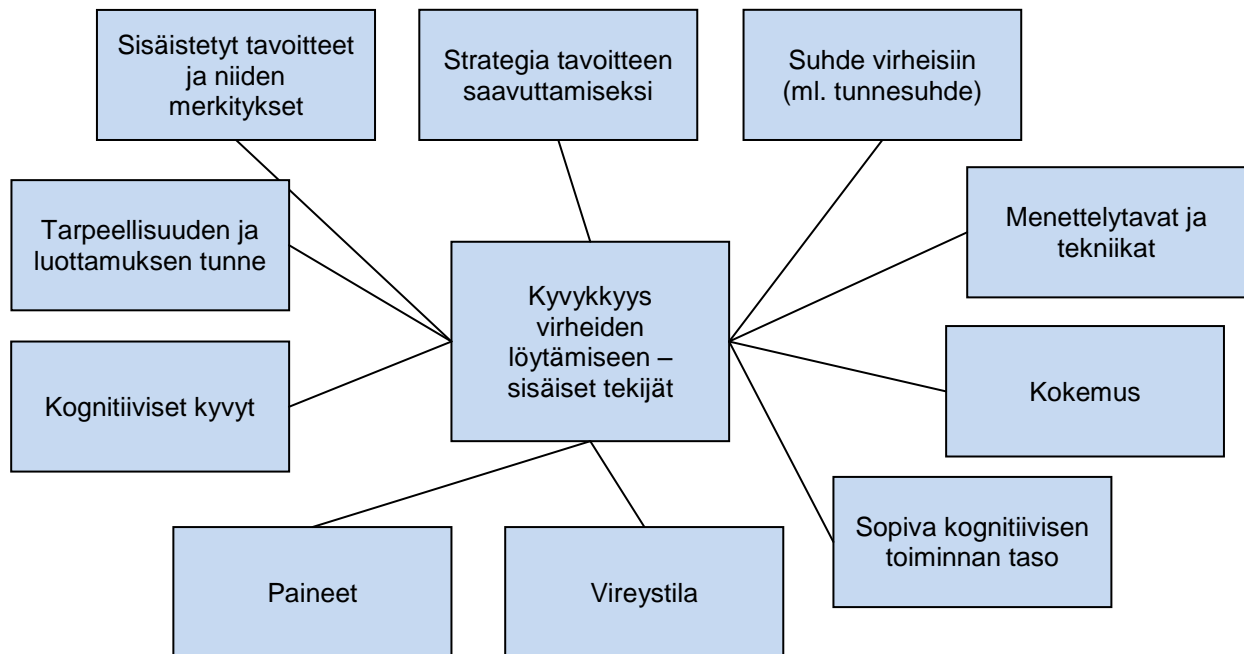


Sisällysluettelo

Miten virheitä löydetään?	2
Ymmärtämisen tavoite ja toiminnan kognitiiviset tasot	3
Välihuomautus tietopohjaisen toiminnan luonteesta	4
Tasot ovat myös oppimisen vaiheita	5
Muutama väliajatus reflektoitavaksi tässä välissä.....	5
Miten estetään ”ylirutinoituminen” ja säilytetään herkkyys?	5
Eri testityyppien keskeiset oppimisen ja ymmärtämisen kohteet	6
Lopuksi	8

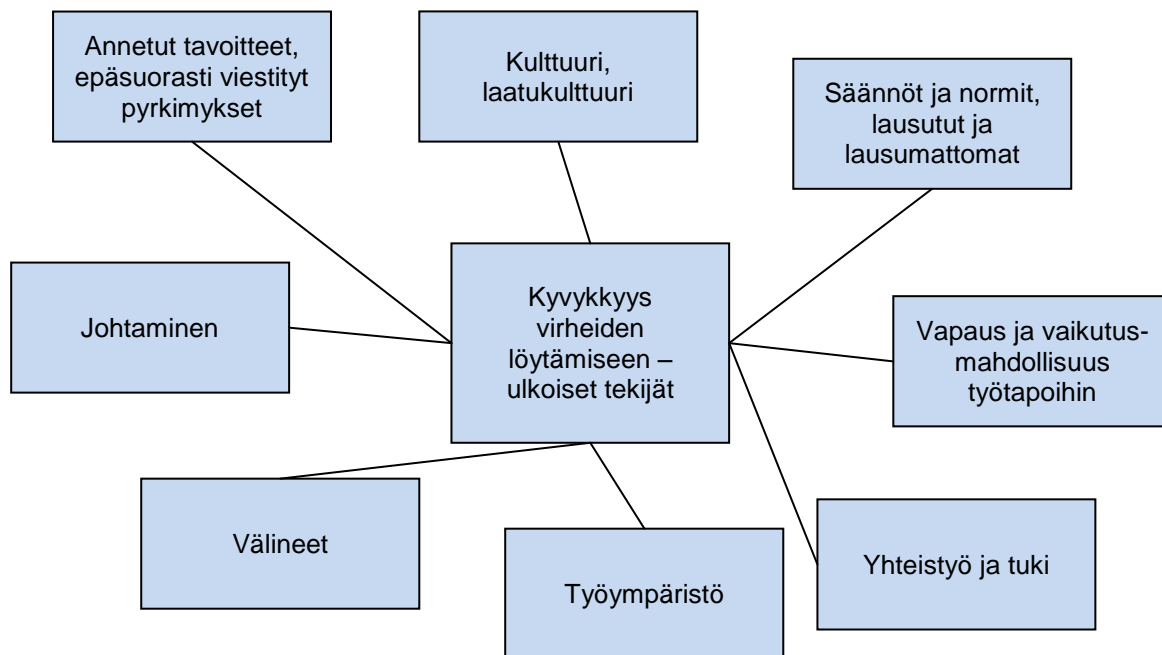
Asioita, jotka vaikuttavat virheiden löytämiseen

Etsittäessä jonkin kohteen virheitä niin toiminnallisuudessa kuin missä tahansa laatutekijöissä käytettävyydestä tietoturvallisuuteen on yleensä kyse älyllisestä ihmisten työstä. Elämme menetelmäkeskeisessä maailmassa, mutta kuitenkin kaiken keskustassa on ihmisen toiminta kohteen parissa – kohteen arviointia (vaikkapa testausta) suunnitellen, sitten arviointia suorittaen – tehden testausta, analysoiden kohdetta tai seuraten, kun joku työskentelee kohteen parissa – tai jollain muulla tavalla. Ytimessä on siis ihmisen psyykinen kyvykkyys tässä tilanteessa. Ensimmäisessä kuvassa on jäsennetty keskeisiä siihen vaikuttavia ihmisen sisäisiä tekijöitä.



Kuva 1. Sisäisiä tekijöitä kyvykkyydelle virheiden löytämiseen

Tietenkään ihminen ei ole vapaa olio, vaan keskeisiä samaan asiaan vaikuttavia tekijöitä on yhtä paljon ihmisen ulkopuolella. Seuraavassa kuvassa joitakin niistä.



Kuva 2. Ulkoisia tekijöitä kyvykkyydelle virheiden löytämiseen

Mikään näistä sisäisistä ja ulkoisista asioista ei riitä yksinään. **Mutta silti me käsittelemme tällä kertaa lähinnä yhtä sisäistä tekijää: Sopiva kognitiivisen toiminnan taso.** Hypoteesina on, että sen kautta voisimme löytää jotain uutta ja hedelmällistä. Saapa nähdä!

Ymmärtämisen tavoite ja toiminnan kognitiiviset tasot

On selvää, että arvioidakseen tai testatakseen jotain kohdetta, sitä on tarpeen ymmärtää ja sen käyttöä on syytä osata ainakin jonkin verran. Mutta onko enemmän parempi? Ei välttämättä.

Tavoite EI ole oppia testauksen kohde mahdollisimman hyvin, vaan oppia ja ymmärtää se tavalla, joka maksimoi virheiden löytämisen. Eivätkö nämä ole sama asia? Eivät valitettavasti – tai onneksi.

Ristiriita: Jos ohjelman osaa liian hyvin, testaus on kognitiivisesti ns. taitopohjaista – kuten polkupyörällä ajaminen. Joskus voidaan tietysti havahtua, ja totumuksen voima katkeaa. Jos ihmistä pyydetään selittämään, mitä tekee taitopohjaisesti – tai havahtumisen jälkeen tekemään sama tietoisesti, se ei välttämättä onnistu. Jos ihminen alkaa liikaa pohtimaan tuhat kertaa syöttämäänsä pankkiautomaatin tunnusta, se ei muistukaan...

Tällä tasolla toimii esimerkiksi regressiotestaaja, testatessaan kymmenettä kertaa jotain toimintoa.

Ja testatessa tällä tasolla:

- Toiminnan tietoisuustaso on hyvin heikko (kuin autoilijalla aamulla matkalla työpaikalleen)
- Ei nähdä kuin se, mitä on totuttu näkemään
- Ei nähdä isojakaan muita oireita ja ongelmia
- Ihmisen hahmontunnistus selittää virheetkin oikeiksi
- Ei osata kyseenalaistaa mitään
- Mieli liikkuu muissa asioissa
- Ja siksi ei löydetä bugeja!

Parempi taso on aloittelijan tietopohjainen taso, jossa jokainen tilanne käsitellään analyttisesti ilman valmiita malleja, kaikkea esillä olevaa tulkiten, avoimena kaikelle. Se on samalla taso, jolla toimii tutkiva testaja.

Näiden väliin mahtuu ns. sääntöpohjainen taso, jossa tilanteissa toimitaan tietoisesti, mutta valmiiden sääntöjen mukaisesti, tyyliin ”JOS lämpötila on alhainen, SILLOIN laitan pipon päähän ulos mennessä”. Testaus valmiiksi suunnitelluilla testitapauksilla on tällaista. Tällä tasolla on lähtökohtana se, että maailma on valmiiksi ajateltu, nyt vain seurataan niitä sääntöjä, jotka on opittu, koska ne toimivat.

Säännöt ovat sisäisiä, mielessä olevia ajattelumalleja, joita ei saa sekoittaa testajalle annettuihin kirjallisiin sääntöihin. Säännöissä on se huono puoli, että niistä valitaan ensimmäinen ”sopiva”, esimerkiksi riittävä sääntö testin hyväksymiseksi – ja usein testajan mieli toimii juuri niinpäin: haetaan perustelua testin hyväksymiseksi... Kriittistä on se, miten hyvin testaja ymmärtää, mikä on oleellista. Mutta väistämättä jää paljon tärkeitä huomioita tekemättä. Kuitenkin, silloin, kun virheitä on paljon ja ne löytyvät helposti, työ on tuloksellista.

Ja kun työssä rutinoituu, aivoja tarvitaan aina vain vähemmän ja siirrytään enemmän ja enemmän taitopohjaiseen toimintaan. Mutta joskus voidaan havahtua, kun saadaan riittävät ärsyke ja nähdä asiat uusin silmin ja palata taas tietopohjaiseen tarkasteluun. Ärsykeitä voivat olla: ihan uusi tuoteversio, käyttäjän kohtaaminen, tuoteriskianalyysi, hyvä kurssi, pitkä tauko, työskentely toisenlaisten tuotteiden parissa, jne...

Esimerkki tulostustoiminnon testauksesta eri tasoilla:

- Tietopohjainen testaus: Hmm... tässä on tulostusdialogi... mitähän kaikkea täällä on... mitähän tuo ja tuo merkitsee... jossain voinee määrittää tulostettavan sivun... mitähän sen eri valinnat tekevät ja vaikuttavat muualle
- Sääntöpohjainen testaus: Tämä on tulostusdialogi. Etsin kontrollin, jossa valitaan tulostettavaksi nykyinen sivu. Käynnistän tulostuksen. Tarkistan tuloksen tulosteesta,
- Taitopohjainen testaus: [Hyvää musiikkia!] Klikkaa Print, sitten Current page, ja Ok [Vaihda kappaletta]. Eiköhän se printti ole ihan sama kuin ennenkin.

(Näitä J. Rasmussenin 1980-luvulla kehittämiä toiminnan tasoja on käytetty jo vuosikymmeniä erilaisissa tilanteissa. Itse tutustuin niihin 1990-luvun vaihteessa inhimillisten virheiden tutkimusten yhteydessä VTT:llä.)

Välihuomautus tietopohjaisen toiminnan luonteesta

Huomautettakoon vielä yksi asia: Tietopohjainen toiminta ei varsinkaan asiantuntijatyössä merkitse suoraviivaista älykäästä” toimintaa, vaan tilanteesta riippuen yhtä tai useampaa seuraavista:

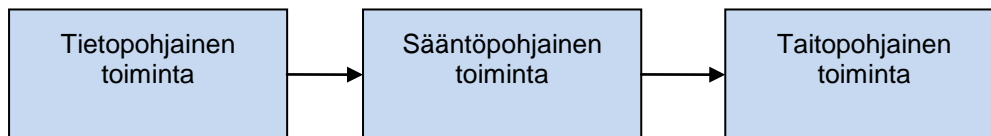
- Älyllinen ajattelu
- Ajattelu ennen toimintaa
- Kaikenlaisten havaintojen tekeminen päätöksenteon tueksi
- Ajattelun seurauksena voi olla sääntöjen noudattaminen, mutta ei välttämättä eikä toisinpäin
- Luottaminen myös intuition ja kokemuksen tuottamaan piilotietoon, mutta kyky reflaktoida sitä aidosti
- Kokonaisuuden monimuotoinen ja moniparadigmainen ajattelu

Lisäksi, termi ei ota kantaa esimerkiksi kokeilun määrään. Asioita voidaan kokeilla tai ei. Se ei vaikuta toimintamme kognitiiviseen tasoon. Mutta!

- Kokeilu auttaa huomaamaan, miten vanhat säännöt toimivat tai eivät toimi.
- Ristiriitaisesti kokeilu myös suosii alempia toiminnan tasoja, jos vanhat säännöt näyttävätkin toimivan...
- Kokeilu on aina hyvä tapa tarkentaa hypoteeseja.
- Useimmissa tapauksissa suositteluvani järjestys on se, että ensin pitää miettiä ja sitten tehdä eli kokeilla eli testata, sillä ajattelun ja analyysin kautta saadaan kuva siitä, mitä pitää tai kannattaa selvittää kokeellisesti ja mihin asioihin pitää kiinnittää huomiota.

Tasot ovat myös oppimisen vaiheita

Nämä tasot ovat myös oppimisen vaiheita. Aloittelija toimii tietopohjaisesti (autoilija opetetaan liikenteen logiikkaa). Vähitellen omaksutaan sääntöjä (liikennesäännöt, omat säännöt) ja rutiinin karttuessa aletaan toimia automaatti päällä.



Oppiminen —————>

Kuva 3. Oppiessaan ja rutiointuessaan ihminen siirtyy toimimaan enemmän taitopohjaisesti, ilman tietoista kontrollia.

Muutama väliajatus reflektoitavaksi tässä välissä

Mieti, mitä nämä tarkoittavat:

- Testaajan mieli = aloittelijan mieli
- Mielen pitäminen avoinna = keskittymätön keskittyminen
- "Curiosity killed the bug"
- Kuuluisassa videossa seurataan koripalloa. Havainnoijien tehtävä on laskea, kuinka monta kertaa pelaajat heittävät pallon toiselle pelaajalle. Se laskenta onnistuu oikein hyvin. Mutta! Samaan aikaan pelikentällä kävelee hyvin näkyvästi gorillapukuun pukeutunut henkilö. Erittäin moni havainnoija ei huomaa häntä lainkaan, koska keskittyy syöttöjen laskemiseen! Mistä tämä kertoo?

Miten estetään "ylirutinoituminen" ja säilytetään herkkyys?

Olellainen kysymys on nyt se, miten tuetaan toimimista enemmän tietoisilla tasoilla. Tähän haasteeseen liittyy myös riskejä, sillä silloin on vaarana toimia liikaa abstraktioiden tasolla. Tarvitaan sellainen keinovalikoima, jossa tiettyä asia parannetaan sekä keskittyvillä että näkökulmia avaavilla keinoilla.

Seuraavassa on muutamia mahdollisia keinoja.

- Kohteen ja testauksen tavoitteiden ymmärtäminen
 - Tutustutaan konseptiin / tuoteryhmään mahdollisimman erilaisilla ohjelmilla kuin se, jota testataan
 - Tuetaan tietoisuutta tarkastelemalla kohdetta mallien kautta
 - Pehdytään kohteen laatuominaisuuksiin ja ei-toiminnallisiin vaatimuksiin, koska ne auttavat "näkemään sen, mitä ei ole näkyvässä"
 - Tuoteriskien tunnistaminen ja toimintojen ja ominaisuuksien priorisointi auttaa pysymään skarppina
- Testauksen prosessi ja työnkulut
 - Sopivaan kohtaan työnkuluja tuodaan "mielen avaamisen" vaihe. Esimerkiksi testaamisessa on erilaisia alustustehtäviä (testiympäristön valmistelu yms.), joiden on tärkeää tapahtua taitopohjaisesti ja mahdollisimman tehokkaasti. Mutta sen jälkeen pitää vaihtaa "uudelle vaihteelle", laittaa "toinen hattu päähän", ja siis siirtyä toiselle toiminnan kognitiiviselle tasolle. Mitä keinoja tähän siirtymiseen voi käyttää, riippuu kulttuurista

- Testausprosessin vaiheille suunnitellaan tietoisesti tietty kognitiivisen toiminnan taso. Esimerkiksi edellämainitut alustustyöt suunnitellaan siten, että tavoitteena on sellainen taitopohjaisuus, että testaajan ei pidä juurikaan ajatella. Tämä on työprosessin suunnittelun lähtökohta, joka varmistetaan ja katselmoidaan. Ja vastaava kaikille prosessin vaiheille. Olennaista on siis tietoinen tasotavoite ja sen varmistaminen.
- Rikotaan valmiit suorituspolut erilaisilla roolien ja ajattelumallien ”hatuilla” (astutaan aloittelijan saappaisiin, ollaan kuin tehokäyttäjä, tai kuin lapsi jne...)
- Tuodaan testausmenetelmiin tai testiympäristöön mahdollisimman paljon variaatioita, mikä pitää testaajan ”varpaillaan”
- Testaustyylien muu vaihtelu
- Työnsuunnittelu
 - Samaa käyttöliittymää ei testata pitkiä jaksoja ilma taukoja
 - Tehtävankierto, toimiminen eri testausalueilla, eri järjestelmien parissa
- Reflektio ja kehittyminen
 - Ajattelun ulkoistaminen selittämällä / kirjoittamalla toisille, miten toimii
 - Käytetään ideointitekniikoita prosessin, jopa useissa prosessin vaiheissa
- Organisaatio ja kulttuuri
 - Tuetaan organisaatiokulttuurin intellektuaalisuutta
 - Tehdään jatkuvaa parantamista, koska muuten taannutaan ja taantuessa aina ajattelu huononee
 - Heterogeeniset, erilaisista ihmisistä koostuvat tiimit ovat vähemmän alttiita yhteiselle toiminnan tason rapautumiselle ja ”laitostumiselle”
- Jne...

Eri testityyppien keskeiset oppimisen ja ymmärtämisen kohteet

”Testausta” käsitellään usein yhtenä ja yhtäläisenä asiana (kuten tässäkin paperissa!), mutta oikeasti silloin puhutaan useimmiten käyttöliittymätason toiminnallisuustestauksesta. Testauksen maailma monimuotoistuu koko ajan ja siksi pitää yhä useammin olla tarkka siitä, mistä oikeasti puhutaan, sillä eri testityyppien suunnittelijan ja tekijän osaamistarpeet ovat erilaiset.

Pointti:

On hyvä miettiä, mitä kaikkea testauksen kohteesta ja kontekstista pitää tietää, sillä tämä tietämys on keskeisimpiä asioita, jotka estävät testaajan tietoisuuden tilan alenemisen ja antavat mahdollisuuden tietopohjaiselle testaukselle (silloin, kun sitä tarvitaan).

Siksipä seuraavassa taulukossa on hahmoteltu eri testityyppien osaamisalueprofiilia. Huomattakoon, että se on subjektiivinen ja perustuu ajatuksiin siitä, miten asioita pitäisi tehdä, eikä nykyiseen vallitsevaan tapaan.

Taulukko 1. Eri testityyppien osaamistarpeita (yksinkertaistettu kärjisty...)

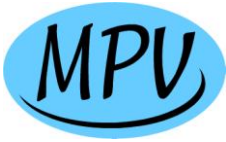
Testityyppi	Konteksti ja käyttö					Suunnittelu ja tekniikka					Prosessit		
	Kulttuuri ja konteksti, käyttäjät	Käyttötarkoitus ja bisnes	Käytettävät, työtehtävät, prosessit	Käyttöpaukset	Sujuva käyttö	Erlaiset käytettävät	Käytettävät toiminnot	Tietovirrat	Sis. arkkitehtuuri	Ulk. arkkitehtuuri	Toteutustekniikat	Keh. projektn itoiminta	Muu testaus ja laadunvarmistus
Järjestelmätason toiminnallisuustestaus	**	***	***	***		***	***	***		***		***	***
Tietoturvatestaus, tietoriskien analysointi ja tietoturvallisuuden analysointi	***	***	***	***		***	***	***	***	***	***		
Käytettävyydestä ja – analyysi eli kaikki k:n varm.	***	***	***	***	***	***	***			(**)			
Regressiotestaus (j-taso)		**	***	***	***		***	***		***	*	*	*
Kuormitustestaus		***	**	***				***	**	***	***		
Järjestelmäintegroitustestaus		***	***	***			**	***		***		***	***
Matalan tason integroitustestaus						***		***	***		***	***	***
Yksikkötestaus								**	***		***	***	**
Hyväksymistestaus	***	***	***	***	***	***	***	***	(***)	***			

Merkinnot:

- *** Erittäin tärkeä osaamis/ymmärtämisalue
- ** Kohtalaisen tärkeä osaamis/ymmärtämisalue
- * Vähän tärkeä osaamis/ymmärtämisalue

(Lisäbonus: Kaavio antaa myös ajatuksia siihen, miten testaajien siirtyminen toisiin testustehtäviin onnistuu: tiedollisesti haastavimmista ja laajemmista on helpompi siirtyä suppeampiin kuin toisinpäin. Tietenkin tässä asiassa pitää ottaa huomioon muitakin näkökohtia.)

Varoitus! Ihmiset tuntien huomautettakoon tähtien suhteen, että niitä ei voi laskea sellaisenaan yhteen, koska eri tähtien painotus ei ole lineaarinen ja ”määrät” ovat oikeasti laadullisuuksia. Käytännössä tällaisia taulukoita on laskettu yhteen mm. QFD-menetelmässä ja silloin *** = 9, ** = 3 ja * = 1. Itsekin olen tehnyt sellaisen QFD-työkalun, joka kaivetaan naftaliinista, kun QFD palaa taas muotiin.)



Lopuksi

Pohtimalla testaajan kognitiivisia tasoja voidaan selvästi huomata sellaisia asioita, jotka ilman tätä käsitteellistä mallia jäisivät pimentoon. Testauksen toimintajärjestelmän kokonaisymmärrys on muutenkin kasvanut niin rikkaaksi ja työn luonnetta on alettu ymmärtää niin paljon, että vastaavanlaisia tarkasteluja voi jatkaa ja löytää uusia kehittymisen mahdollisuuksia kansallisestikin tärkeän ”älykkään testauksen” alueella.

Pitänee muistuttaa, että tämä on vasta ”testaavan subjektin” ensimmäinen taso. Tiimin toiminta, rakentaminen ja tiimidynamiikka ovat ihan yhtä tärkeitä asioita, laajemmista ihmisten organisaatioista puhumattakaan.