
Testausprosessin kehittäminen Neocodex Oy:ssä

Reijo Kauppinen

Opinnäytetyö

Ammattikorkeakoulututkinto



Koulutusala Insinööri			
Koulutusohjelma Tuotantotalous			
Työn tekijä(t) Reijo Kauppinen			
Työn nimi Testausprosessin kehittäminen Neocodex Oy:ssä			
Päiväys	30.10.2013	Sivumäärä/Liitteet	31/1 liite
Ohjaaja(t) Jarmo Pyysalo			
Toimeksiantaja/Yhteistyökumppani(t) Neocodex Oy			
Tiivistelmä			
<p>Tämän opinnäytetyön aiheena oli kuvata Neocodex Oy:n laadunparantamisprojektin aloitusvaihetta, keskittyen testausprosessin kehittämiseen. Neocodex Oy:n suunnitelmissa on ottaa käyttöön ISO 9001-standardin vaatimukset täyttävä laatujärjestelmä vuoden 2013 aikana. Työssä keskityttiin laadun määrittämiseen ohjelmistotuotantoyrityksen kannalta ja erityisesti testaukseen, joka on tärkeä osa Neocodex Oy:n jokaista pääprosessia.</p> <p>Työssä on kuvattu Neocodex Oy:n pääprosessit. Testauksen nykytilanne kartoitettiin työntekijöille tehdyn kyselyn avulla ja kehitystarpeet tuotiin esille. Koska laatujärjestelmän rakentaminen oli Neocodex Oy:ssä tämän työn tekovaiheessa vielä suunnitteluasteella, sen tarkoituksena oli pohjustaa ja antaa kehitysideoita laatujärjestelmän kokoamista varten.</p> <p>Työ koettiin Neocodex Oy:ssä erittäin hyödylliseksi, koska yrityksellä ei ollut selkeää kokonaiskuvaa laatujärjestelmästä.</p>			
Avainsanat Testaus, laadunparantaminen, prosessit, laatu			
Julkinen			

Field of Study Technology, Communication and Transport			
Degree Programme Industrial Management			
Author(s) Reijo Kauppinen			
Title of Thesis Development of the Testing Process in Neocodex Oy			
Date	30.10.2013	Pages/Appendices	31/1 appendice
Supervisor(s) Jarmo Pyysalo			
Project/Partners Neocodex Oy			
<p>Abstract</p> <p>The purpose of the present thesis was to outline the start of a quality system project, focusing on the improvement of the testing process. The quality system will be implemented in Neocodex Oy according to the ISO 9001 standard's requirements during 2013. The present thesis focused on defining quality from the point of view of a software engineering company, and on testing, which is an important part of Neocodex Oy's every main process.</p> <p>The thesis describes Neocodex Oy's main processes. The present state of testing was studied through a survey carried out among the employees, and the development needs were brought out.</p> <p>When writing this thesis, the building of the quality system was still in the early stages in Neocodex Oy. The purpose of this thesis was to outline the quality system project and to give development ideas for it.</p> <p>Neocodex Oy found the thesis very useful, because the company had no overall picture of the quality system.</p>			
Keywords Testing, quality improvement, processes, quality			
Public			

Sisällysluettelo

1 JOHDANTO	7
1.1 Projektin lähtökohdat ja tavoitteet.....	7
1.2 Neocodex Oy.....	8
2 LAADUN MÄÄRITELMÄ	9
2.1 Laatu ohjelmistotuotannon kannalta.....	9
2.2 Laadunhallintajärjestelmä	11
2.3 Laatupolitiikka	12
2.4 ISO 9000-standardi	12
3 TESTAUS	14
3.1 Testauksen lähtötilanne ja kartoitus	14
3.2 Testauksen yleiset periaatteet.....	15
3.3 Testaustasot	17
3.4 Testausprosessin kuvaus.....	19
3.5 Kehittämiskohdat.....	22
4 NEOCODEX OY:N PROSESSIT JA NIIDEN KEHITTÄMINEN	23
4.1 Prosessijohtaminen.....	23
4.2 Projektitoimitukset	24
4.3 Ylläpitopalvelut.....	26
4.4 Tuotekehitys	27
4.5 Asiakkuuksien hallinta.....	28
5 LAADUNPARANTAMINEN OHJELMISTON KEHITYSPROSESSISSA	29
5.1 Johdon tehtävä laadunparantamisessa.....	29
5.2 Jatkuva parantaminen.....	30
6 YHTEENVETO JA POHDINTA	31

LÄHTEET

LIITTEET

1 JOHDANTO

1.1 Projektin lähtökohdat ja tavoitteet

Tässä työssä keskityttiin pääasiallisesti Neocodex Oy:n testausprosessiin, koska se liittyy tärkeänä osana yrityksen projektitoimituksiin, ylläpitopalveluihin ja tuotekehitykseen.

Testausprosessin periaatteiden määrittely oli tärkeää siinä vaiheessa, kun pääprosessit olivat saaneet nimensä ja niiden sisällöt tulevat vuoden 2013 aikana saamaan tarkemmat kuvaukset kaikkine niihin liittyvine prosesseineen.

Tämän projektin aikana asiakkuuksien hallinta ja johtaminen voitiin kuvata yleisellä tasolla. Asiakkuuksien hallinta tarkoittaa markkinointiin, toimitusketjun hallintaan, yleensä ulkoisiin (ostavat, toimittavat asiakkaat) ja sisäisiin asiakkaisiin (oma henkilöstö) liittyviä toimintoja, joilla tavoitetaan asiakkaat Neocodex Oy:n visiolla.

Johtaminen mahdollistaa yrityksen toiminnan, yhdistämällä tuotannon panokset muun muassa työn, pääoman, koneet ja laitteet siten, että liiketoiminnan tavoitteet saavutetaan. Laatujärjestelmän kehittäminen on liiketoimintajärjestelmän kehittämistä tuottamaan lisäarvoa asiakkaalle ja antamaan mahdollisimman hyvän taloudellisen tuloksen.

Testauksen merkitys ohjelmistoyritykselle yhtenä asiakasarvoa tuottavana prosessina, on keskeinen juuri sen vaikuttavuuden takia. Se liittyy kolmeen tärkeään asiakkaalle arvoa tuottavaan prosessiin, jotka ovat projektitoimitukset, ylläpitopalvelut ja tuotekehitys. Insinööriyön tarkoituksena on avata tärkeä testausprosessi ensimmäisten tehtävien joukossa, jotta menestyksellisempään työhön päästään jo laatujärjestelmän rakentamisen alkuvaiheessa. Se mahdollistaa liiketoiminnan tuloksellisuuden parantamisen samalla kun yritys panostaa oman toimintansa kehittämiseen. Usein kehittäminen tarkoittaa Neocodex Oy:ssä sitä, että kehittämiseen käytetty aika on poissa laskutettavasta työstä. Siksi ulkopuolinen apu, konsultteineen, rahoituksineen ja insinööriyöntekijöineen on tärkeä osa yrityksen liiketoiminnan laadun parantamista.

ISO-9000 järjestelmän prosessiajattelussa lähdetään liikkeelle asiakkaasta ja hänen tarpeistaan. Asiakstarpeet pyritään tyydyttämään asiakastarpeiden mukaisilla tuote- ja palveluratkaisuilla. Yrityksessä täytyy suunnitella toimenpiteet ja resurssit, joilla

nämä tarpeet saadaan tyydytettyä kilpailukykyisin hinnoin. On tärkeää että prosessien suunnittelussa onnistutaan hyvin, koska se lisää asiakastyytyväisyyttä ja organisaation työntekijät työskentelevät itseohjautuvasti ja ymmärtävät prosessien tuoman lisäarvon asiakkaalle. (Laamanen, 2001, 21-22.)

1.2 Neocodex Oy

Neocodex Oy on asiantuntijayritys, joka on erikoistunut teollisuuden tiedonkeruuseen ja raportointiin, sekä alihankintana tehtävään ohjelmistokehitykseen. Yhtiön toimipisteet sijoittuvat Varkauteen ja Kuopioon. Yrityksellä on myös tytäryhtiö Caldinari Oy, jonka vahvoihin osaamisalueisiin kuuluu voimalaitoskattiloiden prosessitiedon jalostus. Yrityksellä on vuosien kokemus prosessitiedon analysointi- ja visualisointijärjestelmistä. Henkilöstöllä on vahva ammattitaito asiakkaiden tarpeita vastaavien järjestelmien suunnitteluun, toimitukseen ja ylläpitoon. Raportointi, analyysilaskennat ja prosessinäytöt toteutetaan AWR- raportointijärjestelmällä jo olemassa oleviin sekä uusiin prosessitietojärjestelmiin. Yritys tarjoaa teollisuustietokantojen asennuksen ja ylläpidon sekä analytiikkapalvelinten ylläpidon. Rääätälöidyt ohjelmistoratkaisut asiakkaan yksilöllisiin tarpeisiin kuuluvat yrityksen palveluntarjontaan. Yrityksen tärkeimmät asiakkaat ovat Honeywell Oy, ABB Oy, Foster Wheeler Energia Oy ja Caldinari Oy. (Neocodex Oy:n kotisivut)

Ohjelmointityössä yritys käyttää yleisimpiä lausekieliä ja ohjelmointialustoja. Jokainen ohjelmistoprojekti suunnitellaan yksilöllisesti vastaamaan asiakkaan tarpeita. Muuttuvassa toiminta- tai ylläpitoympäristössä päivitystä vaativat ohjelmistomuutokset kuuluvat palveluihin. Ohjelmistosuunnittelijat toimivat tarvittaessa yrityksen omien tuotekehitys- tai projektiorganisaatioiden osana asiakkaan omissa tiloissa. Tämän seurauksena suunnittelijoiden osaamista kehitetään pitkäjänteisesti asiakkaiden tarpeiden mukaan ja suunnittelijat ovat käytettävissä asiakkaan kanssa yhteisesti sovituina ajanjaksoina. Raportointiprojektit yritys toimittaa avaimet käteen periaatteella. Raportointiprojekteissa on ajatuksena auttaa asiakasta optimoimaan tuotantoprosessiaan niin laadullisesti kuin taloudellisestikin samalla vähentäen myös mahdollista ympäristökuormitusta. (Neocodex Oy:n kotisivut)

Yritys järjestää omiin tuotteisiin liittyvää käyttö- ja ylläpitokoulutusta. Koulutusta on saatavilla myös tietokantojen hallintaan, tietokantaohjelmointiin, raportointiin, räätälöityihin ohjelmointikursseihin ja Web-ohjelmointiin. (Neocodex Oy:n kotisivut)

2 LAADUN MÄÄRITELMÄ

2.1 Laatu ohjelmistotuotannon kannalta

Ohjelmistoalan ammattilaiset pitävät itsestäänselvyytenä, että käytettävä ohjelma sisältää tietyn määrän virheitä ja sen vuoksi ohjelman käyttö ei ole aina vaivatonta. Tavallinen käyttäjä ei tiedä virheiden olemassa olosta niin kauan, kun ne eivät häiritse ohjelman käyttämistä. Ohjelmistotuotteen myyminen eroaa huomattavasti esimerkiksi auton mynnistä. Ohjelmien käsikirjoissa saatetaan luetella ohjelmiston tunnetuimmat ohjelmiston viat ja tavat miten kiertää ne. Sama ei tulisi kuuloonkaan autojen myynnin yhteydessä. Kuka ostaa autoa, jonka käyttöohjeissa on lista sen vioista. (Haikala & Märijärvi, 2004, 192.)

Laatutermi tulkitaan useasti monella tapaa. Kun puhutaan laadusta yleisesti, hyvä laatu viestii asiakkaalle yleensä positiivisia asioita. Ohjelmistotuotannossa laatu tarkoittaa yleensä ohjelman virheettömyyttä ja ajan tasalla olevia käyttökelpoisia dokumentteja. Hyvää laatua on myös ohjelman hyvä suorituskyky ja luotettavuus käytön kannalta. (Haikala & Märijärvi, 2004, 191-192.)

Tuote hankitaan täyttämään jokin tietty tarve, jonka vuoksi tuotteen ominaisuudet ovat jo osa laatumielikuvaa. Laadukas tuote yleensä myös yllättää käyttäjänsä positiivisesti. (Haikala & Märijärvi, 2004, 191-192.)

Yrityksen näkökulmasta pelkän laadukkaan tuotteen tekeminen ei ole ainut asia mihin täytyy panostaa, sillä taloudellinen näkökulma pitää ottaa myös huomioon. Korkealaatuisia tuotteita valmistavan yrityksen täytyy olla myös taloudellisesti kannattava. Laadun jatkuva ylläpitäminen on yritykselle haasteellinen tehtävä, koska kilpailu ja asiakkaan muuttuva laatumielikuva tuotteen kannalta muuttuu jatkuvasti ympäristön vaatimuksien painostamana. Tästä voi pitää hyvänä esimerkkinä esimerkiksi puhelimien kehitystä viimeisen kymmenen vuoden aikana. (Haikala & Märijärvi, 2004, 191-192.)

Haikalan ja Märijärven (2004, 193) mukaan *"ISO-standardeissa laatu määritellään siten, että laadukas tuote täyttää sille asetetut vaatimukset, sekä toiminnalliset että ei-toiminnalliset"*. Toiminnallisilla laatuvaatimuksilla tarkoitetaan ohjelmistotuotannossa yksinkertaisesti ohjelmiston toimintoja. Ohjelmiston nopeutta ja luotettavuutta etsi-

tään ei- toiminnallisten piirteiden kautta. Wesseliuksen ja Ververin mukaan laadun määrittämisen voi jakaa osakomponentteihin, joita ovat objektiivisesti arvioitavat komponentit, subjektiivisesti arvioitavat komponentit ja kokonaan arvioimattomaan komponenttiin. (Haikala & Märijärvi, 2004, 191-194.)

Objektiivisessa laadun määritelmässä arvioidaan sitä miten tuote täyttää sille vaatimusmäärittelyssä asetetut vaatimukset. Objektiivisesta tuotteesta pystytään mittaamaan ja laskemaan hyvin sille asetetut vaatimukset. Tuotteen valmistukseen liittyviä objektiivisesti arvioitavia määreitä ovat myös projektin aikataulussa ja budjetissa pysyminen sekä ohjelmiston virheet. (Haikala & Märijärvi, 2004, 194.)

Asiakkaalla on tietyt odotukset tuotteen laadusta, jota pystytään arvioimaan subjektiivisen komponentin kautta. Jos tuote täyttää asiakkaan vaatimukset, subjektiivisessa laatukomponentissa tarkastellaan sitä miten hyvin vaatimukset on kirjattu ja toteutettu, eli miltä tuote asiakkaan mielestä tuntuu. Objektiivisesti ei pystytä määrittelemään miltä tuote tuntuu tai miten se täyttää asiakkaan odotukset, joten subjektiivinen laatu-komponentti on välttämätön. Kyselytutkimus on hyvä tapa toteuttaa subjektiivinen mielikuva tuotteesta, koska silloin saadaan selville miten hyödylliseksi ja helppokäyttöiseksi asiakas tuotteen määrittelee. (Haikala & Märijärvi, 2004, 194.)

Kolmas laatukomponentti on arvioitavissa vasta tulevaisuudessa, koska sillä pyritään arvioimaan vielä ennalta arvaamattomia asiakkaan tarpeita ja toiveita, sekä sitä miten tuote käyttäytyy ohjelmiston ennalta arvaamattomissa virhetilanteissa. Tämän laatu-komponentin ennalta arvaamattomiin tilanteisiin ohjelmistotuottaja pystyy varautumaan vain osittain esimerkiksi hyvällä ja huolellisella suunnittelulla. (Haikala & Märijärvi, 2004, 194-195.)

Yhteenvedona voi kuitenkin todeta, että laadun määrittäminen on aina subjektiivista, koska mittauksista huolimatta mittaustulokset ovat aina suhteellisia. Asiakas mittaa aina kuitenkin laadun määrän mittalukuarvoa tai vaikutelmaa muihin vastaaviin tuotteisiin, omiin tarpeisiin ja mielikuviinsa. (Haikala & Märijärvi, 2004, 195.)

Jos laatua tarkastellaan liiketaloudellisista näkökulmista, laatu on sitä, että asiakas ja toimittaja ovat tyytyväisiä. Ohjelmistotuotannossa noudatetaan liiketaloudellisia periaatteita ja asiakaslähtöistä toimintaa, koska ohjelmistoa tehdessä tavoitteena tulisi olla asiakkaan tarpeiden täyttäminen ja kokonaisratkaisun sopivuus asiakkaalle. Budjetissa ja aikataulussa pysyminen on yksi tärkeä tavoite ja osa kokonaisratkaisun

onnistumista asiakkaan näkökulmasta. Jokaisella alalla on omat tekijänsä, mitkä vaikuttavat asiakastyytyväisyyteen ja laatumielikuvaan, nämä kannattaa ottaa selville. Asiakastyytyväisyyteen vaikuttavat monet muutkin asiat kun tuotteen laatu, hinta ja sen toimivuus, esimerkiksi tuotteen myynnin yhteydessä muuta lisäarvoa tuottavat asiat, kuten asennus-, konsultointi-, koulutus-, käyttö ja ylläpitopalvelut. (Haikala & Märijärvi, 2004, 195-196.)

2.2 Laadunhallintajärjestelmä

Ennen vuotta 2000 laadunhallintajärjestelmä tunnettiin nimellä laatujärjestelmä, joka määritellään ISO 9000 standardissa seuraavasti: *”Johtamisjärjestelmä, jonka avulla suunnataan ja ohjataan organisaatiota laatuun liittyvissä asioissa”*. Haikalan ja Märijärven teoksessa Ohjelmistotuotanto (Haikala ja Märijärvi, 197) kuvataan edellä mainittua standardia sanoin: *”Standardin mukaan johto kykenee kohottamaan laatutasoa ja asiakastyytyväisyyttä seuraavilla periaatteilla: asiakaskeskeisyys, johtajuus, henkilöstön osallistuminen, prosessimainen toimintamalli, järjestelmällinen johtamistapa, jatkuva parantaminen, tosiasioihin perustuva päätöksenteko ja molempia osapuolia hyödyttävät suhteet toimituksissa”*. (Haikala ja Märijärvi, 197) Laadunhallinta pitää myös sisällään laadunohjauksen, laadunvarmistamisen, laadunsuunnittelun ja laadun parantamisen.

Laadunohjauksella valvotaan sitä, että laatuvaatimukset täyttyvät sillä tekniikalla ja niillä toimilla mitkä on standardissa määritelty. Tuotteen valmistukseen tai asiakaspalveluun kuuluu yleensä tietty ylimmän johdon määrittelemä prosessi, josta laatujärjestelmä muodostuu. Ohjelmistotuotantoa ajatellen työprosesseihin liittyy organisaation rakenne, vastuut, menettelyohjeet, työympäristö ja resurssit. Tuottavuuteen ja laatuun vahvasti ohjelmistotalalla vaikuttava tekijä on edellä mainittu työympäristö. Laatujärjestelmä joko dokumentoituna tai dokumentoitumattomana, voidaan määritellä yrityksen tavaksi toimia kaikilla toiminnan osa-alueilla.

Ohjelmistotuotanto on näkymätöntä, joten on erityisen tärkeää, että työntekijät noudattavat hyviä toimintatapoja ja ohjelmistotuotteen valmistukseen käytettävää toimintaprosessia, jotta saadaan mahdollisimman hyvä lopputulos. On myös modernia ajatella että jokainen työntekijä on itse tietyn valtuuksin ja velvollisuuksin vastuussa työnsä laadusta. Harvoin tiukka valvonta ja ohjaus sopivat suomalaisen ammattiyhdyksen rinnalle. Kun työ toteutetaan noudattamalla tiettyä toimintaprosessia voidaan prosessien eri vaiheita seurata ja mitata tarvittaessa. Näin ongelmiin päästään puut-

tumaan ja sitä kautta kehittämään yrityksen laaduntuottokykyä. Pelkkä prosessiajattelu ei kuitenkaan korvaa työntekijöiden ammattitaitoa, siksi hyvällä johtamisella saadaan myös paljon tuloksia aikaan. (Haikala & Märijärvi, 2004 197-199.)

Lecklinin (2002) mukaan ”*laatujärjestelmä voidaan määritellä rakenteeksi, jonka avulla johdon tahtotila viedään systemaattisesti läpi koko organisaation*”. Laatujärjestelmän tavoitteet ovat erilaisia yrityksistä riippuen, jonka vuoksi siitä ei kannata tehdä liian raskasta, jolloin sitä on hankalaa toteuttaa päivittäisessä työssä. Lecklin pitää myös prosessiajattelua tärkeänä osana laatujärjestelmää. (Lecklin, 2002, 31-33.)

2.3 Laatupolitiikka

Johdon tulee luoda perusta laatutavoitteiden asettamiselle ja toteutukselle. Sen täytyy asettaa asiakkaiden tarpeisiin ja vaatimuksiin soveltuva yrityksen laatupolitiikka. Laatupolitiikka on saatettava tiedoksi koko henkilöstölle ja varmistettava että se ymmärretään. Johdon on seurattava, että laatupolitiikkaa toteutetaan koko yrityksessä ja seurattava sen toteutumista ja kehitettävä sitä tarpeen vaatiessa. Lyhyt ja ytimekäs laatupolitiikka on kaikkein tehokkain. (Lecklin, 2002, 43.)

Yrityksissä on usein käytössä ISO-standardin vaatimusten mukaan kirjattu laatupolitiikka, joka standardissa määritellään sanoin: ”*Johdon julkituoma laatuun liittyvä organisaation yleinen tarkoitus ja suunta. Laatupolitiikka muodostaa puitteet laatutavoitteiden asettamiselle*”. Yrityksen laatujärjestelmän toiminnalla pyritään johdon laatupolitiikassa asettamaan tavoitetila, jonka seurauksena käynnistyy laadunkehityshankkeita, jotta päästään haluttuun lopputulokseen. (Haikala & Märijärvi, 2004, 199.)

2.4 ISO 9000-standardi

ISO 9000-standardisarja on saanut alkunsa 1960-luvun sotilasstandardeista ja sen tarkoituksena oli parantaa teollisuuden kilpailukykyä. British Standards Institutin kehittämät BS-standardit toimivat ISO-standardin kehittäjien pohjana. ISO 9000 standardisarjaa käytetään paljon ohjelmistoalalla, vaikka se on alunperin suunniteltu teollisuusaloille. ISO 9000-standardisarja pitää sisällään kolme varsinaista standardia. ISO 9000-standardi sisältää yleiset periaatteet ja termistön. ISO 9001 määrittää organisaation laatujärjestelmän vaatimukset, sertifiointia tutkittaessa tarkastetaan täyttääkö yritys laatujärjestelmän vaatimukset. ISO-9004 on pelkästään soveltamisohje,

joka tukee yritystä, kun luodaan jatkuvaan kehittämiseen perustuvaa laadunhallintajärjestelmää. Ohjelmistoalan yritysten sertifiointi toteutetaan seuraamalla yrityksen toimintatapojen yhdenmukaisuutta ISO-9001-standardin mukaisesti. (Haikala & Märijärvi, 2004, 211-212.)

ISO-9001 sisältää laatujärjestelmän määritelmät, jotka ohjaavat yritystä toiminnoissa, sertifiointissa sekä sopimuksien laatimisessa. ISO-9001 mukaan *"kansainvälisen standardin tarkoituksena ei ole se, että kaikilla laadunhallintajärjestelmillä täytyy olla yhdenmukainen rakenne tai asiakirjat, vaan prosessimaisella toimintamallilla pyritään tehostamaan yrityksen toimintoja"*. Kun yrityksen toiminnot jaetaan prosesseihin, laadunhallintajärjestelmän vaikutus paranee ja asiakkaiden vaatimuksiin pystytään vastaamaan, joka taas parantaa asiakastyytyvyyttä. (SFS 2008)

Prosessimaisessa ajattelumallissa yrityksen täytyy tuntea toimintonsa ja niiden vaikutus toisiinsa pystyäkseen toimimaan tehokkaasti. Käyttämällä yrityksen resursseja ja johtamalla sisäisiä toimintoja tuottohakisesti syntyy joukko toimintoja, jotka muodostavat prosessin. Jokainen prosessi vaikuttaa osaltaan seuraavan prosessin onnistumiseen. Tämä ajattelumalli antaa yritykselle mahdollisuuden hallita prosesseja ja niiden vuorovaikutusta toisiinsa. Kun prosessimaista toimintamallia käytetään laadunhallintajärjestelmän osana, sen vaatimukset on ymmärrettävä ja todeta prosessien tehokkuus ja todeta niiden tuoma lisäarvo. Prosessien jatkuvaa parantamista tulee seurata objektiivisilla mittauksilla. (SFS 2008)

3 TESTAUS

3.1 Testauksen lähtötilanne ja kartoitus

Neocodex Oy:ssä oli tämän projektin teon aikana 2013 käynnissä toiminnan ja laadun kehittämishanke, joka oli nimetty ncxLaatu-projektiksi. Se käynnistyi testaukseen ja laadunvarmistukseen liittyvällä kyselyllä, jonka avulla kartoitettiin nykyistä testauskäytäntöä, dokumentointia ja laadunvarmistusta Neocodex Oy:n eri toiminnoissa. Kyselyn vastaukset olivat tärkeä osa tätä insinööriyötä ja toiminnan kartoitusta, jonka vastaukset käsiteltiin luottamuksellisesti. Kysely löytyy liitteenä opinnäytetyöstä (Liite 1).

Neocodex Oy:n tekemien ohjelmistojen ja toimitusten virheettömyyttä varmistettiin pääasiassa siten, että ohjelmoijat itse pyrkivät saamaan sovellukset toimimaan annettujen määritysten mukaan. Sovellusta testattiin myös siten, että toinen työntekijä suoritti testauksen. Sovellussuunnittelijat karsivat virheitä jo koodausvaiheessa. Kun ohjelmaan lisättiin koodia, niin sen jälkeen se testattiin ja kokeiltiin ohjelman uuden ominaisuuden toimivuutta. Kyseessä on iteratiivinen prosessi siten, että koodia lisätään vähän kerrallaan, jonka jälkeen ohjelmaa testataan. Isommissa muutoksissa testauksen voi suorittaa toinen henkilö, joka kirjaa ja raportoi ohjelman toimivuudesta. Järjestelmällisen ja suunnitelmällisen testauksen osalta Neocodex Oy:n toiminnassa oli kehittämisen varaa ja usein suurin testausvastuu jäi tekijälle itselleen. Koska ohjelmantekijä osasi jo käyttää sovellusta, ei hän silloin havaitse ohjelman virheitä. Asiakas löytää ohjelmaa käyttäessään sellaisia ohjelmiston puutteita, jotka jäivät tekijältä huomaamatta, koska asiakas käyttää ohjelmaa eri tavalla.

Yritys haluaa varmistaa, että ohjelma vastaa asiakkaan vaatimuksia, ja että määrittelyvaiheen asiakasvaatimukset välittyvät mahdollisimman hyvin ohjelmiston toteutukseen. Ongelmia tuottaa asiakkaan vaatimuksien ja määrittelyjen puutteellisuus tai peräti olemattomuus. Tämä näkyy siinä, ettei asiakas aina tiedä määrittelyvaiheessa mitä tarvitsee. Vasta ohjelman valmistuessa alkaa asiakkaalle syntymään tarpeita ohjelman suhteen. Asiakasvaatimukset ovat myös iteratiivisia, mikä lisää kustannuksia, vaikka valmiiseen ohjelmaan lisättävät lisäominaisuudet voivat olla helppoja tehdä, samalla voi jotakin vanhaa toiminnallisuutta hävitä tai muuttua. Jälkeenpäin ohjelmakoodiin tehtävä muutos vaatii enemmän aikaa ja panostusta kuin sen lisääminen suunnitteluvaiheessa.

Suunnitteludokumentteja, joita käytetään testaussuunnitelman pohjana ovat sähköposti, asiakkaalta saatu tuloste tai kaupanteon jälkeen yleensä tarjouksessa esitetyt vaatimukset. Varsinaista testaussuunnitelmaa ei ollut, eli testaus tapahtuu käytännössä ”yrityksen ja erehdyksen kautta”. Aiemmin käytössä ollut AWR Report Builder –ohjelma mahdollisti ongelman/kehityskohteen kirjaamisen BugTrackeriin, josta kehittäjä otti tapauksen käsittelyyn testausta varten. Testaus suoritettiin toisen henkilön toimesta ja tällöin testattiin kuvauksessa ollut tilanne ja todettiin muutoksen toimivuus.

Testauksen tulokset dokumentoidaan osittain BugTrackeriin sen käytön yhteydessä, mutta yleensä siihen merkitään vain, että ohjelma toimii. BugTracker ohjelman käyttö olisi edelleen mahdollista, mutta sitä ei hyödynnetä tarpeeksi.

Kyselyn tuloksena saatiin myös muotoiltua lista työntekijöiden toiveista liittyen laatu-
projektiin:

- koodauksen ohjausta
- koodausopas käyttöön
- käyttötapauksen kirjaaminen toimintatavaksi (testausten osalta).

Laatuprojektin avulla toivotaan selkeää ohjeistusta ja pohjustusta ohjelmointityön aloittamiselle.

Koodauksen ohjeistamisella ja koodausoppaalla (esimerkiksi luokkien, muuttujien, metodien ja funktioiden nimeämiset) saataisiin yhtenäistettyä eri henkilöiden tekemää koodia, jolloin mahdollisten ongelmien korjauksessa olisi helpompaa tulkita toisen tekemää koodia. Käyttötapaukset, joiden mukaan ohjelmaa käytetään ja testaus tulisi suorittaa, täytyisi myös ottaa toimintatavaksi.

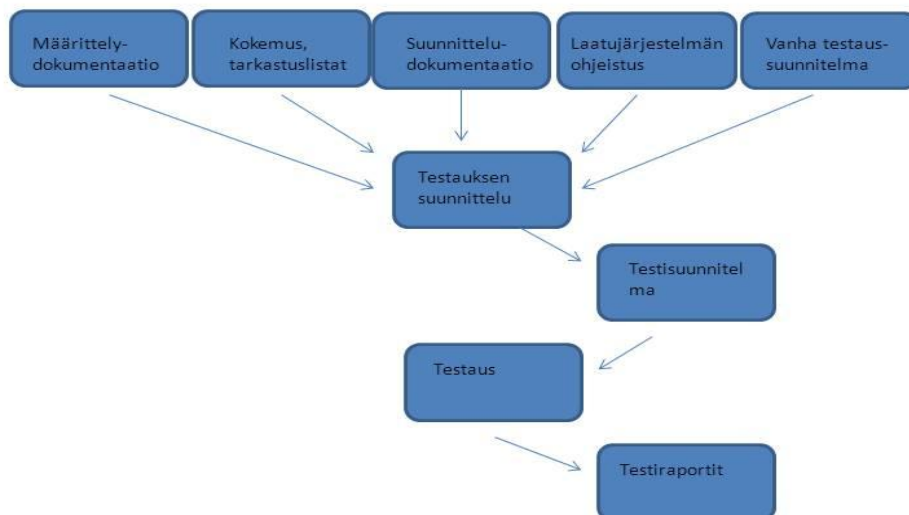
3.2 Testauksen yleiset periaatteet

Ohjelmistotestauksen periaatteena on havaita vikoja ohjelmistosta. Testaus suoritetaan toisen samantasoisien työntekijän toimesta, koska tekijä harvoin huomaa omia virheitään. Useimmiten virheitä syntyy väärinymmärretyistä määrittämisistä, jonka seurauksena ongelma jatkuu myös muissa työvaiheissa kuten suunnittelussa, ohjelmoinnissa ja testauksessa. Työntekijä suhtautuu myös usein työhönsä vakavasti, kun hän tietää toisen henkilön suorittavan tarkastuksen ohjelmakoodille. Tavoitteina tarkastusmenettelyssä on löytää virheet mahdollisimman hyvissä ajoin, varmistaa työnteki-

jöiden yhteisymmärrys projektin tavoitteista, jakaa tietoa samanlaisten projektien kanssa työskentelevien kollegoiden kesken, varmistua työlle annettujen vaatimusten täytymisestä, antaa muodollinen päätös työvaiheelle ja saada mittauskelpoista tietoa tuotteesta. (Itä-Suomen Yliopisto 2013)

Ohjelmistotestauksen työvaiheet muodostuvat neljästä osasta, jotka ovat suunnittelu, testiympäristön luonti, testin suorittaminen ja tulosten tarkastelu. Testauksen työvaiheisiin liittyy läheisesti virheiden jäljitys ja korjaus, johon kuuluu yli puolet ohjelmistoprojektin ajasta. Yrityksen kannattaa panostaa testauksen läpivientiin parhaalla mahdollisella tavalla, ottaen huomioon siihen käytössä olevat resurssit esimerkiksi aika, raha ja välineet.

Testaus tarkoittaa mitä tahansa kokeilemistä. Haikalan ja Märijärven (2004, 184) mukaan *”testaus määrittellään suunnitelmalliseksi virheiden etsimiseksi ohjelmaa tai sen osaa suorittamalla. Testaus voidaan toteuttaa umpimähkäisesti kokeilemalla joillain syöttöaineistoilla, kun yritetään osoittaa ohjelman toimivuus”*. Kun tavoitteena on virheiden löytäminen, pienellä huolellisella suunnittelulla päästään ajallisesti tehokkaampiin tuloksiin. Tarkastuksia ja ohjelmakoodin staattista analysointia voidaan kutsua myös testaukseksi.



Kuvio 1. Kuvaus testauksen suunnittelusta. (Haikala & Märijärvi, 2004, 298).

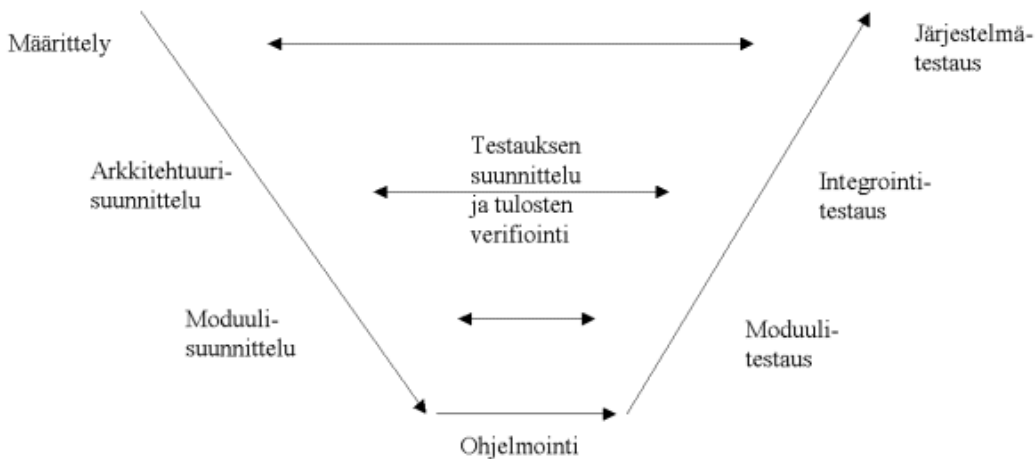
Testausta suunniteltaessa on hyvä olla tietoinen ohjelman syöte- ja tulosavaruudesta, sekä on tiedettävä mihin lopputulokseen pyritään. Tämän edellytyksenä täytyy olla spesifikaatiot (esimerkiksi käyttötapaukset), joiden mukaan testitapaukset pystytään suunnittelemaan ja päättämään hyväksyttävät lopputulokset. Kuviosta 1 nähdään testauksen suunnittelun vaiheet.

Yksinkertaisissakaan tapauksissa testauksella ei pystytä osoittamaan ohjelman virheettömyyttä. Testauksen avulla ohjelmasta löydetään vain virheitä. Testaamisella katetaan hyvin pieni osa ohjelmistotestauksen virheistä ja reaaliaikaisissa järjestelmissä vieläkin vähemmän. Siitä huolimatta testauksen kannattaa panostaa. Testatun ohjelman virheettömyyteen ei silti pidä luottaa, vaikka tulokset olisivat hyviä. Vaikka testauksella on ohjelmistokehityksessä suuri merkitys, avainasemaan nousee huolellinen suunnittelu ja ohjelmoinnin merkitys. Ohjelmakoodin testaus on havaittu kustannustehokkaaksi menetelmäksi virheiden löytämiseksi, vaikka se koetaan usein aikaa vieväksi.

Virhe testauksessa tulee poikkeamasta määritelmässä. Spesifikaatio on pakollinen, koska ilman sitä lopputuloksen virheettömyyttä ei voida tietää. Toiminnallinen ja tekninen määrittely ovat yleisimpiä ohjelmistotestauksessa käytettäviä spesifikaatioita. Siitä huolimatta parhaasta spesifikaatiostakin löytyy aina puutteita, eikä ongelmat ratkea niitä tutkimalla. Suunnitelmiin ja sopimuksiin kirjataan mitä näille ristiriitatilanteille tehdään. Arvion mukaan käytössä olevissa ohjelmistoissa noin viisi prosenttia ohjelmistovirheistä on sellaisia, joita ei koskaan havaita. (Haikala & Märijärvi, 2004, 283-288.)

3.3 Testaustasot

Ohjelmistotestauksen tasoja ovat moduuli eli yksikkötestaus, integrointitestausta ja järjestelmättestaus. Järjestelmättestauksen alalajeja ovat erillinen kenttättestaus ja hyväksymistestaus. V-mallin testauksen suunnittelu tehdään aina testaustasoa vastaavalla suunnittelutasolla, eli järjestelmättestaus määrittelyvaiheessa, integrointitestausta suunnitteluvaiheessa ja moduulitestausta moduulitestausta vaiheessa. V-mallin testauksen tasot näkyvät kuviossa 2. Jotta tulokset todetaan oikeiksi, verrataan niitä vastaaviin dokumentteihin. (Haikala & Märijärvi, 2004, 288.)



Kuvio 2. Testauksen V-malli. (Haikala & Märijärvi, 2004, 289).

Moduulitestauksessa tarkastetaan vain yksittäistä moduulia, joka muodostuu, noin 100-1000 ohjelmakoodin rivistä. Sen toimintaa verrataan yleisesti tekniseen määrittelydokumenttiin. Integrointitestauksessa testattavat osat muodostuvat useista moduuleista ja painopisteenä on moduulien välisten rajapintojen toimivuus. Tuloksia verrataan tekniseen määrittelyyn. Kun halutaan tarkastella koko järjestelmän toimivuutta, siihen käytetään järjestelmätestausta. Tuloksia verrataan määrittelydokumentaatioon ja asiakasdokumentaatioon. Järjestelmätestauksen osa-alueisiin kuuluvat muun muassa kuormitustestit, luotettavuustestit, asennustestit ja käytettävyydestit.

Mitä pidemmällä testaustasolla ollaan, sitä suuremmat kustannukset virheiden korjauksesta aiheutuu. Korjaus saattaa myös tuoda mukanaan uusia virheitä ja muutokset voivat kohdistua useisiin moduuleihin. Kun virheitä löytyy, uudelleentestausta suositellaan kaikilla testaustasoilla. Tällaisissa tilanteissa automatisoitu testaus tulee yritykselle huomattavasti edullisemmaksi.

Järjestelmätestausvaiheessa toteutetaan käytettävyydestausta mahdollisimman riippumattomalla tarkastushenkilöllä. Käytettävyydestausta liittyy yleensä käyttöliittymän testaukseen. Testausta varten tehdään otanta tulevista käyttäjistä, joiden avulla määritellään ohjelmiston toimivuutta. Koehenkilöt voivat olla myös ohjelmistojen käytettävyyden arviointiin perehtyneitä ammattilaisia. Ennen kuin ohjelmisto päätyy asiakkaalle, voidaan suorittaa alfa-testaus, jossa asiakas testaa ohjelmaa toimittajan tiloissa. Beta-testaus eroaa edellisestä siten, että testaus suoritetaan itsenäisesti asiakkaan tiloissa.

Testauksen riittävää laajuutta on hankala määrittellä. Tuotekehitysvaiheessa testauksen laajuus määrittyy tuotteen vioista johtuvien kustannusten ja aikataulun aiheuttaman tuoton vähenemisen välillä. Jos tuotekehitysvaiheessa lyhennetään testaukseen käytettävää aikaa, saadaan tuote nopeammin markkinoille ja siitä tuloja. Mutta samalla kustannukset voivat samalla kasvaa, kun virheitä sisältävä ohjelma aiheuttaa korjauskustannuksia. Jos taas pidennetään testausaikaa, menetetään myyntituloja siltä aikaa kun tuote on poissa markkinoilta. Tämän yhtälön kanssa yrityksessä painitaan kun yritetään optimoida myyntitulojen sisääntulo ja virheiden vähyys.

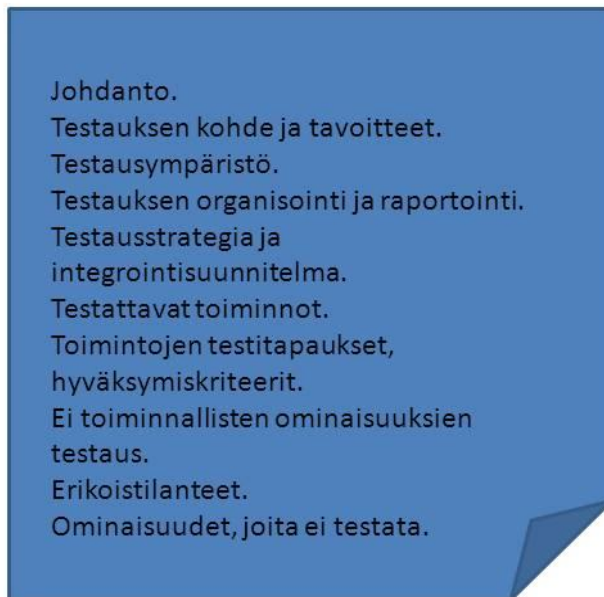
Testaussuunnitelmassa täytyy määrittää hyväksymiskriteerit, sille missä vaiheessa testaus lopetetaan. Järjestelmätestauksen riittävyttä arvioidaan seuraamalla virhekkäyrää, kun virhekkäyrä tasaantuu voidaan testaus lopettaa. Virhekkäyrän yhteydessä on hyvä käyttää myös muita mittareita, koska virhekkäyrän tasaantumista odotellessa resurssit ja aikataulut voivat ylittyä. Testauksen riittävyyden mittaamisessa käytettäviä muita menetelmiä ovat mutkikkuusmittaus, kattavuusmittaus ja virheiden kylväminen. (Haikala & Märijärvi, 2004, 288-296.)

3.4 Testausprosessin kuvaus

Haikalan ja Märijärven (2004, 299) mukaan *”alustava suunnitelma laaditaan määrittelyvaiheessa, ja sitä täydennetään myöhemmin suunnitteluvaiheessa”*. Testaussuunnitelma voidaan sisällyttää projektisuunnitelmaan, toiminnalliseen määrittelyyn ja tekniseen määrittelyyn. Suunnitelmasta tulee ilmi mitä testejä käytetään ja milloin, sekä miten ne toteutetaan ja millaiset odotukset ovat lopputuloksella. Lopettamiskriteerien määrittäminen on erittäin tärkeää. Testaukselle varatun ajan päättymiselle voidaan asettaa lopettamiskriteerinä se, että 100% testaustapauksista on ajettu ja analysoitu, sekä 100% ilmenneistä vioista on korjattu ja tämän jälkeen ajettu 100% testaus, joka tarkoittaa ohjelman testausta kaikilla eri testaustasoilla. (ks. Kuvio 3.)

Kaikilta testaustasoilta löydetty virheet täytyy raportoida ja analysoida. Dokumentoitavia tietoja ovat esimerkiksi virheenkuvaus, vakavuustaso, milloin virhe on löydetty, miten se olisi voitu löytää aikaisemmin, milloin virhe oli syntynyt ja olisiko se voitu estää. Virheen huomattessaan asiakas ilmoittaa siitä erityisellä virheilmoituslomakkeella. Näin virheiden dokumentointi raporttien muotoon auttaa saamaan selville testaukseen käytettyä aikaa ja tuo ilmi laatujärjestelmän kehityskohteita. ISO 9001-vaatimusten mukaan virheet dokumentoidaan järjestelmätestausraporttiin. Myös reklamaatioista johtuvat virheiden korjaukset kirjataan. Helpoin ja suositeltu menetelmä on

käyttää testauspäiväkirjaa. Päiväkirjaan tulee kirjata millainen virhe, missä ja milloin tehtiin, milloin löydettiin, milloin se olisi pitänyt löytää, miksi virhettä ei huomattu aikaisemmin, miten se korjattiin, korjaukseen käytetty aika ja miten sen syntyminen olisi voitu estää. (Haikala & Märijärvi, 2004, 298-300.)



Kuvio 3. Testaussuunnitelman sisältö (Haikala & Märijärvi, 2004, 300).

Tuotteen valmistuessa tarkastuksen vaiheisiin kuuluvat tarkastuksen valmistelu, tarkastustilaisuus ja jälkihoito. Ohjelmiston tekijä ja organisaation johtaja tekevät päätöksen, kun tuote on valmis tarkastettavaksi. Tarkastusmenettelyn tavoitteena on etsiä suunnittelun virheet, erot aikaisempiin määrittelyihin ja että työ noudattaa annettuja ohjeistuksia. Tarkastajat määrätään ja valvoja päättää ketä muita tarkastukseen saa osallistua. Valmisteluvaiheessa valitaan tarkastettava dokumentti, määrittelevät dokumentit, ohjeistot, sekä tarkastuslomakkeet. Tämän jälkeen järjestetään aloituskokous, jossa valvoja kertoo tarkastajille tarkastettavan dokumentin sisällön ja tarkastuksen tavoitteet ja mahdollisesti kertoo jatkotoimista. Kokouksen päätyttyä jokainen tarkastaja perehtyy annettuun materiaaliin, tarkastuslistoihin, ohjeisiin, standardeihin

ja yleisimpien virheiden tilastoihin ja kirjaa käyttämänsä ajan, havaitsemansa virheet ja raportoi niistä valvojalle. Valvoja tekee selvityksen tarkastajien antamista raporteista työntekijöille, jotka tutustuvat virheraportteihin ja laativat niihin vastaukset.

Tarkastus alkaa kun ohjelmoijat ovat valmistelleet vastauksensa ja tarkastajien raportit on palautettu valvojalle. Tarkastusaikaa siirretään jos edellä mainituissa asioissa on puutteita. Tarkastustilaisuudessa tarkastaja käy yhdessä työntekijän kanssa läpi tarkastettavan kohteen huomautukset tai virheet ja yhteistyö toimii parhaiten kun työntekijä ei ryhdy puolustuskannalle. Tarkastajat keskittyvät virheisiin ja antavat työntekijälle kiitosta kun siihen on aihetta. Virheestä kirjataan paikka, kuvaus, virheen luokka ja syy jos se on helposti havaittavissa. Lisäselvitystä vaativissa tapauksissa epäselvät kohdat käydään läpi. Valvojan tehtävä on huolehtia, että tilaisuudessa etsitään virheitä, mutta niitä ei korjata. Valvoja tekee päätöksen tarvittaessa uusintatarkastuksesta, jos prosessi tai virheiden jakauma ei ole normaali tai joku osallistujista esittää tarkastuksen suorittamista uudelleen. Valvoja varmistaa, että kirjausmerkintöjä on riittävästi ja käy läpi yhteenvetolomakkeet. Mahdollisimman hyvän lopputuloksen saamiseksi tarkastus kestää enimmillään kaksi tuntia. Tarkastus keskeytetään jos paljon pieniä virheitä löytyy, koska suuret virheet jäävät tällöin vähemmälle huomiolle.

Jälkitöihin kuuluu pöytäkirjan viimeistely, joka toimitetaan osallistujille ja tallennetaan laatutietokantaan. Ohjelmiston tekijät korjaavat löydetyt virheet ja käsittelevät ne valvojan kanssa. Valvoja varmistaa, että tarkastustiedot viedään laatutietokantaan ja ilmoittaa johdolle, että tarkastus on viety loppuun hyväksytysti.

Tarkastuksia voidaan suorittaa myös virtuaalisesti videoneuvottelun avulla tai esimerkiksi sähköpostilla. Tällaisissa tapauksissa valvoja huolehtii, että osallistujat tekevät heille määrätyt toimet, jolloin tarkastus voi jatkua eteenpäin.

Tarkastus epäonnistuu yleensä seuraavista syistä: valmistelu ei ole riittävää, tilanteessa on liian paljon osallistujia tai väriä henkilöitä tai tarkastettavaa koodia on liian paljon.

Tietojen keräyksen päämääränä on tarkastuksen läpivienti asianmukaisin tavoin, tuoda esiin ongelmat, määrittää tarkastusmenettelyn tehokkuus ja antaa tietoa tulevia tarkastuksia varten. Yrityksen johto ei saa arvioida työntekijöitä tarkastajien tekemien raporttien mukaan, siksi nämä laatutietokantaan viedyt yhteenvetoraportit on syytä anonymisoida.

Laatutietokanta on kokoelma, joka koostuu tuotteen tai työskentelyn laatuun liittyvistä dokumenteista ja niiden pohjalta tehdyistä yhteenvedoista. ISO-9001- standardin mukaan laatutietokannan jatkuva ylläpito on osoitus vaatimuksien noudattamisesta ja laatujärjestelmän tehokkaasta toiminnasta. Tarkastuksen tulokset ja toiminnot dokumentoidaan arkistoon. (Itä-Suomen Yliopisto 2013)

3.5 Kehittämiskohdat

Kyselyn pohjalta havaitut kehityskohdat ohjelmoijien toiminnassa ja päivittäisessä työssä nousevat selvästi esiin vastauksissa. Eniten esille tullut asia on testaussuunnitelman puuttuminen tai sen puutteellisuus. Ohjelmien virheitä etsitään omin voimin johon tuhrautuu paljon aikaa ja resursseja. Varsinaisen testaussuunnitelman mukaan testauksen suorittaa aina toinen henkilö, koska tekijä ei omia virheitään välttämättä huomaa. Näin testausvastuu ei jää tekijälle itselleen. Testaussuunnitelman pohjustukseen on hyvä hyödyntää edellä mainitussa teoriassa esitettyä testaussuunnitelman pohjaa (kuva 2, s.20), jotta päästään ajallisesti mahdollisimman tehokkaihin tuloksiin ja laatuun työprojektin edetessä.

Suurena kehittämisenarvoisena asiana huomattiin myös käyttötapausten kirjaamiseen liittyvä satunnainen käytäntö. Käyttötapausten luovat monelle asiakasprojektille jo valmiin pohjan, koska tietyt käyttötapaukset kertovat tekijälleen jo mihin lopputuloksiin tietyllä koodilla päästään. Tämä säästää paljon tekijän aikaa. Käyttötapaukset toimivat myös testauksen kannalta tärkeässä asemassa, koska aikaisempia käyttötappauksia verrataan testattaessa tuotettuun ohjelmakoodiin.

Asiakkaan kanssa tehtyyn kartoitusvaiheeseen kaivataan myös jonkunlaista yhtenäistä pohjustusta, koska työn tekemisen loppuvaiheessa tehtävät muutokset ovat yleensä asiakkaille kalliita ja aiheuttavat myös paljon työtä.

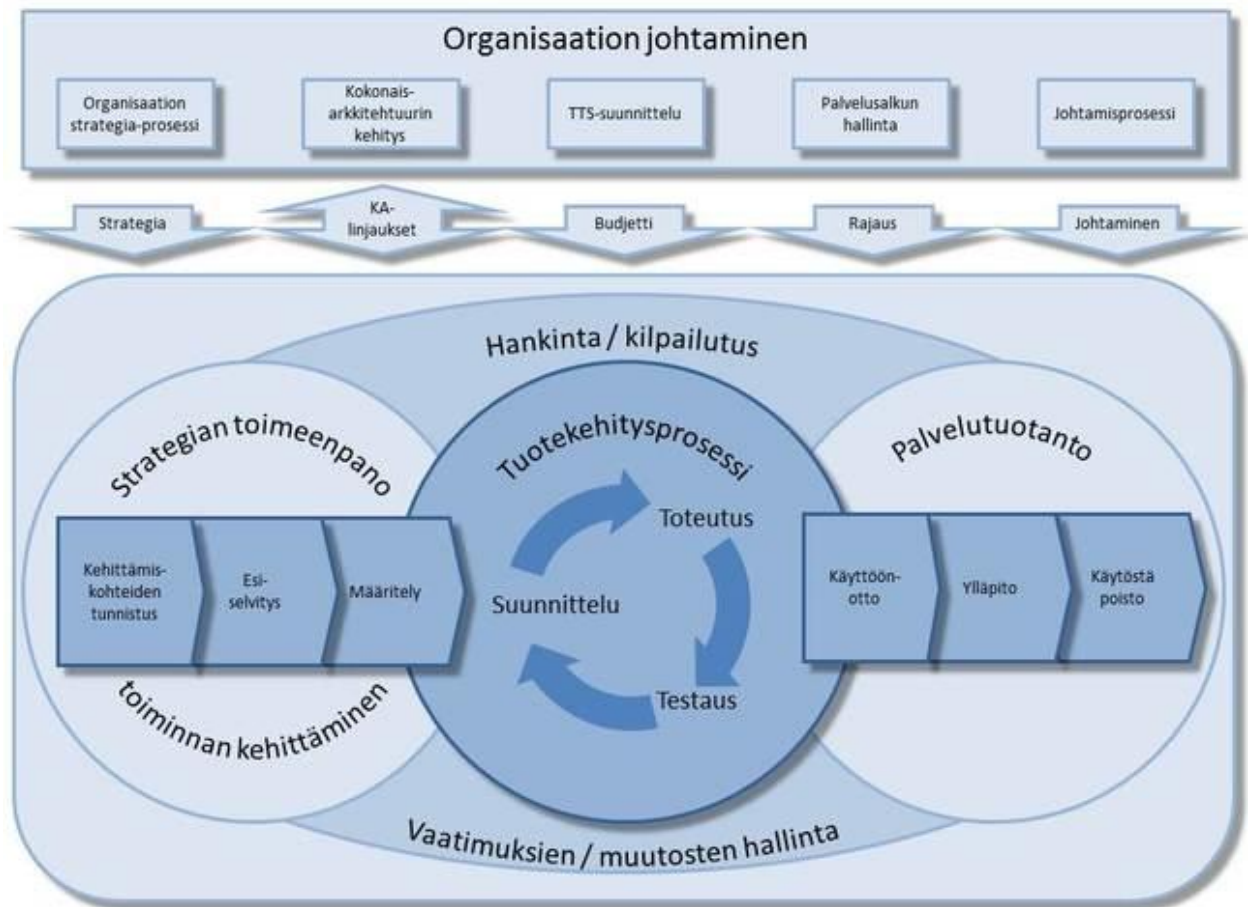
Koodaukseen liittyen toimintatapoihin toivottiin yhteisiä toimintatapoja, johon on ratkaisuna kaikkien käytössä oleva koodausopas.

4 NEOCODEX OY:N PROSESSIT JA NIIDEN KEHITTÄMINEN

Neocodex Oy:n pääprosesseihin kuuluvat projektitoimitukset, ylläpitopalvelut ja tuotekehitys, joita käsitellään seuraavissa kappaleissa Neocodexin toimintojen mukaisesti. Prosessijohtamista ja asiakkuuksienhallintaa käydään läpi vain yleisellä tasolla.

4.1 Prosessijohtaminen

Johtaminen koostuu monesta eri elementistä, joiden tarkoitus on jatkuvan toiminnan parantaminen ja tuloshakuisuus. Alla oleva kuvio 4 kuvaa palvelun elinkaarta ja johtamisen prosessia sen osana.



Kuvio 4. Palvelun elinkaari (JHS)

Monesta yrityksen toiminnoista syntyvän ketjun tarkoitus on vastata asiakkaiden tarpeisiin ja luoda lisäarvoa. Näiden toimintojen ketjulla yritys saavuttaa tuloksia, siksi niitä täytyy myös johtaa sen mukaisesti. Prosessijohtamisen lähtökohtana on tunnistaa tarvittavat toiminnot ja saada organisaatio toimimaan niiden mukaisesti kustannustehokkaalla tavalla. Tavoitteiden asettaminen ja kehityssuunnitelman luominen on toimivan prosessijohtamisen edellytys. Ongelmia arvon luonnille tuovat yleensä tiedonkulun puute tai uusien ihmisten nopea kouluttaminen. (Laamanen & Tinnilä, 2009, 9-11.)

Prosessimainen toimintamalli saadaan käytäntöön yleensä mallintamalla se, jota kautta asiakkaalle arvoa tuovat toiminnot pystytään kuvaamaan yksityiskohtaisemmin ja viemään käytäntöön. Liian yksityiskohtaista toimintojen mallintamisesta ei kannata tehdä, sillä se voi jatkossa aiheuttaa päänsäivää. Kun tässä onnistutaan se näkyy pitkäkestoisina asiakassuhteina ja asiakkaat kokevat saavansa vieläkin parempaa palvelua. Työntekijöiden motivaatioon onnistunut mallintaminen vaikuttaa kasvavana työntona kun arvonluonnin kokonaisuus ymmärretään käytännössä. Organisaatio pystyy jatkossa hyödyntämään asiakkaiden tarpeita, tehdessään toimintaan liittyviä kehitystoimia ja näin ollen kehittämään tuotteitaan ja palveluitaan entistä asiakaslähtöisempään suuntaan.

Laatujohtamisen yhteydessä puhutaan yleensä myös prosessiajattelusta ja jatkuvasta parantamisesta. Prosessijohtamisessa täytyy ottaa huomioon yrityksen käytössä olevat resurssit ja se miten kuluja voidaan minimoida, jotta saadaan paras mahdollinen tuotto. Tietotekniikka on osasyynä prosessien mallintamiseen, koska sen avulla moni toiminto tehdään yhdenmukaisemmin ja kokonaisuus hahmotetaan selkeämmin. Prosessiajattelun kehittämisen tarkoituksena on saada työntekijät tekemään asiat erilailla, kannattavammin kuin aikaisemmin. Siksi sanotaankin että prosessijohtaminen kohdistuu suoraan organisaation toimintoihin. (Laamanen & Tinnilä, 2009, 9-12.)

4.2 Projektitoimitukset

Projekti alkaa asiakkaalta tulevalla tarjouspyynnöllä, joka on yleensä kirjallinen. Tarjouspyyntöön vastataan ja kerrotaan asiakkaalle arvio ajasta milloin tarjous lähetetään. Asiakkaan tarjouspyyntö tallennetaan asiakasta varten luotuun asiakashakemistoon. Tarjouspyynnön avoimet kysymykset käydään läpi joko sähköpostitse tai asiakaspalaverissa. Dokumentit tallennetaan asiakaskansioon.

Tämän jälkeen luodaan palaverimuistio valmiin dokumentin pohjalta, joka tallennetaan asiakaskansioon. Kun tarjottava kokonaisuus on määritelty tehdään valmis laskelma tarjouspohjaa hyödyntäen. Valmis laskentapohja tarjoaa tuntipohjaisen hinnoittelun ja kiinteähintaiset työt. Muilta osin, esimerkiksi kehitysluontoiset kokonaisuudet määritellään ajan käytön ja hinnoittelun osalta yhdessä tehtävästä vastaavan työntekijän ja hinnoittelusta vastaavien henkilöiden kanssa. Tarjous jaetaan varsinaiseen tarjousosaan, joka sisältää välttämättömät asiat ja erillisiin vaihtoehtoihin asioihin. Tämän jälkeen dokumentti tallennetaan tarjoushakemistoon.

Asiakkaalle valmistellaan vakiomuotoiset tarjousesitteet sekä mahdolliset uusien tuotteiden tarjousesitteet. Tämä osuus vie paljon aikaa, koska pelkkä tekstimuotoinen esite ei ole riittävä, vaan mukaan on liitettävä kuvaa tiedon esittämismuodosta ja toiminnallisesta käyttöliittymästä. Dokumentit tallennetaan asiakaskansioon.

Tarjous lähetetään pääsääntöisesti asiakkaalle sähköpostilla ja tarjous tallennetaan tarjoushakemistoon. Tarjouksen esittely sisältää usein myös yhtiön, tuotteiden ja palvelujen esittelyn. Asiakaspalaverissa tarjous jaetaan vielä varsinaisiin ja vaihtoehtoihin tarjousosiin.

Tarjouksen tarkennuspalaverissa tehdään tarkennettu tarjouspalaverimuistio, joka tallennetaan asiakaskansioon. Palaverissa esille tulleet uudet työt ja tuotteet hinnoitellaan ja päivitetään tiedot palaverimuistioon. Palaverissa käydään läpi mahdolliset esitemuutokset ja uusien esitteiden teko. Dokumentit tallennetaan asiakaskansioon.

Viimeisen vaiheen tarjousneuvottelussa tehdään niin sanottu tarjouslaskennan hermistely ja käydään läpi, missä asioissa on mahdollisuuksia joustaa ja missä on suurimmat riskit. Määritellään osakokonaisuuksia varsinaiselle tarjoukselle ja mitkä jäävät optioiksi. Dokumentit tallennetaan asiakaskansioon.

Asiakkaan hyväksyessä tarjouksen ja tilausvahvistuksen saapuessa vahvistetaan toimitusaika sekä muu aikataulutus. Samalla määritetään omien resurssien kiinnitys projektiin.

Projekti aikataulutetaan ja projektiympäristö varataan. Mahdollisten palvelimien ja ohjelmistojen tilaus hoidetaan ennen projektin alkamista.

Projektin alkaessa pystytetään projektiympäristö, kun kaikki ohjeen mukaiset tiedot ovat kasassa. Perussovellukset asennetaan ja parametrisoidaan ja testataan niiden toimivuus. Tämän jälkeen projektikohtaiset sovellukset määritellään ja ohjelmoidaan, jonka jälkeen niiden toimivuus testataan. Tehdastestauksen suunnittelu eli Factory Acceptance test voidaan aloittaa, kun tarvittavat työkalut ja ohjelmistot on asennettu. Sen jälkeen FAT-suunnittelun tuloksia tarkastellaan ja esille tulevat lisäykset ja muutostarpeet käydään läpi.

4.3 Ylläpitopalvelut

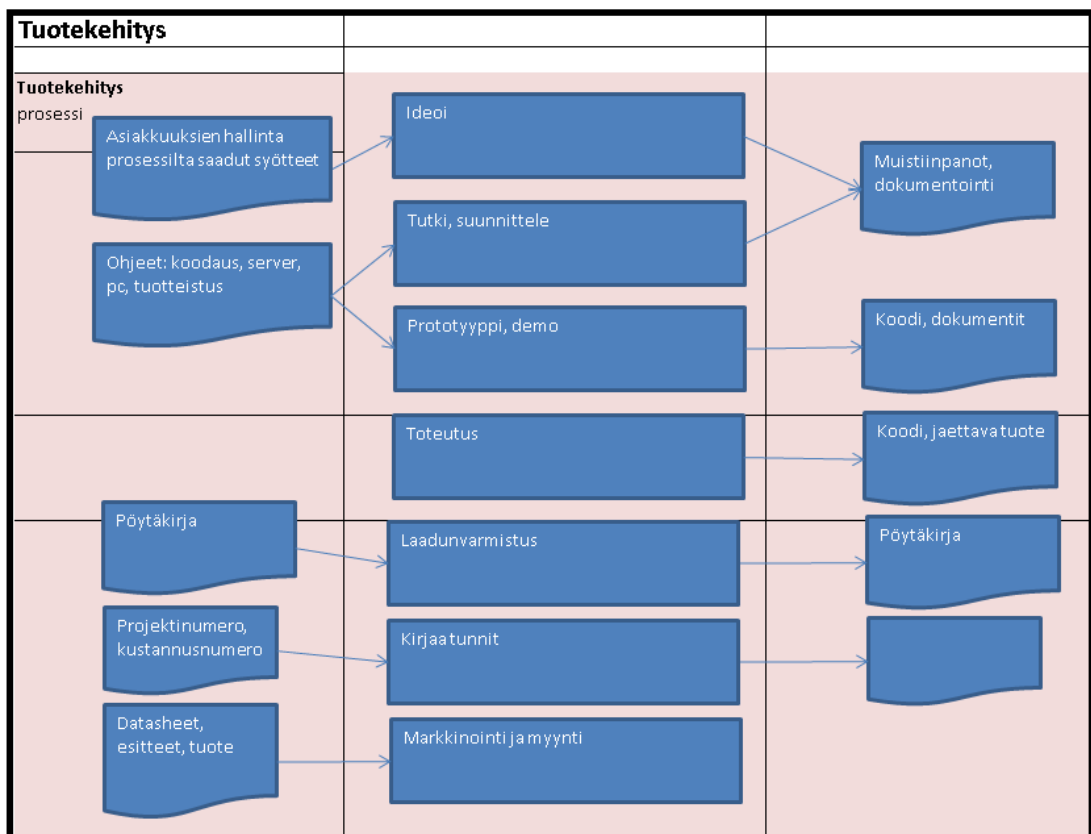
Ylläpitopalvelut tarkoittaa ohjelmistojen ja mahdollisesti muiden järjestelmien toiminnan jatkuvuuden turvaamiseksi tehtäviä töitä. Ylläpitoon päätyvä ohjelmisto on yleensä Neocodex Oy:n toimittama ja tekemä. Neocodex ylläpitää myös toisten valmistamien ohjelmistojen. Ylläpidosta tehdään erillinen sopimus, jos sitä ei ole projekti-toimituksessa sovittu. Ylläpitoon siirrytään ohjelmiston takuuajan jälkeen. Ylläpidon resursseja ovat: henkilöt (omat ja asiakkaan), tietokoneet, verkkoyhteydet ja dokumentit.

Neocodex Oy:n ylläpitoon siirtyvät ne ohjelmistot, joista on tehty sopimus asiakkaan kanssa ja takuu aika ohjelmistoissa on umpeutunut. Usein asiakas on käyttänyt ohjelmistoa takuuajan ja Neocodex Oy tarjoaa asiakkaille ylläpitopalvelua. Jos asiakas haluaa ottaa käyttöön ylläpitopalvelun, siitä kirjoitetaan kirjallinen sopimus. Sopimuksessa sovitaan hinnasta, käyttöön oton aikataulusta, palvelun laajuudesta ja saatavuudesta. Ylläpitopalveluille varataan tarvittavat henkilö- ja laitteistoresurssit, joilla palvelua tuotetaan. Kun asioista on sovittu ja asiakas ottaa käyttöön ylläpitopalvelun, niin asiakas voi jatkaa ohjelmiston käyttöä ylläpitopalvelun turvin. Laadunvarmistamiseksi asiakkaalle lähetetään jälkimarkkinointina kysely ohjelmistoon ja ylläpitopalvelun toimivuuteen liittyen. Laadunvarmistamista toteutetaan myös ylläpitopalvelua käyttävän asiakkaan ja toimittajan asettamien vaatimuksien mukaisesti. Ylläpitopalvelu päättyy sopimuksen mukaisesti.

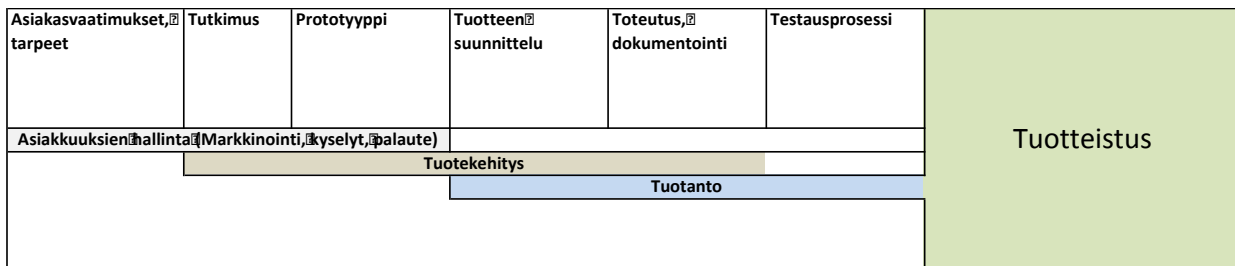
Jos asiakas ei ole halukas ottamaan käyttöön Neocodex Oy:n tarjoamaa ylläpitopalvelua, hän saa jälkimarkkinointina lähetettävän kyselyn kuukauden kuluttua ohjelman käyttöönotosta. Yritykselle on tärkeää kerätä myös tässä tapauksessa tietoa asiakkaalta hänen käyttökokemuksistaan, koskien toimitettua ohjelmistoa.

4.4 Tuotekehitys

Tuotekehitysprosessia ei ole toteutettu Neocodex Oy:ssä kuvioiden 5 ja 6 mukaisesti. Tuotekehityksen tarkoituksena on pyrkiä täyttämään ulkoisen eli maksavan asiakkaan tai sisäisen asiakkaan eli Neocodex Oy:n tarpeet. Asiakkaan tarpeet kartoitetaan ja tutkitaan ja pyritään löytämään niihin erilaisia ratkaisuja. Ratkaisuista tehdään niin sanottuja prototyyppejä ja erilaisilla ohjelmavariaatioilla tehdään kokeiluja. Tuotekehitysprosessia ei ole yhtiön toimesta hyväksytty alla olevien tehdasmallin ja tuotekehitysmallin muodossa, vaikka tuotekehitys olisi hyvä tehdä määrätietoisena projektina. Tuotekehityksellä on vahva sidos asiakkuuksien hallinta- ja projektitoimitus prosesseihin. Niistä saatua tietoa asiakkaan käyttäytymisestä ja tarpeista voidaan hyödyntää tuotekehityksessä. Tuotekehityksellä on oltava vahva sidos yrityksen liiketoimintaan siten, että sen luomat innovaatiot saadaan muutetuksi yrityksen tuloksiin.



Kuvio 5. Keskeneräinen tuotekehitysmalli. (Neocodex Oy).



Kuvio 6. Tuotteistuksen tehdasmalli. (Neocodex Oy).

4.5 Asiakkuuksien hallinta

Markkinoinnin tarkoituksena on tavoittaa mahdollisimman monta potentiaalista asiakasta ja saada uusia asiakkaita sekä ylläpitää olemassa olevia asiakassuhteita. Myynnin osalta parhaaseen mahdollisimpaan tulokseen päästään tapaamalla asiakas henkilökohtaisesti, tavoitteena saada asiakas vakuuttuneeksi tarpeistaan ja siitä että yritys tarjoaa hänelle parhaan ratkaisun. Myynnin tarkoituksena on poistaa asiakkaalta kaikki ostamisen esteet, jotta kauppa syntyy.

Pitkäaikaiset ja kannattavat asiakassuhteet syntyvät asiakaskeskeisen johtamismallin tuloksena, tätä kutsutaan asiakkuuksien hallinnaksi. Customer Relation Management eli CRM kuvaa asiakassuhteen koko kiertokulun sen alusta loppuun saakka. Yrityksen täytyy tuottaa asiakkaalle positiivisia kokemuksia ja olla kilpailukykyinen, jotta asiakassuhde vakiintuu ja jalostuu entisestään. Pitkäaikainen ja sitoutunut asiakas parhaassa tapauksessa myös suosittelee yrityksen palveluja eteenpäin.

Myynnin, markkinoiden, toimituksen ja palveluiden yhtenäistäminen toimii edellytyksenä toimivalle CRM:lle. Monella yrityksellä on erityyppisiä asiakkaita, kun asiakkaat jaetaan tiettyihin segmentteihin, löydetään paras mahdollinen prosessi/toimintamalli tiettyjen asiakasryhmien tyydyttämiseen. Tänä päivänä asiakkaista kerätään paljon tietoa mm. heidän ostokäyttäytymisestään, palvelutarpeistaan jne. Kun yritys lähestyy asiakkaita CRM lähestymistavan kautta se parhaimmillaan pystyy ennakoimaan asiakkaitten tarpeita ja antamaan hyviä kokemuksia asiakkaalle omasta toiminnastaan. Myös mahdollisiin ongelmiin pystytään varautumaan etukäteen. (Laamanen & Tinnilä, 2009, 19.)

5 LAADUNPARANTAMINEN OHJELMISTON KEHITYSPROSESSISSA

5.1 Johdon tehtävä laadunparantamisessa

Laadunhallintajärjestelmäprojektin aloitusvaiheessa on hyvä keskustella ensin yritysjohton kanssa, koska johdon sitoutuminen projektiin ja sen kehittämiseen on erittäin tärkeää. Laatujärjestelmää tulee myös valvoa ja ylläpitää, jotta sitä pystytään jatkuvasti kehittämään prosessimaisen ajattelumallin kautta. Aluksi yritysjohton täytyy saada työntekijät ymmärtämään asiakasvaatimukset, lakien ja viranomaisten vaatimusten tärkeys päivittäisessä tekemisessä. Työntekijöiden täytyy toimia johdon asettaman laatupolitiikan mukaisesti. Laatupolitiikka asettaa puitteet laatutavoitteiden asettamiselle. Johdon täytyy suorittaa tietyin väliajoin johdon omat katselmuksot ja tarjota tarvittavat resurssit, jotka tukevat laadunhallintajärjestelmän toimivuutta parhaalla mahdollisella tavalla. (SFS 2008) Laatupolitiikkaa kuvattiin tarkemmin jo aikaisemmassa luvussa 2.3.

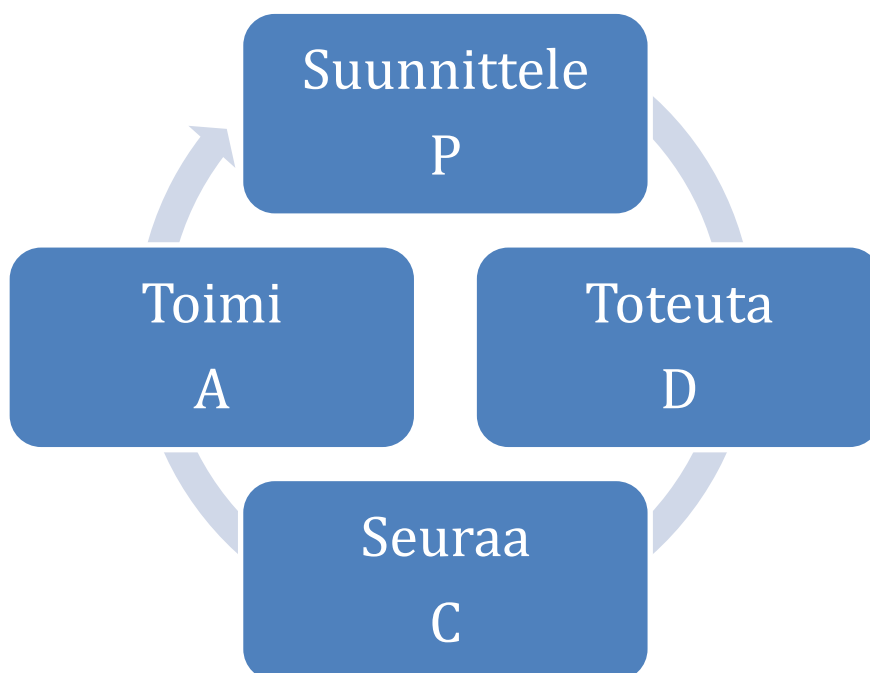
Yritysjohto määrittää laatua koskevat tavoitteet laadunhallintajärjestelmää tehtäessä myös tuotevaatimuksien osalta. Laatutavoitteita asettaessa ei kannata tavoitella liian korkealle, mutta tavoitteiden tulee olla kuitenkin mitattavissa ja laatupolitiikan kanssa yhdenmukaisia. (SFS 2008)

Laadunhallintajärjestelmän toimivuudesta vastaa yleensä johdon nimittämä toimihenkilö, jolle on annettu valtuudet ja vastuu luoda toimivat prosessit ja ylläpitää niitä. Hän raportoi johdolle laadunhallintajärjestelmässä asetettujen tavoitteiden ja prosessien toimivuudesta ja kertoo myös parannustarpeista jos niihin on aihetta. Asiakasvaatimukset ovat etusijalla, joten sen tietoisuuden edistäminen koko organisaatiossa kuuluu myös valtuutetun toimenkuvaan. Laadunhallintajärjestelmän toimivuuden seuraamiseen johdon on laadittava suunnitelma, johdonkatselmuksot, joka toteutetaan ennalta määrättyinä ajankohtana. Katselmuksot tarkoituksena on varmistaa, että järjestelmä on edelleen toimiva, jos laatupolitiikkaan tai laatutavoitteisiin on tarve tehdä muutoksia, ne tehdään katselmuksot seurauksena. (SFS 2008) Tässä opinnäytetyössä ei oteta kantaa laatujärjestelmän auditointiin eikä serfiointiin, koska laatujärjestelmä oli tämän työn tekohetkellä vielä suunnitteluasteella.

5.2 Jatkuva parantaminen

Prosessimaisen ajattelumallin lähtökohtana ovat aina yleensä asiakkaan tarpeet. Asiakastyytyväisyyttä seuraamalla pysytään ajan tasalla siitä, onko yritys pystynyt vastamaan asiakkaiden tarpeisiin. (SFS 2008).

Alla olevassa kuviossa 7 on esitetty prosesseihin perustuva laadunhallintajärjestelmän malli, jota voidaan soveltaa kaikissa yrityksen prosesseissa. Mallia kutsutaan nimellä PDCA eli Suunnittele, Toteuta, Arvioi, Toimi.



Kuvio 7. PDCA, (SFS 2008).

Suunnitteluvaiheessa työlle asetetaan tavoitteet ja määritetään siihen tarvittavat prosessit. Tavoitteena on täyttää asiakkaan vaatimukset ja että yritys toimii toimintapolitiikkansa puitteissa. Toteutusvaiheessa on tarkoitus toteuttaa prosessit. Arviointivaiheessa seurataan ja mitataan prosesseja ja tuotteita. Sen jälkeen niitä verrataan tavoitteisiin, asiakkaan vaatimuksiin ja yrityksen omaan toimintapolitiikkaan. Edellä mainituista toimenpiteistä tehdään erillinen raportti. Toimintavaiheessa suoritetaan tarvittavat toimenpiteet, joilla prosessien jatkuva parantaminen saavutetaan (SFS 2008).

6 YHTEENVETO JA POHDINTA

Laatujärjestelmä rakentuu Neocodex Oy:lle liiketoiminnan tarpeista käsin ja on kehitetty liiketoimintaa palvelemaan. Tämän opinnäytetyön antaman pohjustuksen avulla laatujärjestelmäprojektissa on helpompi löytää yrityksen toiminnassa olevat kehitystarpeet ja panostaa niihin. Työn keskittyessä testaukseen liittyviin toimintoihin yritys saa valmiit periaatteet käyttöönsä. Haasteena on jalkauttaa ne käytäntöön ja toimia niiden mukaisesti. Työntekijät on hyvä saada ymmärtämään yhtenäiset toimintatavat, jotka tuovat arvoa asiakkaille.

Projekti etenee syksyn 2013 aikana suunnitteluasteelta varsinaisiin toimenpiteisiin laatujärjestelmäprojektin vetäjäksi valitun toimihenkilön eteenpäin viemänä. Yrityksen henkilökunnalle on pidetty laatujärjestelmään liittyviä esitystilaisuuksia asiaan perehtyneiden ammattilaisten johdosta. Tarkoitus on saada johto ja henkilökunta sitoutumaan laatujärjestelmän kehittämiseen. Neocodex Oy:n yhteyshenkilön pyynnöstä opinnäytetyössä on painotettu testausta ja johdon vastuuta laatujärjestelmän luomisessa.

Opinnäytetyön tekemiseen toi haasteita suhteellisen tiukka aikataulu ja projektin aloituksen viivästyminen. Pääprosessien kuvaaminen osoittautui haasteelliseksi varsinkin johtamisen ja asiakkuuksien hallinnan osalta, koska valmista prosessiajattelumallia ei yrityksellä ollut. Siksi edellä mainitut prosessit on kuvattu yleisellä tasolla tukemaan niiden määrittämistä projektin edetessä. Laadunparantaminen ja ohjelmistoalalla käytetty 9001-standardin mukaiset vaatimukset teorian osalta sisältyvät opinnäytetyöhön. Testauksen nykytilanne ja siihen liittyvä yksityiskohtainen teoria käydään työssä läpi mikä olikin yrityksen toiveena. Kehityskohteet tuotiin ilmi verraten nykytilannetta ja teoriaa, joten siltä osin työn tavoitteet saavutettiin.

Neocodex Oy:n yhteyshenkilö koki työn erittäin hyödylliseksi, koska heillä ei yrityksenä ollut selkeää kokonaiskuvaa siitä, mitä laatujärjestelmä pitää sisällään. Opinnäytetyö toimii hyvänä alkusysäyksenä ja dokumenttina yritykselle projektin käynnistyessä. Neocodex Oy:n tulevaisuus tulee varmasti näyttämään entistä paremmalta laatujärjestelmän käyttöönoton myötä.

LÄHTEET

Neocodex Oy:n kotisivut. <http://www.neocodex.fi> Viitattu 10.4.2013

Haikala I. & Märijärvi J, 2004. Ohjelmistotuotanto. 10. uudistettu painos. Talentum Media Oy, Hämeenlinna.

Lecklin, O. 2002. Laatu yrityksen menestystekijänä. 4. uudistettu painos. Talentum Media Oy, Jyväskylä.

SFS. ISO 9001 (2008). ISO 9001:2008. Suomen standardoimisliitto. Helsinki.

Itä-Suomen Yliopisto 2013. <http://cs.joensuu.fi/tSoft/laadunhallinta.htm> Viitattu 31.5.2013

Laamanen K. 2001. Johda liiketoimintaa prosessien verkkona. Suomen Laatu keskus Koulutuspalvelut Oy. Keuruu.

JHS: <http://docs.jhs-suositukset.fi/jhs-suositukset/JHS182/JHS182.html> [Viitattu 05.06.2013]

Laamanen K. & Tinnilä M, 2009. Prosessijohtamisen käsitteet. 4. Uudistettu painos. Teknologiainfo Teknova Oy. Espoo.

LIITTEET

ncxLaatu / Testaus ja laadunvarmistus

Tausta: Neocodex Oy:ssä on käynnissä toiminnan ja laadun kehittämishanke ns. ncxLaatu-projekti. Projekti on tiivistetysti esitetty projektiesitteessä (projektiesitys.pdf). Näiden seuraavien kysymysten tarkoituksena on kartoittaa nykyistä testauskäytäntöä, dokumentointia ja laadunvarmistusta eri Neocodexin toiminnoissa. Kysymyksissä keskitytään testaukseen.

1. Miten Neocodexin tekemien ohjelmistojen ja toimitusten virheettömyys varmistetaan tällä hetkellä?
2. Kuvaile testausta ja sitä miten virheitä karsitaan kehitteillä olevasta ohjelmasta? (kuvaus voi olla teksti, vuokaavio tms.)
3. Miten varmistetaan että ohjelma vastaa asiakkaan vaatimuksia? Miten määrittelyvaiheen asiakasvaatimukset siirtyvät ohjelmiston toteutukseen?
4. Millaisia suunnitteludokumentteja on käytössä?
5. Miten testaus suunnitellaan tällä hetkellä? Mistä saadaan testaus suunnitelma?
6. Miten testauksen tulokset dokumentoidaan tällä hetkellä?
7. Yleistä asiaa. Mitä mielestäsi tässä vaiheessa olisi hyvä huomioida laatuprojektissa?

Huom!

Vastaukset ovat tärkeä osa nykyisen toiminnan kartoitusta. Vastaukset käsitellään luottamuksellisesti. Vastauksia ei julkaista missään ennen kuin on saatu yrityksen johdon lupa jonkin asian julkaisemiseen.

www.savonia.fi

