

**JIRA SERVICE MANAGEMENT -OHJELMISTON  
KÄYTTÖÖNOTTOPROJEKTI**



Ammattikorkeakoulututkinnon opinnäytetyö

Tietojenkäsittelyn koulutus, Hämeenlinnan korkeakoulukeskus  
syksy, 2021

Susanna Rajanen

---

Tekijä Susanna Rajanen

Vuosi 2021

Työn nimi Jira Service Management -ohjelmiston käyttöönottoprojekti

Ohjaajat Lauri Salminen, Lasse Seppänen

---

## TIIVISTELMÄ

Opinnäytetyön aiheena oli Jira Service Management -ohjelmiston käyttöönottoprojekti Lifecycle Services -tiimin tikettienhallinnan kehittämiseksi ohjelmisto- ja IT-palvelualan yrityksessä. Jira Core -versio on ollut yrityksessä käytössä jo vuosia, mutta yritykseen liittyneellä service deskillä vasta lyhyen aikaa. Käyttöönottoprojektin päätavoite oli yrityksen Lifecycle Services -tiimin tikettienkäsittelyprosessin yhdenmukaistaminen ja tehostaminen sekä toimintaohjeiden käytettävyyden parantaminen. Työn teoriaosuuden sisältö koostuu pääosin service desk -organisaation esittelystä, Lean-prosessinohjausmallin ja Jira-järjestelmän teoriasta. Teoriaosuudessa käsitellään myös Jiran lisäosan dokumentaatiohallintaan olennaisesti kuuluva Atlassian Confluence.

Opinnäytetyön toimeksiantaja oli Bilot Oyj. Opinnäytetyön toteutus aloitettiin Lifecycle Services -organisaation projektinvetäjän toiveiden kartoituksella ja keskusteluilla loppukäyttäjien kanssa. Kartoituksen jälkeen hankittu Jiran lisäosa konfiguroitiin testikäyttöön ja projektin testaajat testasivat lisäosan ominaisuuksia. Testauksen jälkeen lisäosaa muokattiin testauksessa tehtyjen havaintojen perusteella paremmin organisaation käyttöön soveltuvaksi ja tämän jälkeen lisäosa pilotoitiin osalla asiakkuuksista. Pilotoinnin aikana havaittujen huomioiden perusteella kehitettiin lisäosan konfigurointia vastaamaan mahdollisimman hyvin käyttäjien tarpeita. Pilotoinnin jälkeen lisäosan käyttö laajennettiin koskemaan kaikkia mahdollisia jatkuvien palveluiden asiakasprojekteja.

Käyttöönottoprojektin aikana havaittiin, että lisäosassa on prosessoinnin ja dokumentaation kannalta erinomaisia ominaisuuksia, mutta lisensoinnista johtuvien rajoitusten vuoksi lisäosan kaikkia ominaisuuksia ei saatu hyödynnettyä suoraan asiakasprojektien käyttöön ja jotkin lisensointiin liittyvät rajoitukset hankaloittivat tikettienkäsittelyprosessia. Tulevaisuudessa ongelmat poistunevat lisensointiresurssien korjauksella.

Avainsanat Service Desk, Jira, Lean, käyttöönottoprojekti

Sivut 51 sivua ja liitteitä 1 sivua

---

Author Susanna Rajanen

Year 2021

Subject Jira Service Management deployment project

Supervisors Lauri Salminen, Lasse Seppänen

---

## ABSTRACT

The commissioner for this thesis was Bilot Oyj. The subject of this thesis is Jira Service Management project deployment for Lifecycle Services use. The point of implementing Jira Service Management expansion to existing Jira Core application in use was to improve the service desk issue handling process with Service management application. The goal was to centralize the service desk instructions and documents to Jira Service Management knowledge base and to enhance the time used to handle customer requests.

Theory part of this thesis consists of theory of service desk organization and service management, theory of Lean process model, theory of change process and also application related theory of Jira and Confluence, both applications related to Jira Service management project.

The project started with planning sessions and end user discussions and main goals were mapped by project manager. After planning Service Management expansion was acquired and configured for test use and tested by Jira admins and end users. After test phase Service Management expansion was configured further to respond test findings. After further configuration expansion was piloted with suitable customer projects and findings made during pilot phase were used to modify system to suit better for the end users. After pilot Service Management expansion use was extended to other customer projects.

During implementation project some excellent features were discovered especially regarding to ticket handling process automation and document management, but due to license resources some expansion features were not available for current customer project use and some workarounds were devised to tackle these problems. In the future problems will be reduced when more licenses are available.

Keywords Service Desk, Jira, Lean, implementation project

Pages 51 pages and appendices 1 page

## Sanasto

Tiketti	Asiakkaalta tuleva viesti, pyyntö tai ilmoitus, jota käsitellään antamalla sille käsittelynumero ja välittämällä se tiketöintijärjestelmään, jossa siihen liittyvää prosessia voidaan koordinoida ja seurata
Ensivaste	Tiketinkäsittelijän ensimmäinen vastaus asiakkaalle, että tämän lähettämä pyyntö on otettu käsiteltäväksi
JIRA	Tiketöintijärjestelmä, ongelmanseurantaohjelmisto, projektinhallintaohjelmisto
Jira Service Management	Jira-tiketöintijärjestelmän service desk -palveluille suunniteltu ITSM-työkalu
ITSM	IT service management eli IT-palvelunhallinta
Confluence	Dokumentaationhallintajärjestelmä
Service desk	Asiakaspalvelukeskus, helpdesk, asiakaspalvelun kontaktipiste asiakkaalle
Lifecycle Services	Jatkuvat palvelut, ylläpito
Lean	Lean-malli
ITIL	Kokoelma IT-alan käytäntöjä palveluiden hallintaan ja johtamiseen

## Sisällys

1	Johdanto .....	1
2	Service desk .....	3
2.1	Service deskin toiminta .....	5
2.2	Palvelutasosopimus .....	6
2.3	Laadunhallinta ja service desk .....	6
3	Lean .....	8
3.1	DMAIC-ongelmanratkaisuprosessi.....	9
3.2	Lean asiantuntijatyössä.....	10
3.3	Lean-ajattelu ohjelmistokehitysalalla .....	11
3.4	Hukka Lean- ja Agile-organisaatioissa.....	12
4	Muutosprosessi ja -vastarinta .....	14
4.1	Toimivan muutoksen keskeiset osa-alueet.....	14
4.2	Muutosinnon löytäminen ja muutoksessa tukeminen .....	18
5	Atlassian – Jira ja Confluence .....	20
5.1	Jira Core.....	20
5.2	Jira Service Management.....	21
5.2.1	Automaatiot .....	21
5.2.2	Sähköposti-ilmoitukset.....	22
5.2.3	Kalenterit ja SLA .....	24
5.2.4	Jonot.....	24
5.2.5	Tietämyskanta .....	24
5.2.6	Asiakasportaali .....	25
5.2.7	Asiakastyytyväisyyskysely .....	25
5.2.8	Käyttöoikeudet.....	26
5.3	Confluence .....	26
6	Jira Service Management -projektin suunnittelu ja toteutus .....	27
6.1	Yrityksen tikettienkäsittelyprosessi .....	28
6.2	Projektin kulku .....	30
6.2.1	Testiprojektin rakentaminen – projektityyppi .....	31
6.2.2	Tikettityypit .....	32
6.2.3	Työnkulku .....	33
6.2.4	Näytöt, kentät ja prioriteetit.....	34
6.2.5	Asiakasoikeudet .....	35

6.2.6	Asiakasilmoitukset.....	36
6.2.7	Automaatiot .....	36
6.2.8	Tietämyskanta .....	37
6.2.9	Jonot ja Dashboard.....	38
6.2.10	Asiakastyytyväisyyskysely .....	38
6.2.11	Asiakasportaali .....	39
6.3	Pilotointivaihe .....	39
6.4	Käyttöönottovaihe .....	40
6.5	Jatko ja seuranta .....	41
7	Johtopäätökset ja pohdinta.....	42
8	Yhteenveto .....	46
	Lähteet.....	47

## Kuvat

Kuva 1	DMAIC-prosessin vaiheiden linkittyminen .....	9
Kuva 2	Jira Core -version perustyönkulku .....	21
Kuva 3	Jiran automaatiösäätöesimerkki Atlassianin sivulta (Atlassian n.d.-s).....	22
Kuva 4	Yrityksen tikettienkäsittelyprosessi .....	29
Kuva 5	Yrityksen tikettienkäsittelyprosessi projektin valmistuttua .....	31
Kuva 6	Jira Service Management -projektityypit.....	32
Kuva 7	Testiprojektin pyyntötyypit (Request types) .....	33
Kuva 8	Service Request Fulfillment -työnkulku Jirassa.....	33
Kuva 9	Testiprojektin lopullinen työnkulku Jirassa .....	34
Kuva 10	Testiprojektin prioriteettityypit Jirassa .....	35
Kuva 11	Osa testiprojektin "Comment added" -automaatiösäätökaaviosta .....	37

## Liitteet

Liite 1	Aineistonhallintasuunnitelma
---------	------------------------------

## 1 Johdanto

Toiminnallisen opinnäytetyön päätavoitteena oli eheyttää ohjelmistoyrityksen b-2-b-service deskin tikettienkäsittelyprosessia ottamalla käyttöön jo käytössä olevaan Jira-tikettienhallintajärjestelmään palvelunhallintaohjelmisto Jira Service Management.

Lähtötilanteessa Jira oli ollut yrityksen Lifecycle Services -käytössä tikettijärjestelmänä jo monia vuosia, mutta osalla Lifecycle Services -organisaatioon hiljattain liittyneistä jatkuvista palveluista vasta vähän aikaa. Käyttöönottoprojektin alkaessa lisäosa hankittiin vastaamaan havaittuihin epäkohtiin service deskin tikettienkäsittelyprosessissa. Ennen projektia käydyissä keskusteluissa oli ilmennyt, että osa Lifecycle Services -organisaatioon kuuluvan service deskin työntekijöistä koki Microsoft Outlookista käsin tapahtuvan tikettien vastaanoton ja seurannan haastavaksi ja työlääksi lukuisien seurattavien sähköpostikansioiden vuoksi. Tämän lisäksi tikettien käsittelyn tueksi tehdyt ohjeistukset sijaitsivat hajallaan erilaisissa yksittäisissä tiedostoissa lukuisissa Microsoft Teams-alustan kansioissa sekä osin jo myös dokumentaationhallintajärjestelmässä Confluencessa, joten organisaatiossa oli tarve yhdenmukaistaa ohjeiden dokumentaationhallintaprosessia ja kehittää tehokkaampia tapoja säilyttää tikettienkäsittelyohjeita käyttäjien saatavilla.

Käyttöönottoprojektin päätavoite oli auttaa yrityksen service deskiä siirtymään monia manuaalisia työvaiheita sisältävästä viestien käsittelystä yhdelle alustalle Jiraan, jolloin tarve käyttää samanaikaisesti sekä sähköpostiohjelmistoa, Teamsia, erilaisia dokumentteja että tiketinhallintajärjestelmää vähenisi. Viestien käsittelyyn toivottiin lisää sulavuutta Jira Service Managementin lisäksi siihen kuuluvan Confluence-tietämyskantaintegraation avulla, jolla ohjeistukset saataisiin keskitetyksi yhteen helposti tavoitettavaan paikkaan. Useammasta erilaisesta tikettienkäsittelyn tukena olevasta ohjelmistosta (Outlook, Word, Excel, Teams ja Jira) yhden ohjelmakokonaisuuden pääasialliseen käyttöön siirtymisestä toivottiin saatavan hyötyä sekä service deskin turhien manuaalisten työvaiheiden vähentämisessä, tiedonhaun helpottumisessa sekä näiden kautta service deskin tikettienkäsittelytoiminnan tehostamisessa. Laajempaan kokonaisuuteen toiminnan tehostamisen ajatuksessa oli tikettikäsittelijöiden ajan vapautuminen servicedesk-toiminnan kehittämiseen ja moniulotteisempiin asiantuntijatehtäviin.

Opinnäytetyön toimeksiantaja oli Bilot Oyj. Ohjelmistoyritys Bilot Oyj perustettiin vuonna 2005 ja yritys kehittää pääasiallisesti liiketalouden ratkaisuja sekä pilvipalveluita SAP- ja Microsoft-tuotteiden avulla. Bilot Oyj on kansainvälinen ohjelmistoalan yritys, jolla on toimistoja Suomessa, Puolassa ja Ruotsissa. Kasvuun tähtäävällä yrityksessä on tällä hetkellä töissä noin 200 IT-alan ammattilaista.

Käyttöönottoprojektin toteuttamiseen kuului ohjelmistotoimittajan asentaman ohjelmiston käyttöönotto tarvittavine testiprojekteineen, testauskierroksineen, testiprojektien muokkauksineen sekä testausten jälkeinen pilotointi ja lopullinen käyttöönotto asiakaskohtaisesti sekä servicedesk-henkilöstön vaiheittainen perehdytys lisäosan käyttöön, prosessi- ja käyttöohjeen laatiminen sekä dokumentoiminen.

Käyttöönottoprojektin lopuksi projektin onnistumista arvioitiin Jira-pääkäyttäjien, lopputuotteen käyttäjien ja tiiminvetäjän antaman palautteen ja kehitysehdotusten perusteella, tikettienkäsittelyprosessiin tehtyjen muutosten ja käytettävyyden sekä lisäosan kehitystyön tulevaisuudennäkymien pohjalta.

Tutkimuskysymykset:

- Mikä on servicedesk ja mitä se tekee kohdeyrityksessä?
- Mikä on toimiva service desk-prosessi? (toimiva palveluprosessi, toimiva muutosprosessi)
- Minkälainen on Lean-prosessinohjausmalli?
- Miten toteutetaan Bilot Oyj:n service desk -toimintojen yhdenmukaistaminen Jiran avulla?



## 2 Service desk

Service desk- tai service desk -nimityksellä on useita synonyymejä esimerkiksi asiakaspalvelukeskus, helpdesk, palvelukeskus tai asiakastuki. Tässä opinnäytetyössä asiakaspalvelukeskuksesta käytetään toimeksiantajan käyttämää nimitystä service desk. Service desk on asiakkaan ja palveluntuottajan välinen kontaktipiste. Tyypillinen service desk hoitaa häiriötapahtumia ja palvelupyynnöitä sekä kommunikoi käyttäjien kanssa. Service desk on asiakaspalvelukeskus, jossa asiakkaat saavat apua IT-palveluntuottajaltaan. Tavasta riippumatta service deskin tavoite on tuottaa asiakkaalle laadukasta palvelua oikeaan aikaan. (Atlassian, n.d.-a) Service desk on organisaation kontaktipiste, joka määrittelee asiakas- tai käyttäjäkokemuksen ja voi myös toimillaan joko pilata, että parantaa sen. Service desk on käyttäjien ja kehityksen välillä linkki, joka tuntee liiketoiminnan ja asiakkaan. Parhaat service deskit ymmärtävät yhteisen arvon lisäämisen tärkeyden ja paras arvo on asiakas, joka kokee saaneensa service deskiltä hyvän asiakaspalvelukokemuksen. (McCullough, 2020)

Service desk on keskitetty yhteyspiste asiakkaan, käyttäjien, IT-palveluntuottajien ja kolmansien osapuolten välillä. Service desk vastaa häiriönhallinnasta (Incident Management) ja palvelupyynnöprosesseista ja sen päätavoite on palauttaa palvelu normaalitilaan mahdollisimman nopeasti. Service desk on asiakkaan tärkein kontaktointipiste, joten sen vuoksi service deskin toiminnalla on suuri vaikutus asiakastyytyväisyyteen ja -tyytymättömyyteen. (Jäntti, Shrestha & Cater-Steel, 2012, s. 203). Service deskin tehtävät voivat olla häiriötapahtumia, palvelupyynnöitä, valituksia tai palautteita. Service deskin työntekijät voivat myös tunnistaa muutospyyntöjä ja ongelmia. Käytännössä IT-service desk tuottaa asiakkaille ja käyttäjille tukea erilaisiin IT-palveluihin. (Jäntti, Shrestha & Cater-Steel, 2012, s. 203) On service deskin tärkeimpiä tehtäviä kommunikoida tehokkaasti asiakkaiden kanssa. Aktiivinen kommunikointi takaa yleensä paremman asiakaspalvelukokemuksen. Tämä sisältää asiakkaiden pitämisen tietoisena heidän lähettämiensä palvelupyynnöiden tilanteesta. Asiakaskokemus on vuorovaikutusta organisaation ja asiakkaan välillä. IT-palveluiden hallinnassa service desk on asiakaskokemuksen rajapinta. (Jäntti, Shrestha & Cater-Steel, 2012, ss. 203–204)

Service Desk and Incident Manager -nimisen teoksen kirjoittaneen Peter Wheatcroftin mukaan IT-organisaation toimintona palvelu ei ole sama kuin tuotettu tuote. Sitä ei voi tuottaa massana, pakata uusiin paketteihin eikä jakaa eri kanavia pitkin, sillä palvelu on dialogia asiakkaan ja palvelutuottajan välillä. Koska palvelu on se osa-alue IT-palvelutuotantoa, josta monesti valitetaan, on Wheatcroftin mielestä tärkeää määritellä palvelulle tavoitettavat raamit. Service deskiin otetaan yleensä yhteyttä siinä vaiheessa, kun prosessit tai teknologia eivät toimi asiakkaan näkökulmasta tavoitellulla tavalla. Kyseessä on usein negatiivinen tilanne, joten tilanteen hoitaminen vaikuttaa asiakastyytyvyyteen. Wheatcroftin mielestä asiakkaan luottaessa service deskiin ja suhtautuessa positiivisesti saamaansa palveluun, se vaikuttaa koko IT-organisaatioon. Näin ollen käynnissä oleva, toimiva palvelusuhde on paljon tärkeämpi asiakkaalle kuin palvelun muotoilu. Hyväkään tuote ei kanna, ellei palvelutiimillä ole resursseja ja taitoja saada tuotetta toimimaan kunnolla. Huono palvelukokemus heijastelee negatiivisesti tuotteeseen. (Wheatcroft, 2014, ss. 2–4) Sulava käsittely on olennaista palvelutoiminnolle jo senkin takia, että palvelu, jonka asiakas kokee hitaaksi, lisäarvottomaksi ja ratkaisukyvyttömäksi, voi saada asiakkaat kiertämään service desk -palvelut tavoittelemalla suoraan ratkaisukonsultteja (Wheatcroft, 2014, s. 16).

Nykyajan informaatioyhteiskunnassa asiakas vaatii jatkuvasti enemmän yrityksiltä. Ensisijaiset vaatimukset ovat tasokas asiakaspalvelu ja edullinen hinta eikä asiakaspalvelusta välttämättä haluta maksaa. Sen sijaan yritysten tavoite sekä menestymisen edellytys on voiton tuottaminen, joten näitä kahta yhtälöä ei ole helppo sovittaa yhteen. Tasokas asiakaspalvelu edellyttää yleensä riittävää henkilökuntaa asiakkaan vaatiman palvelutason toteuttamiseksi. (Aarnikoivu, 2005, s. 14)

Automaation kehittyessä service deskin fokus on siirtymässä teknisten tilanteiden hoitamisesta ihmisten ja liiketoiminnan tukemiseen. Service deskiä käytetään myös muuhun kuin rikkoutuneiden asioiden hoitamiseen. Sen tarkoitus on monesti koordinoita, ohjeistaa, järjestellä asioita ja toimia muutenkin tärkeänä osana palveluoperaatioita. Siinäkin tapauksessa, ettei service desk ole toiminnoiltaan erityisen tekninen, se on tärkeässä roolissa palvelun tuottamisessa ja sitä tulisi aktiivisesti tukea yhteistyöryhmien taholta juuri sen suuren käyttäjävaikutuksen vuoksi. Toinen syy service deskin tärkeyteen on sen käytännön ymmärrys laajemmasta liiketoimintayhteydestä ja -prosesseista sekä käyttäjistä. Service desk ei tuo lisäarvoa pelkästään häiriötilanteiden kirjaamisella, vaan myös ymmärtämällä ja toimimalla asiaan liittyvien

liiketoimintakontekstien mukaan. Service deskin tulisi olla empaattinen ja hyvin informoitu linkki palveluntuottajan ja käyttäjien välillä. (Axelos, 2019 s. 378)

## 2.1 Service deskin toiminta

Service deskiin liittyy lukuisia erilaisia yhteydenpitokanavia lähtien aina puhelinsoitoista, palveluportaaleista, mobiilisovelluksista, chat-ohjelmista, sähköpostista, sosiaalisen median työkaluista, keskustelufoorumeista aina kiinteisiin palvelupisteisiin (Axelos, 2019 s. 379). Joissakin service deskeissä palvelua tarjotaan tietyssä palveluaikana, mikä voi vaatia henkilöstöltä vuorotyöskentelyä palvelun tuottamisen takaamiseksi. Service desk voi olla keskitetty palvelu tietyssä lokaatiossa tai se voi olla virtuaalinen, jolloin service deskin työntekijät voivat työskennellä monista eri paikoista käsin. (Axelos, 2019, ss. 380–381)

Service deskin henkilökunnalta vaaditaan koulutusta ja osaamista monilta teknisiltä ja liiketoiminnallisilta alueilta. Erityisesti service deskiltä vaaditaan osaamista asiakaspalvelukyvyistä, kuten empatiasta, häiriötilanteiden analyysistä, priorisoinnista, tehokkaasta kommunikoinnista sekä emotionaalista älykkyydestä. Tärkeä kyky on kyky ymmärtämään ja diagnosoimaan häiriötilanteet liiketoiminnan prioriteetin perusteella käyttämällä apuna tarjolla olevia kykyjä, tietoa, ihmisiä ja prosesseja. (Axelos, 2019 s. 381)

Service desk tuo ITIL Foundation 4:n periaatteiden mukaisesti arvoa palvelun arvoketjuun. Arvoketjun pääkohdat ovat kehittyminen, sitoutuminen, suunnittelu ja muutos, saavuttaminen tai rakentuminen sekä toimittaminen ja tuki. Service deskissä kehittyminen tarkoittaa ITILin mukaan sitä, että toimintoja monitoroidaan ja arvioidaan keräämällä asiakaspalautteita jatkuvan kehityksen mahdollistamiseksi. Näin tuetaan jatkuvaa kehitystä, linjausta ja arvon luontia. Sitoutuminen puolestaan tarkoittaa sitä, että Service desk on pääkanava taktiseen ja operatiiviseen yhteydenpitoon käyttäjien kanssa. Suunnittelu- ja muutosvaiheessa service desk tuottaa ITILin mukaan käyttäjille kommunikaatiokanavan koskien uusia ja muuttuneita palveluita, kun taas saavuttamisen ja rakentumisen vaiheessa service deskin henkilökuntaa voidaan käyttää ratkaisemaan palvelupyyntöjä ja häiriötilanteita. Toimittaminen ja tuki tarkoittaa ITIL-

näkökulmasta puolestaan sitä, että service desk on koordinoitipiste häiriötilanteille ja palvelupyynnöille. (Axelos, 2019, s. 382)

## 2.2 Palvelutasosopimus

Palvelutasosopimus eli Service Level Agreement (SLA) on dokumentoitu sopimus palvelun tuottajan ja asiakkaan välillä ja se kuvaa sekä vaaditut palvelut että odotetun palvelutason (Axelos, 2019 s. 384). ITILin mukaan SLA-sopimuksessa asiakas on osapuoli, joka maksaa palvelusta ja käyttäjä on osapuoli, joka käyttää palvelua. Service deskin SLA sisältää tavanomaisesti palvelutavoitteita vastausajan, ratkaisuaajan ja asiakastyytyväisyyden suhteen. (Hjelt & Syynimaa, 2021, s. 3) Palvelutasosopimukset ovat pitkään olleet työkaluja suorituksen mittareita asiakasnäkökulmasta (Axelos, 2019 s. 384). Palvelutasonhallinta tunnistaa mittarit ja laskelmat, jotka antavat todellisen kuvan asiakkaan kokemuksesta ja tyytyväisyystasosta (Axelos, 2019 ss. 387) .

## 2.3 Laadunhallinta ja service desk

Laadunhallintaa juuri service deskin johtajien näkökulmasta tutkineet Hjelt ja Syynimaa määrittelevät kirjoituksessaan, että IT-palveluiden tuottajille on olennaista pitää asiakas tyytyväisenä ja tässä tavoitteessa service deskin toiminta on tärkeässä asemassa. Korkealaatuisen palvelun tuottaminen on ratkaisevaa asiakastyytyväisyyden kannalta. (Hjelt & Syynimaa, 2021, s.2) Laadunhallinnan tarkoitus on kasvattaa organisaation tehokkuutta ja sen resurssien käyttöastetta. Palvelun laatu vaikuttaa asiakkaisiin joko positiivisesti tai negatiivisesti. Asiakas ei kuitenkaan tuota keinoja palvelun laadun parantamiseksi, vaan palvelun tuottaja tekee sen keräämäänsä dataa hyödyntämällä. Palvelun laatua on hankala mitata, koska palvelun laatu eli erinomaisuus on yhtä kuin tavoitteiden saavuttaminen tai ylittäminen ja sen mittaaminen vaatii sekä prosesseja että palvelun lopputulemia. ITILin mukaan kriittiset menestystekijät on määritelty kaikissa IT-palveluissa. Kriittinen menestystekijä tulee saavuttaa, jotta palvelu onnistuu. Suorituskykyilmaisinta käytetään määrittelemään onko kriittiset menestystekijät saavutettu. Jokainen suorituskykyilmaisoin tulisi olla erityinen, laskettava, saavutettava, olennainen ja aikaan

sidottu. Suorituskykyilmaisimet tulisi määritellä varovaisesti, jotta ne johtavat toivottuihin lopputuloksiin. (Hjelt & Syynimaa, 2021, ss. 3–4)

Service deskin johtajia tutkimuksessaan haastatelleet Hjelt ja Syynimaa määrittelivät keräämiensä tulosten pohjalta, miten kyseessä olevan kansainvälisen ja laajan organisaation service deskin johtajat näkevät laadun käsitteenä ja kuinka laadunhallintaa tehdään. Kyseessä olevan organisaation koon ja laajuuden perusteella tutkimusta voi Hjeltin ja Syynimaan mukaan pitää katsauksena alan tilasta, vaikkei tuloksia voi välttämättä suoraan siirtää muihin konteksteihin. Lähes kaikki Hjeltin ja Syynimaan tutkimukseen vastanneista service deskin johtajista olivat sitä mieltä, että laatu määritellään SLA-tavoitteiden saavuttamisen perusteella. Kun SLA on saavutettu, asiakas on tyytyväinen. Palvelun laatu nähtiin tutkimuksen perusteella palvelun lopputuloksena. Palvelun tuottamisen prosessien ei ymmärretty vaikuttavan laatuun. Laadun määrittely vaikutti myös kriittisten menestystekijöiden määrittelyyn, sillä laadunhallinta keskittyi tutkimukseen vastanneilla palvelun lopputuleman hallintaan resurssien kautta eli käytännössä kriittiset menestystekijät olivat tutkimuksen mukaan sellaisia, jotka vaikuttivat SLA-tavoitteiden saavuttamiseen. Hjeltin ja Syynimaan mukaan rajoittunut käsitys laadusta johtaa siihen, että laatua mitataan ainoastaan palvelutason perusteella, sillä ihmiset tapaavat keskittyä asioihin, joiden perusteella heitä mitataan. Mittaamalla laatua ainoastaan palvelutasojen toteutumisen perusteella keskitytään ainoastaan kapasiteettiin ja ihmisten taitoihin ja vaikka palvelun lopputulos voi parantua, palveluiden jatkuvuus voi olla puutteellista. Lisäksi keskittyminen siirtyy pois kriittisistä menestystekijöistä, mikä voi johtaa alioptimointiin. (Hjelt & Syynimaa, 2021, ss. 4–6)

### 3 Lean

Opinnäytetyön prosessijohtamismalli perustuu Lean-konseptiin. Lean-sanan käyttö perustuu Lean-tuotanto -termiin (Lean Production), joka tuli tunnetuksi japanilaisten autotehtaiden tuottavuuden parannusta käsittelevästä kirjasta ”The Machine that Changed the World” vuodelta 1990. Lean-tuotanto on tapa valmistaa asioita ja konsepti on alun perin lähtöisin Fordin tuotantoideasta vuodelta 1900, josta se kulkeutui Toyotalle ja myöhemmin teoriaksi Lean-tuotannosta. Itse termi Lean syntyi John Krafickin keksinnöstä hänen todetessaan ”Se (Toyota Production System) käyttää vähemmän kaikkea luodessaan saman määrän arvoa, joten kutsukaamme tätä LEAN:ksi”. Nykyään Lean on Six Sigman mukaan laatujohtamisen periaatteiden soveltamista tuottamiseen.

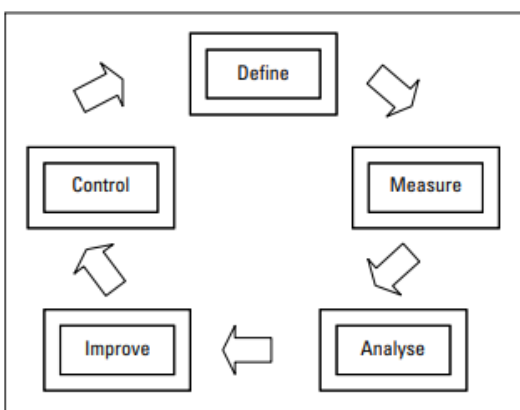
Tavoitteena on keskittyä kokonaisuuden optimoimiseen sekä tuottaa asiakkaalle parasta mahdollista arvoa. (Six Sigma, n.d.) Lyhyesti Lean näkee resurssien tuhlauksen kaikissa muissa tapauksissa paitsi laadun tuottamisessa loppukäyttäjälle turhana ja eliminotavana asiana, sillä Lean on systemaattinen lähestymistapa tunnistaa ja eliminoida hukka jatkuvan kehityksen kautta virtauttamalla tuotetta asiakkaan vaatimusten mukaisesti (Cobb, 2011, s. 22).

Lean-ajattelutapa nähdään monesti tuotantokonseptina, mutta monet siihen liittyvät tekniikat ja työkalut on suunniteltu palveluorganisaatioissa (Morgan & Brenig-Jones, 2012, s. 9). Periaate on ymmärtää, miten työ tehdään, miten sitä voi tehdä paremmin, nopeammin ja sulavammin ja lyhentää prosessiin kuluva aikaa. Sama pätee kaikkiin prosesseihin. Toimiminen Leanin mukaan tarkoittaa myös ihmisten ottamista mukaan prosesseihin ja heidän luovan potentiaalinsa hyödyntämistä. Leanissa tarkoitus on auttaa heitä pystymään tehdä asioita, saada heidät tuntemaan itsensä osaaviksi sekä haastaa ja parantaa heidän prosessejaan ja työtapojaan. (Morgan & Brenig-Jones, 2012, s. 10) Olennaista Leanissa on myös se, että asiakas määrittelee arvon. Arvo on se, minkä asiakas on valmis maksamaan. Asiakkaan tarpeiden tyydyttämiseksi organisaation tulee tuottaa palveluita tai tuotteita oikeaan aikaan, oikealla hinnalla ja laadulla. Tämän onnistumiseksi yrityksen tulee tunnistaa ymmärtää, miten prosessit toimivat, parantaa ja sulavoittaa virtausta ja vähentää prosessien turhia työvaiheita sekä vähentää tai estää hukkatyötä. (Morgan & Brenig-Jones, 2012, s. 15)

### 3.1 DMAIC-ongelmanratkaisuprosessi

Lean-ajattelussa olemassa olevia prosesseja pyritään parantamaan systemaattisesti DMAIC-ongelmanratkaisuprosessilla (Define, Measure, Analyse, Improve ja Control). Suomeksi nämä ongelmanratkaisuprosessin vaiheiden nimet ovat määrittely, mittaus, analysointi, parannus ja ohjaus. Määrittelyvaiheessa kuvataan kyseessä oleva ongelma tai kehityskohde. Ilman aiheeseen liittyvää dataa määrittely saattaa perustua arvauksiin, joten mittausvaiheessa pyritään käyttämään faktoja ja tietoa prosessien ja niiden toiminnan ymmärtämiseksi, jotta ongelma pystytään kuvaamaan tehokkaammin. Analyysivaiheessa tilanne tulkitaan faktojen ja tiedon perusteella, jotta ongelman juurisyy pystytään löytämään. Kun juurisyy on löydetty, päästään parannusvaiheeseen, jossa tunnistetaan potentiaaliset ratkaisut, valitaan mahdollisista ratkaisuissa paras mahdollinen sekä testataan ja pilotoidaan sitä. Tällä tapaa pyritään vahvistamaan valittu lähestymistapa. Parannusvaiheessa käytetään tarvittaessa aiemmin kerättyä dataa vahvistamaan parannusprosessia. Parannusvaiheen jälkeen siirrytään ohjausvaiheeseen, jossa ratkaisu otetaan käyttöön. Onnistumista seurataan mittareilla ja pyritään varmistamaan se, että saavutettu kehitys pysyy eikä alkuperäinen ongelma enää palaa. Vaiheet linkittyvät toisiinsa kuvan 1 mukaisesti. (Morgan & Brenig-Jones, 2012, s. 25)

Kuva 1 DMAIC-prosessin vaiheiden linkittyminen



### 3.2 Lean asiantuntijatyössä

Toimenantajayrityksessä service desk-työ on käytännössä asiantuntijatyötä. Torkkolan Lean asiantuntijatyön johtamisessa -teoksen mukaan myös asiantuntijatyön näkökulmasta tärkeää on tavoitella työn sujuvaa etenemistä. Suurimmat esteet työn sujuvalle etenemiselle, Leanissa terminä käytetylle virtaukselle, ovat vaihtelu, ylikuormitus ja hukka. Asiantuntijatyössä vaihtelu tarkoittaa monesti työkuorman vaihtelua, henkilöstön osaamisen tasoeroja, yllättäviä kiireellisiä työtehtäviä eli niin sanottuja tulipalojen sammuttamisia tai työorganisaation toimintatavoista syntyvää vaihtelua. Tärkeimpänä esteenä virtaukselle asiantuntijatyössä on Torkkolan mukaan ylikuormitus. Ylikuormitus tarkoittaa ihmisen, järjestelmän tai jonkin laitteen ylikuormitusta. Asiantuntijatyössä se tarkoittaa henkilöstön uupumista, joka vähentää kykyä oppi uutta ja parantaa työtapoja. Ihminen on tärkeässä asemassa työn virheettömyyden ja kehittämisen onnistumisessa. Hukka puolestaan tarkoittaa tekemistä, joka ei tuota arvoa eli tekemistä, josta asiakas ei halua maksaa. Hukkaa ovat Torkkolan mukaan seuraavat työn tekemisen vaiheet: ylituotanto, asiantuntijatyössä keskeneräinen työ, odottaminen, ylimääräinen työntekijän tai materiaalin liike, siirtäminen, virheet ja uudelleen tekeminen sekä epätarkoituksenmukainen käsittely. Lisäksi henkilöstön kannalta hyvä lähestymistapa hukan poistamiseksi on tunnistaa turhia työvaiheita, tarpeettomia hyväksyntätapoja ja häiriötä aiheuttavia järjestelmiä sekä poistaa niitä yksitellen. (Torkkola, 2015, ss. 24–26)

Torkkolan mielestä asiantuntijaorganisaatiossa ei aina voi ennustaa suorituskykyä, sillä työ vaatii jatkuvaa priorisointia ja joidenkin tehtävien kiirehtimistä muiden kustannuksella. Työtä tulee monista eri lähteistä ja asiantuntijan on hallittava omaa ajankäyttöään. Paineen- ja kiireensietokyky ovat asiantuntijatyössä vaadittuja ominaisuuksia. Todellista kapasiteettia tai kysyntää on Torkkolasta hankalaa määritellä, koska osa työkuormasta tulee suoraan asiantuntijalle henkilökohtaisia reittejä pitkin. Aina kapasiteetin lisääminen ei myöskään tarkoita resurssien lisäämistä eikä tähän kustannussyistä ole yleensä mahdollisuuksiakaan. Kapasiteettia voi parantaa myös työhön käytettyä aikaa lyhentämällä esimerkiksi automatisoimalla työvaiheita. Muita vaihtoehtoja ovat prosessien sujuvoittaminen, keskeneräisen työmäärän rajoittaminen, virheiden vähentäminen, monimutkaisuuden vähentäminen ja satunnaisen vaihtelun vähentäminen. (Torkkola, 2015, ss. 78–80)



### 3.3 Lean-ajattelu ohjelmistokehitysalalla

Opinnäytetyö tuotetaan ohjelmistokehitysalalla toimivalle yritykselle. Yleisesti Leanin viisi pääperiaatetta ovat arvon tuottaminen asiakkaalle, arvovirtaketjun suunnittelu, imuohjaus (Pull) eli asiakkaan tarpeisiin perustuvan arvoketjun perustaminen, virtaus eli toteutusta hidastavien tai estävien pullonkaulojen poistaminen ja sitä kautta prosessin tehokkuuden lisääminen, ihmisten arvostaminen eli oppimisen ja ihmisten arvostamisen kulttuuri sekä täydellisyyden tavoittelu jatkuvan kehityksen muodossa. (Cobb, 2011, ss. 23–24) Nämä Lean-periaatteet ovat myös ketterän ohjelmistokehityksen pohjana. Tärkeää ajattelumallissa on päättää tuottavatko prosessissa tapahtuvat aktiviteetit arvoa. Jokaisessa prosessissa on kolmea erilaista työtä: lisäarvoa tuottavaa työtä, arvoa tuottamatonta, mutta pakollista työtä (esimerkiksi lain vaatimat toiminnot, yrityksen säännöt ja muut vastaavat syyt) sekä hukkatyötä, joka kuluttaa resursseja, mutta ei tuota minkäänlaista arvoa asiakkaan silmissä. Näiden konseptien lisääminen vaatii käytännössä prosessin vaiheiden arvioimista sen perusteella tuottavatko vaiheet arvoa asiakkaalle vai eivät. (Cobb, 2011, s.21–22)

Lean-ideologia muokattiin ohjelmistokehitysalalle Tom ja Mary Poppendieckin toimesta 2000-luvun alussa. Ohjelmistokehityksessä Lean on ensisijaisesti ajattelutapa. Ajattelutapana Lean sisältää arvoja ja periaatteita, joita kutsutaan Lean-terminologialla ”thinking tools” -nimityksellä eli ajattelutyökaluiksi. Lean-ajattelun omaksumiseksi tulee aloittaa Lean-ajattelutapa Leanin arvoista. Lean-arvoja ohjelmistokehityksen näkökulmasta ovat:

- Eliminate waste eli vähennä hukkaa. Vähennä hukkaa -termi tarkoittaa sellaisten työvaiheiden lopettamista, jotka eivät tuota arvokasta lopputulosta.
- Amplify learning eli kasvata oppimista. Kasvata oppimista arvona -vaihe tarkoittaa palautteen käyttämistä kehittymisen lisäämiseksi.
- Decide as late as possible eli päättää mahdollisimman myöhään. Mahdollisimman myöhään päättäminen tarkoittaa tärkeiden asioiden päättämistä viimeisimmässä mahdollisessa vaiheessa, kun tietoa on kerätty mahdollisimman paljon.

- Deliver as fast as possible eli toimita mahdollisimman nopeasti. Mahdollisimman nopeasti toimittaminen vaatii myöhästymisten hinnan ymmärtämistä ja nopeaan toimitukseen pyrkimistä.
- Empower the team eli voimista tiimiä. Tiimin voimistaminen tarkoittaa keskittyneen ja tehokkaan työympäristön luontia ja tiimin voimaannuttamista.
- Build integrity in eli rakenna eheyttä. Eheyden rakentaminen tarkoittaa ohjelmistokehitystä, jossa pyritään luomaan käyttäjille helposti käytettävä, järkevä kokonaisuus.
- See whole eli näe kokonaisuus. Viimeisin arvoista, kokonaisuuden ymmärtäminen, koostuu oikeasti tärkeiden asioiden havainnoimisesta, mitä projektissa tapahtuu aina huonoista puolista lähtien. (Stellman & Greene, 2014, ss. 269–271)

Learning Agile: Understanding Scrum, XP, Lean and Kanban -teoksen kirjoittaneiden Stellmanin ja Greenen mukaan olennaista Lean-ajattelussa on myös vaihtoehtoajattelu (Options thinking) ja sitoutuminen (Commitment). Lean-ajattelussa tiimit sitoutuvat tuottamaan arvoa, mutta antavat itselleen vaihtoehtoja sen toteuttamiseksi. Ketterästi toimivat tiimit voivat siis sitoutua tuottamaan arvoa, mutta määritellä itse tavan sen toteuttamiseksi. Päätöksenteko voidaan Lean-ajattelun mukaisesti jättää viimeiseen mahdolliseen ajankohtaan. Sitoutumista on tuottaa arvokas, toimiva ohjelmisto sovittuna ajankohtana, mutta vaihtoehto on toimittaa tietty toiminto tiettyinä ajankohtana. Julkaistu työlista on täynnä vaihtoehtoja, ei sitoumuksia. Vaihtoehtoajattelun tueksi voidaan käyttää myös settipohjaista kehitystapaa (set-based development), jolloin tutkitaan useampia vaihtoehtoja arvon tuottamiseksi ja lopuksi valitaan paras vaihtoehto. (Stellman & Greene, 2014, ss. 273–278)

### **3.4 Hukka Lean- ja Agile-organisaatioissa**

Lean-periaatteet voidaan nähdä myös toimenpiteinä ohjata ketterää ohjelmistokehitystä. Leanin periaatteissa kaikki aktiviteetit ja työt, jotka eivät lisää asiakkaan arvoa, ovat hukkaa. Nämä turhuudet tulisi tunnistaa ja poistaa prosesseista. Alahyarin, Gorchekin ja Svenssonin tutkimuksessa pyrittiin haastattelemaan määrittelemään, kuinka hukka määritellään, miten se priorisoidaan ja määritellään sekä miten sitä pyritään vähentämään tai poistamaan ketterissä

ohjelmistokehitysorganisaatioissa. (Alahyari, Gorschek & Svensson, 2018, s. 1) Seitsemän keskeisintä turhuutta ohjelmistokehityksessä ovat kyseisen tutkimuksen mukaan keskeneräiset työt, ylimääräiset ja tarpeettomat prosessit, ohjelman ylimääräiset ominaisuudet, tehtävästä toiseen vaihtaminen, odottaminen, liike eli vaadittu panostus ja liike kulloisenkin ongelman ratkaisemiseksi, sekä viat. Näiden turhuuksien lisäksi mukaan lasketaan myös jonkin jo unohdetun asian uudelleen opetteleminen sekä jäljelle jääneet, jollekulle toiselle siirretyt työt. (Alahyari, Gorschek & Svensson, 2018, s. 2) Sekä Leanin että Agilen (ketterän kehityksen) periaatteisiin kuuluu hukan vähentäminen. Perinteisemmissä kehitysmetodologioissa päätyökalu hukan poistamiseksi on prosessin arvioiminen ja kehitys. Tämä pitää paikkansa myös Leanissa ja Agilessa, mutta työskentelytapojen laadun päätae on Alahyarin, Gorschekin ja Svenssonin mukaan opiskeluaktiiviteettien jatkuva saavuttaminen, tietoisuus ja hukan tunnistaminen ja poistaminen hyödyntämällä esimerkiksi retrospektiivejä, eli katsausta jo tapahtuneisiin asioihin. On tärkeää optimoida kokonaisuus, koko organisaatio ja tuote koko ajan. Ihmiset ovat ohjelmistokehitysprosesseissa pääkomponentteja, joten sen vuoksi tulee aina ottaa huomioon myös ihmiset ja ihmisiin liittyvät tekijät hukan juurisyiden ja hukan vähennysstrategioiden löytämiseksi. Joka tapauksessa toisen hukka voi olla toiselle välttämättömyys ja sen vuoksi tietoisuus, määrittely, keskustelu ja yhteisymmärrys ovat edellytyksiä hukan tunnistamiseksi ja vähentämiseksi. Kaikista tärkeintä hukan määrittelemiseksi kuitenkin on keskustella ja saada yhteisymmärrys siitä, mikä on kussakin organisaatiossa hukkaa. Alahyarin, Gorschekin ja Svenssonin tutkimuksen lopputulosten mukaan hukan konsepti tunnistetaan ja sitä pidetään tärkeänä, mutta siitä keskustelemiseksi ja yhteisymmärryksen luomiseksi tehdään harvoin toimenpiteitä. (Alahyari, Gorschek & Svensson, 2018, ss. 92–93)

## 4 Muutosprosessi ja -vastarinta

Onnistu muutoksessa -teoksen kirjoittaneen Pontevan mukaan organisaatioissa koetaan pienet ja suuret muutokset hyvin eri tavoin. Muutos on kuitenkin jatkuvaa ja luonnollinen osa organisaatioiden arkea. Muutoksen toteuttaminen on aina haaste. Ponteva esittää, että esimiehen tulisi aina osata selittää muutos ja sen merkitys alaisilleen mahdollisimman ymmärrettävästi. Esimiehen lisäksi johdon tulisi sitoutua muutokseen ja osoittaa sen tarpeellisuus. (Ponteva, 2010, ss. 9–11)

Uuden järjestelmän käyttöönotto on organisaatiossa aina muutos. Muutosvastarinnasta teoksessaan kirjoittanut Palmer esittää, että muutokset eivät yleensä epäonnistu teknisistä vaan humanistisista syistä. Palmerin mukaan muutoksentekijät eivät tällöin ole ottaneet huomioon ihmisten reaktioita rutiinien muuttumiseen. Nämä epäonnistumiset luovat ajanhukkaa, tuottavuuden ja moraalien menetyksiä. Ne vaikuttavat myös niihin liiketoiminnan tavoitteisiin, joihin muutoksen oli tarkoitus vaikuttaa. Ennen muutoksen aloittamista tulisi Palmerin mielestä keskustella tiimin jäsenten kanssa valmiudesta muutokseen ja kartoittaa organisaation kyky hyväksyä tuleva muutos sekä kapasiteetti tulevaan muutokseen. (Palmer, 2004, ss. 1– 4)

Muutosvastarinnan käsittely on kriittinen onnistumistekijä muutoksessa. Yleisiä muutosvastarinnan muotoja ovat onnistuneesta muutoksesta kirjoittaneen Pontevan mukaan muutoksesta irtisanoutuminen eli muutokseen sitoutumattomuus, samaistumattomuus, epävarmuus suunnasta ja pettymys eli suuttumus ja kielteisyys. Ponteva esittää, että johdonmukaisella ja jatkuvalla muutosviestinnällä voidaan kääntää muutosvastarinta muutosta myötäileväksi toiminnaksi. Toimivia muutoksenhallinnan keinoja ovat Pontevan mukaan avoin keskustelu, tuki ja tiedotus, osallistaminen, jatkuva muutosviestintä, koulutus ja tuki sekä jatkon hallinta. (Ponteva, 2010, ss. 24–26)

### 4.1 Toimivan muutoksen keskeiset osa-alueet

Palmer on esittänyt, että heti tulevan muutoksen alusta lähtien johdon ja vastuullisen johtajan tulee tukea muutosta. Kaikissa tilanteissa tulee varmistua siitä, että muutos sopii tulevaan

ympäristöön: organisaatorakenteeseen, liiketoimintakulttuuriin, työprosesseihin ja niin edelleen. Toimivan muutoksen malliin kuuluvaksi ja suurimman onnistumisvarmuuden edellytykseksi Palmer suosittelee seuraavia osa-alueita: muutoksen johtaminen, yhteisen tarpeen luominen, vision muotoileminen, sitoutumisen jalkauttaminen, edistymisen monitorointi, työn päättäminen ja muutoksen vakauttaminen. (Palmer, 20, ss. 7–9)

Palmer kirjoittaa muutoksen johtamisen olevan olennaista muutokselle, sillä muutoksen hallinnan tuki ja kommunikaatio ovat tärkeimmät onnistumisedellytykset muutokselle. Onnistunut johtajuus vaikuttaa muihin muutoshasteisiin. Johtajuudella tavoitellaan seuraavia positiivisia lopputuloksia muutoksen suhteen: selvästi tunnistettavat johtajat antavat näkyvää, aktiivista ja julkista tukea ja sitoutumista projektille, sponsorit ja tiiminvetäjät voivat ohjata resursseja projektille, sponsorit ja tiiminvetäjät ovat halukkaita johtamaan esimerkillään, muutoksen johtajat omistavat omaa keskittymistään, aikaansa ja intoaan muutokselle sekä muutoksella on johtajuuden ansiosta selkeät tavoitteet ja toimintatavat. (Palmer, 2004, ss. 10–12)

Toimivan muutoksen mallissa Palmer suosittelee muutostarpeen selventämistä kaikille asiaan kuuluville tahoille, joihin projekti vaikuttaa. Oikein tehtynä tämä osa muutosmallista luo yhteisen omistajuuden kaikille osanottajille ja käynnistää muutosaloitteen. Yhteisen tarpeen luomisen positiiviset lopputulokset ovat Palmerin mukaan muutoksen tarpeellisuuden ymmärtäminen kaikkien asiaankuuluvien henkilöiden taholta sekä vahvistuksen asiaan kuuluvien tahojen osalta, että epätyytyväisyys nykytilaan on suurempi voima kuin muutokseen tarvittava aika, panos ja muutoksen luoma epäjärjestys. (Palmer, 2004, s. 21)

Kun ihmiset ovat myöntäneet muutoksen tarpeellisuuden, muutoksen lopputulos tulee visioida Palmer esittää. Tämä vision muotoileminen toimii ohjeena halutun lopputuloksen saavuttamiseksi. Muutosmallin tässä vaiheessa osanottajat ovat Palmerin mukaan tiedostaneet muutoksen tarpeen. Jos visio tavoitteesta puuttuu, osallistujat saattavat keskittyä enemmän ongelmiin kuin muutoksen mahdollisuuksiin. Tavoitellun vision selkeyttäminen auttaa osallistujia keskittymään ratkaisuihin. Tavoitellut lopputulokset vision muotoilemisen vaiheessa Palmerin mielestä ovat selkeä kuva muutoksen lopputuloksesta ja tulevaisuudennäkymä, joka on asiakaskeskeinen,

haastava, helppo ymmärtää, useamman kuin yhden toimijan tavoite, osallistava ja toimintaa aikaansaava. (Palmer, 2004 ss. 31–32)

Palmerin päätelmien mukaan muutosmallissa sitouttamisen jalkauttamisen vaiheessa muutoshankkeella tulee olla riittävät resurssit käytössä ja prosessin hidastamisen tai pysähtymisen ehkäisemiseksi sen tulee välttää vastustuksen herättämistä organisaatiossa. Sitouttamisen vaiheessa tavoitellut lopputulokset ovat avainhenkilöiden tunnistaminen ja käännäyttäminen, vastustuksen lähteiden tunnistaminen ja suunnitelma siihen reagoimiseksi sekä muutoksen kannattajien löytäminen. Jos muutoksen haluaa onnistuvan, Palmerin mukaan riittävä määrä henkilöitä tulee voittaa muutoksen puolelle. Keinoja sitouttamisen onnistumiseksi ovat esimerkiksi vastustuksen ymmärtäminen ja hallinta auttamalla tiimin jäseniä ymmärtämään ihmisten luonnollista reaktiota muutokseen ja tekemään suunnitelma prosessin edetessä mahdollisesti ilmenevien haasteiden voittamiseksi sekä ammatillisten erimielisyyksien hallinta ratkaisemalla eriävyyksiä ja konflikteja. Muutosvastarinta on luonnollista ja sitä tapahtuu aina muutosten yhteydessä. Palmerin mukaan ihmiset eivät niinkään vastusta muutosta kuin joutumista muutettaviksi. Muutoksen alusta lähtien muutoksen vetäjän tulee kommunikoida sen suhteen, miten muutos tulee vaikuttamaan ihmisiin. Muutos otetaan yleensä vastaan paremmin, jos ihmiset kokevat hallitsevansa sitä ja ymmärtävät, että muutoksella on hyvä tarkoitus joko helpottamalla heidän elämäänsä tai auttamalla liiketoimintaa kokonaisuutena. (Palmer, 2004, ss. 39–41)

Palmerin muutosmallissa muutosvastarinta olisi hyödyllistä hyväksyä avoimesti ja kysyä vastustajien kritiikkiä ja epäilyksiä yrittämättä kumota ne. Epäilykset ja kritiikki kannattaa ottaa harkittavaksi ja keskustella huolista avoimesti. Lisäksi muutoksen kannattajia kannattaa tukea ja heidät kannattaa ottaa mukaan muutoksen käyttöönottoon. Useimmiten puolet osallistujista suhtautuu muutokseen neutraalisti ja voi lopuksi kallistua joko muutosmyönteiseksi tai muutosvastaisiksi, ellei heitä vakuuteta muutoksen puolelle. Muutoksen vastustajat voivat vaikuttaa muutokseen neutraalisti suhtautuviin henkilöihin, joten heidän kanssaan kannattaa käydä rehellistä keskustelua muutoksesta. Kaikki eivät koskaan tule kannattamaan muutosta, mutta tärkein vakuutettava Palmerin mukaan ovat kuitenkin muutosten vastustajien sijasta muutokseen neutraalisti suhtautuvat tahot. (Palmer, 2004, ss. 41–42) Muutokseen liittyvissä

konfliktitilanteissa ja erimielisyyksissä kannattaa muistaa seuraava prosessiohje, joka pätee Palmerin mukaan neuvotteluihin yleiselläkin tasolla: ymmärrä toisen osapuolen näkökulmaa, selitä oma näkökulmasi, etsi yhteisymmärrystä sekä jos olet vieläkin eri mieltä, kokeile toista taktiikkaa. (Palmer, 2004, ss. 49–50)

Seuraava vaihe Palmerin muutosmallissa on edistymisen seuranta. Jokaista projektia tulee seurata ja mitata. Korjaavia toimenpiteitä ei voi suorittaa, ellei ole perillä siitä, että asiat ovat menneet pieleen. Edistymistä voidaan seurata erilaisilla prosessiin sopivilla mittareilla. Tavoitteena tässä vaiheessa on muutoksen lopputulosten seuranta, tulosten ja edistyksen havaitsemisen aiheuttama voimauttaminen, varmistuminen siitä, että rajapyykit ja aikataulut ovat seurannassa ja jaettavissa. (Palmer, 2004, ss. 55–56)

Toiseksi viimeinen vaihe muutosprojektissa on työn lopettaminen. Palmerin mukaan työn lopetusvaiheessa muutos ei välttämättä ole vielä aktualisoitunut eivätkä muutoksen kohteena olevat tahot ole vielä kokeneet uutta todellisuutta. Tässä vaiheessa muutoksen vastarinnan lopettamiseksi on tärkeää johdon tuella ilmaista selvästi, että muutos toteutetaan haluttuna ajankohtana ja muutoksen jälkeen se otetaan käyttöön kaikkien toimesta. Tavoitteena työn lopettamisen vaiheessa on ilmaista selkeästi uuden järjestelmän käytön aloituspäivä, organisaation riittävä ymmärrys uudesta järjestelmästä ja tarvittava sitoutuminen sen käyttöön, näkyvä ja tuntuva muutoshankkeen vahvistaminen organisaatiossa sekä uuden hankkeen integroituminen jatkuvien työaktiiviteettien muodossa. Palmer esittää, että toimiva työkalu lopetusvaiheeseen on vahvistamisvaihe, jossa juhlitaan ja hyväksytään muutosprojekti julkisesti tiiminvetäjän ja sponsoreiden toimesta tuoden sitä kautta organisaatiolle näkyväksi, että projektia tuetaan. Julkinen hyväksyntä luo myös projektin tekijöille tuntemuksen siitä, että heidän antamansa panos on tunnustettu. Toinen työn lopettamisen vaiheen työkalu on pilotointi. Palmer suosittelee, että muutos kannattaa ottaa käyttöön selkeästi määritellyssä ja helposti hallitussa alueessa. Tämä pienentää epäonnistumisen kokoa ja antaa mahdollisuuden vielä parantaa lopullista tuotetta pilotoinnin tuoman palautteen perusteella. Kolmas tärkeä työkalu työn lopettamisen vaiheeseen on koulutus. Uudet järjestelmät epäonnistuvat tai saavat vastustusta siitä syystä, että organisaatiota ei koulutettu riittävästi uuteen prosessiin. Koulutusta tulisi tehdä testiympäristössä ja sen tueksi on suositeltavaa myös ottaa mukaan neutraaleja projektin

ulkopuolisia henkilöitä pilotoimaan. (Palmer, 2004, ss. 63–69)

Lopuksi, kun Palmerin muutosmallin edeltävät vaiheet on käyty läpi, muutos tulee sulauttaa organisaation kulttuuriin ja ympäristöön ja sitä tulee tukea organisaatorakenteella, koulutuksella, kehityksellä, palkinnoilla, tunnustuksella, kompensaatiolla ja ylennyskäytänteillä. Vaihteen tavoiteltu lopputulos on tunnistaa olennaiset alueet projektin pitkäaikaisen onnistumisen tukemiselle sekä muuttaa systeemejä ja rakenteita halutun lopputuloksen mukaiseksi. (Palmer, 2004, s. 71)

## **4.2 Muutosinnon löytäminen ja muutoksessa tukeminen**

Muutosinnon täytyy löytyä yksilöstä itsestään esittää puolestaan Ponteva, mutta esimies voi auttaa muutosinnon löytymisessä. Tärkeintä on esimiehen, henkilöstöjohdon ja organisaation johdon taholta viestiä muutoksen olevan myönteinen ja organisaatiota rikastuttava asia. Työntekijää sitoo organisaatioon tuntemus merkityksellisyydestä, turvallisuudesta ja käyttökelpoisuudesta. Ponteva esittää, että oman tekemisensä ja taitonsa merkitykselliseksi tunteva ihminen sitoutuu organisaatioon paremmin ja tuntee voivansa tehdä työtään ilman uhkia. Johtamistavoilla voidaan lisätä työntekijän turvallisuudentunnetta ja edistää työntekijän organisaatioon kiinnittymistä. Kyvykkyys, hyvinvointi, itsenäisyyden kokemus sekä yhteys muihin ovat perustarpeita, joiden toteutumisella saavutetaan työntekijän sisäinen motivaatio sitoutua organisaatioon. Sitoutumiseen vaikuttavat myös johtaminen, työn vaihtelevuus, haastavuus ja itsenäisyys. Muutosinnon löytämiseksi sisäinen motivaatio on olennaista. (Ponteva, 2010, s. 18)

Pontevan mukaan muutoksessa onnistumista tukee esimiehen työntekijöiden inhimillinen kohtelu, johon sisältyy alaisten kohtaaminen ihmisinä, heidän kuuntelemisensa ja epäonnistumisen salliminen kohtuullisissa rajoissa. Pontevan mielestä reiluus muutosta koskeissa asioissa sekä tiedon ja uutisten jakaminen muutoksiin liittyvissä asioissa on olennaista, sillä työntekijöille on tärkeää tehdä merkityksellistä työtä ja olla osa ryhmää. Myönteinen ajattelutapa ja kommunikointi sekä myönteisesti ajattelevien ihmisten kerääminen ympärille ovat keinoja esimiehen taholta vahvistaa myönteisyyttä muutoksessa. Muutoksen onnistumisessa myös innostus ja joustavuus ovat olennaisia. Pontevan mielestä innostusta ja joustavuutta voi jakaa ympärilleen innostamalla



ja innostamalla muita, olemalla latismatta ja kehumalla. Myös arvioinnin säästäminen siihen vaiheeseen, kun kokonaisuus on tiedossa, auttaa innostamisessa. Pontevasta muutoksen onnistumiseen vaikuttaa myös peräänantamattomuus, joka tarkoittaa asioihin tarttumista ja niiden selvittämistä, sinnikkyyttä, niiden tukemista, joilla on vaikeuksia hyväksyä asioita, pienin askelin etenemistä ja asioiden viemistä loppuun sekä kiertotien etsimistä niissä tapauksissa, kun se on tarpeellista. Ponteva esittää, että esimies voi onnistua muutoksessa olemalla mahdollisimman kannustava, etsimällä itseään kannustavia asioita ja kannustamalla eri tavoin, ottamalla työntekijöiden asiat vakavasti ja muistamalla kannustaa viestein, joihin ei sisälly piiloarvostelua. Yllätyksellisyys kuuluu onnistuneeseen muutoksen tukemiseen ja Pontevan mukaan myös avoimuus yllättäville ratkaisuille ja uusien asioiden kokeileminen voivat auttaa muutoksen onnistumisessa. (Ponteva, 2010, ss. 91–107)

Edellä mainittut muutoksen onnistumiskeinot voi kiteyttää Pontevasta inhimillisyyteen, reiluuteen, myönteisyyteen, innostavuuteen, peräänantamattomuuteen, kannustavuuteen ja yllätyksellisyyteen. Nämä ovat hänen mukaansa toimintatapoja, joilla muutoksesta voi selviytyä onnistuneesti sekä toimintatapoja, joita käytännössä yleensä toivotaan toisilta ihmisiltä. Edellämainittujen lisäksi tärkeintä muutoksen onnistumisessa on Pontevan mukaan olla rehellinen itselle ja muille. (Ponteva, 2010, ss. 113–114)

## 5 Atlassian – Jira ja Confluence

Jira-projektinhallintajärjestelmä sekä Confluence-dokumentaationhallintajärjestelmä kuuluvat Atlassianin tuoteperheeseen. Kansainvälisesti toimiva Atlassian perustettiin vuonna 2002 ja sen tuotteita ja tuotteisiin liittyviä lisäosia on tällä hetkellä saatavilla yli 4000. Ensimmäinen Jira-versio tuli markkinoille vuonna 2002 ja Jira Service Management ohjelmisto tuli myyntiin 2020 pohjautuen aikaisempaan Jira Service Desk -ohjelmistoon. (Atlassian, n.d.-b) Alun perin Jira oli tikettienhallintajärjestelmänä kehitetty lähinnä ohjelmistovirheiden seurantaan, mutta vuosien myötä Jira on kehittynyt myös ketterän projektihallinnan työkaluksi Scrumille ja Kanbanille. Jiraa on saatavilla erilaisina versioina esimerkiksi Jira Core, Jira Software ja Jira Service Management. Jiran perusversioiden lisäksi Jiraan on saatavilla monenlaisia maksullisia lisäosia Atlassianin tai kolmansien osapuolien tuottamina. (Kohler, 2013, ss. 5–6)

### 5.1 Jira Core

Ennen käyttöönottoprojektin alkamista toimeksiantajayrityksen käytössä on ollut Jira Core-ohjelma. Jira Core on pääasiassa työnkulunhallintajärjestelmä, johon voi toteuttaa monenlaisia yksilöllisiä prosesseja (Atlassian, n.d.-e). Jira Core -työnkulunhallintajärjestelmää voi käyttää muun muassa projektien toteuttamiseen, laitteistojen seurantaan ja kaikkeen muuhun, missä tarvitaan työnkulkuja. Jira Corea voidaan muokata yksilöllisesti ja se pystytään yhdistämään toisiin tuotteisiin työnseurannan helpottamiseksi. Jira Coressa tapahtumat (Issues) ovat tehtäviä töitä. Näitä ovat esimerkiksi projektin toteuttamiseen liittyvät tehtävät tai laitteistotehtävissä käytetyt laitteet, kuten kannettavat tietokoneet. Erilaisia tapahtumia ryhmitellään Jira Coressa projekteihin, joilla on kullakin omat määritelmänsä. Näillä määrittelyillä varmistetaan, että yksittäisille tapahtumille on kirjattu kaikki tarvittava tieto, kun ne kulkevat projektiin määritellyn työnkulun läpi. Projektin työnkulku määrittelee, miten tapahtumaa voi edistää projektissa. Työnkulkuihin on määritelty statuksia eli tapahtuman tiloja ja transiioita eli liikkeitä eri statusten välillä. (Atlassian, n.d.-d) Kuvassa 2 on esiteltyinä kaaviona Jira-järjestelmän yksinkertainen työnkulku, joka sisältää kolme eri statusta.

Kuva 2 Jira Core -version perustyönkulku



Hallintatyökaluna Jira Core on suunnattu erilaisten projektien hallintaan: tehtävien hallintaan, projektien hallintaan, prosessien hallintaan, henkilöstöhallinnon projekteihin, markkinointiin, operaatioiden hallintaan esimerkiksi tuotantoon sekä talous- ja lakiprojektien hallintaan. (Atlassian, n.d.-g)

## 5.2 Jira Service Management

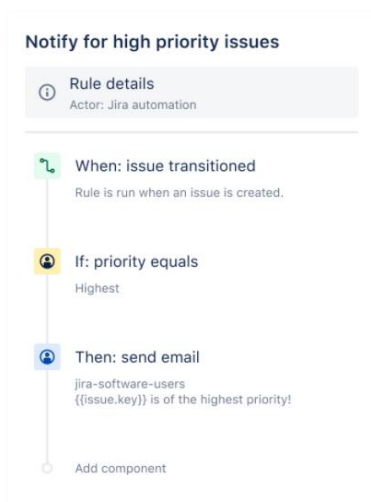
Toisin kuin projektienhallintaan suunniteltu Jira Core, Jira Service Management Project on Atlassian-tuoteperheen erityisesti IT-palvelunhallintaan (IT Service Management) eli ITSM-toimintaan suunnattu osa, joka on rakennettu ketterien ihanteiden pohjalta. Jira Service Management -ohjelman on tarkoitus yhdistää kehitys- ja IT-toiminnot samalle alustalle yhteistyön nopeuttamiseksi. (Atlassian n.d.-c) Jira Service Managementin avulla vastaanotetaan, seurataan, hallitaan ja ratkaistaan tiimin asiakkaiden pyyntöjä eli tikettejä. Tikettejä voidaan lähettää sähköpostin, muokattavan asiakasportaalin ja upotettavan pienoishjelman avulla. Jokainen tiimi voi työskennellä omasta projektistaan käsin ja erilaisille projekteille on Jira Service Managementissa omia projektisapluunoita. Palveluprojekteilla työskennellään asiakkaiden tikkettien parissa ja nämä tiketit ovat tapahtumia, jotka näkyvät jonossa. Samoin kuin Jira Coressa, tapahtumien edistymisen seuranta voidaan hallita työkuluilla. Atlassianin Confluence-tuotetta voidaan käyttää Jira Service Management -ohjelman yhteydessä sisäisen tai asiakaskäyttöön tulevan tietämyskannan (Knowledge Base) luomiseksi. (Atlassian n.d.-h)

### 5.2.1 Automaatiot

Jira Service Management -ohjelmassa on mahdollista rakentaa automaattisia toimintoja (Automations). Automaatioiden idea on auttaa keskittymään merkittävään työhön ja auttaa

poistamaan manuaalisia, toistuvia työvaiheita, kuten tikettien statusten päivityksiä. Automaatioita varten tehdään sääntöjä automaattisten toimintojen toteuttamiseksi. Automaatiosäännöt sisältävät kuvan 3 tapaan kolme osa-aluetta: laukaisijat, jotka käynnistävät säännön, ehdot, jotka varmistavat säännön toteutumisen ja tapahtumat, jotka toteuttavat tehtävät Jirassa. (Atlassian n.d.-i)

Kuva 3 Jiran automaatiosääntöesimerkki Atlassianin sivulta (Atlassian n.d.-s)



Automaatiosäännöille voi myös määritellä statuksia sen mukaan, onko sääntö käytössä, poistettu käytöstä vai vasta luonnosvaiheessa. Yksittäisillä säännöillä on myös seurantalogi (Audit log), josta voi tarkistaa, miten sääntö on toiminut ja mitä sääntöjä ohjelma on toteuttanut. (Atlassian n.d.-i)

### 5.2.2 Sähköposti-ilmoitukset

Jira Service Management lähettää asiakkaille, sisäisille käyttäjille ja ylläpitäjille sähköposti-ilmoituksia järjestelmään avattujen tikettien statuksista ja tikettien puitteissa tapahtuneista aktiviteeteista. Näiden ilmoitusten vastaanottajia, sisältöä ja tyyliä voi muokata. Ilmoitukset perustuvat Jirassa määriteltyihin rooleihin sen mukaan onko käyttäjä asiakas, yrityksen sisäinen käyttäjä eli agentti vai ylläpitäjä. Asiakkaat, jotka avaavat Jiraan tiketin tai seuraavat jotain oman organisaationsa tikettiä, saavat haluttaessa samanlaisia ilmoituksia. Jos asiakas ei halua vastaanottaa tekemästään tai seuraamastaan tiketistä ilmoituksia, joita Jirasta on säädetty

lähteväksi, ilmoitukset voi poistaa käytöstä. Tällöin asiakas saa ilmoituksen ainoastaan, kun tiketti suljetaan. Palveluprojekti voi olla määritelty myös niin, että Jira muodostaa automaattisesti käyttäjätunnuksen asiakkaalle, joka luo ensimmäistä kertaa tiketin Jiraan. Tämän ominaisuuden ylläpitäjä voi tarvittaessa poistaa käytöstä. (Atlassian, n.d.-j)

Agentit ja ylläpitäjät saavat tiketeistä ilmoituksen Jiraan asetetun projektin ilmoitussuunnitelman (Project Notification Scheme) mukaisesti. Agentin omista muutoksista Jiran tiketillä ei tule erikseen ilmoitusta, vaikka agentti suorittaisi asiakkaan asemasta projektin päivityksiä. Jos agentti on nimitetty tiketin käsittelijäksi, hän ei saa ilmoituksia projektin ilmoitussuunnitelman kautta. Samoin, jos Jirassa on sisäinen projekti, jonka kaikilla käyttäjillä on lisenssillinen pääsy Jiraan muussa kuin asiakasroolissa, perinteisesti asiakkaille menevät muistutukset voi kytkeä kokonaan pois päältä. (Atlassian, n.d.-j)

Asiakkaille meneviä ilmoituksia voi muokata palveluprojektin asetuksista. Ilmoituksiin voi määritellä vastaanottajia roolien perusteella. Valittavia rooleja ovat pyynnön jättäjä, muut asiakkaat, lisätyt osallistujat, lisätyt organisaatiot sekä hyväksyntäkäsittelyä vaativien tikettien hyväksyjät. Ilmoituksista voi myös määritellä, jos ei haluta ilmoitusten menevän jollekin tietylle asiakasryhmälle esimerkiksi, jos halutaan rajata, ettei tiketille kirjatusta kommentista lähde ilmoituksia kirjaajalle. (Atlassian, n.d.-j)

Lähteviin ilmoituksiin on mahdollista lisätä yksittäisen tiketin sisältämiä tietoja muokkaamalla ilmoituksia. Tällaisia yksilöllisiä lisättäviä tietoja ovat muun muassa tiketin otsikkotiedot, tiketin käsittelynumero, tiketin raportojan nimi, tiketin status tai lisätyn kommentin sisältö. Jira Service Managementiin voi myös luoda automaattiosäntöjä, jotka lähettävät erillisiä ilmoituksia. Lisäksi asiakasilmoituksiin voi tarvittaessa määritellä käännöksiä tai alueellisia viestejä. Ilmoitusten ulkoasua voi muokata myös yrityksen brändin mukaan muokkaamalla ilmoituspohjien koodia CSS:llä ja Html:llä. (Atlassian, n.d.-j)

### 5.2.3 Kalenterit ja SLA

Jira Service Management -projektissa voi määrittellä tavoiteaikoja tikettien käsittelymisen seuraamiseksi. Jos asiakas on määritellyt palvelutasosopimuksessa tavoiteaikoja, näistä käytetään Jirassa nimitystä SLA, service level agreement. Jira Service Managementissa voi seurata SLA:ta sekä tikettien ensivasteajalle että ratkaisuaikalle. Asetettu SLA määrittelee miten ja milloin aikaa lasketaan sekä milloin tavoite on saavutettu. Jira voi alkaa laskea SLA:n kulumista esimerkiksi siinä vaiheessa, kun tiketti luodaan ja lopettaa laskemisen, kun tiketti on ratkaistu. Tämän lisäksi Jira Service Managementissa voi luoda kalentereita, joihin on määritelty tiimin työajat. Jos kalenteri on kytketty SLA:han ja siihen on määritelty pyhäpäiviä, viikonloppuja tai lounastaukoja, SLA-laskuri pysähtyy näiden kohdalla ja laskenta jatkuu merkattujen ajankohtien kuluttua normaalisti. (Atlassian, n.d.-k)

### 5.2.4 Jonot

Jira Service managementissa tiketit on järjestetty jonoihin. Tikettien saapuessa jonoista muodostettua jononäkymää hyödyntäen pystyy nopeasti määrittelemään, priorisoimaan ja nimittämään tikettejä käsittelijöille. Yksittäisen jonon muodostamassa näkymässä näkee normaalisti vähintään tiketin statuksen, asiakkaan nimen ja tiketin aiheen. Jira Service Management luo projekteille automaattisesti projektin näkymään erilaisia jonoja, mutta näiden lisäksi automaattisesti luotuja jonoja voi muokata ja poistaa sekä projektille voi tehdä kokonaan uusia jonoja. Jonoja voi muodostaa tikettityyppien, prioriteetin tai esimerkiksi SLA:n perusteella. Yhdellä projektilla voi olla maksimissaan 50 jonoa. (Atlassian, n.d.-l)

### 5.2.5 Tietämyskanta

Ylläpitäjä pystyy lisäämään Jira Service Management -projektille Knowledge Basen eli tietämuskannan, jonne projektitiimi voi tallentaa jaettavaksi hyödyllisiä artikkeleita, ohjeita, tietoja ja kysymyksiä tukipyyntöistä sekä esimerkiksi ohjata asiakkaita ratkaisemaan tikettejään tietämuskannassa jaettujen aiempien ratkaisuohjeiden avulla. Tietämuskannan avulla

asiakastikettien ratkaisuja voidaan standardisoida. Tietämyskanta luodaan yhdistämällä Confluence-ohjelmassa luotu sivusto palveluprojektiin. (Atlassian, n.d.-m)

### **5.2.6 Asiakasportaali**

Jira Service Management -projekteissa on kussakin verkossa toimiva asiakasportaali asiakaskäyttöä varten. Jos projektille on lisätty Confluence-tietämyskanta, asiakkaat voivat portaalista käsin etsiä pyyntöihinsä liittyviä artikkeleita tai ohjeita. Projektin ylläpitäjä voi muokata asiakasportaalin nimeä, tervehdysviestiä, visuaalista ilmettä, logoa ja ilmoituksia. Asiakasportaalin linkit, bannerit ja linkkikuvakkeet voidaan määrittellä Jiran ylläpitoasetusten puolelta.

Asiakasportaaliiin voidaan määrittellä, minkälaisia tikettityyppejä asiakkaat voivat portaalissa luoda. Nämä tikettityypit voidaan määrittellä kunkin projektin tikettityypeissä (Request Types). Jira Service Management -projektissa on määriteltä valmiiksi useita tyypillisiä IT-tuen tikettityyppejä, mutta tyyppejä voi muokata tai poistaa. Lisäksi asiakasportaalissa voi myös asettaa näkyville työnkulun siirtymiä, jotta asiakkaat voivat itse päivittää tikettejään. Asiakkaille näkyvät siirtymät toimivat hieman eri tavalla kuin muut työnkulut siirtymät. Työnkulkuihin määritellyt näytöt eivät näy asiakkaille eikä siirtymä vaadi päivittäjältä suorittamaan tehtäviä, joita työnkulkuun on saatettu asettaa. (Atlassian, n.d.-n)

### **5.2.7 Asiakastyytyväisyyskysely**

Asiakastyytyväisyyspalautetta voi kerätä suoraan yksittäiseltä Jira Service Management -projektilta. Ominaisuuden saa aktivoitua lähettämään asiakastyytyväisyyskyselyn heti tiketin ratkaisemisen jälkeen. Vastatut asiakastyytyväisyyskyselytulokset näkyvät suljetuilla tiketeillä ja suljettujen tikettien jonoissa tyytyväisyysarvona (Satisfaction). Lisäksi asiakastyytyväisyyskyselyraportin pohjalta voi järjestelmässä luoda ja muokata raportteja sekä rakentaa trendikuvaajia. (Atlassian n.d.-o)

### 5.2.8 Käyttöoikeudet

Jira Service Management -projektilla pystyy määrittelemään kuka voi avata tikettejä kyseisellä palveluprojektilla ja kenen kanssa asiakkaat voivat jakaa tekemiään tikettejä. Käyttöoikeudet jaetaan palveluprojektilla asiakkaiden osalta kolmeen tasoon:

- asiakkaat, jotka on lisätty projektille voivat avata tikettejä tai
- asiakkaat, joilla on käyttäjätunnukset Jiraan tai
- kuka vain voi lähettää sähköpostilla tiketin palveluprojektille tai avata sen portaalin kautta.

Viimeisintä vaihtoehtoa edeltävistä Jira ei suosittele käytettäväksi, koska sen jälkeen ohjelman omistava yritys ei voi valita kuka voi luoda tunnukset Jiraan ja avata tikettejä. (Atlassian, n.d.-p)

Palveluprojektilla voi myös sallia asiakkaan jakaa tikettejä omassa organisaatiossaan, johon kuuluvat henkilöt voi määrittellä Jirassa projektinhallinnan asetuksista. Rajoituksen voi rajata keneen tahansa palveluprojektilla tai jopa henkilöille, jotka eivät ole vielä asiakasroolissa Jirassa. Nämä tiketille liitetyt osallistujat voivat kommentoida tikettiä, saada tiketin etenemisestä samoja sähköilmoituksia kuin tiketin tekijä sekä kutsua lisää osallistujia seuraamaan tiketin edistymistä. Tikettien jakamisen käyttöoikeuksia voi rajata projektin asetuksista tarpeen mukaan. (Atlassian, n.d.-p)

## 5.3 Confluence

Tämä opinnäytetyö käsittelee Confluencea pilvituotteena, mutta Atlassianin tuoteperheeseen kuuluva Confluence toimii joko pilvessä, datakeskuksessa tai serverillä. Tuotteena sivustomuotoista Confluencea käytetään ensisijaisesti projektiyhteistyöhön, dokumentaatioon, ohjeiden tallentamiseen ja sisällönhallintaan. Confluencessa voi rakentaa sivustoja projektisuunnitelmille, palaverimuistiinpanoille, ohjetietämyskannoille, käytänteille ja moniin muihin tarkoituksiin. Confluencea voi käyttää yhdessä Atlassianin Jira-tuoteperheen kanssa esimerkiksi tietokantana. (Atlassian, n.d.-r)



## 6 Jira Service Management -projektin suunnittelu ja toteutus

Toimeksiantajayrityksessä Bilot Oyj:ssä service desk toimii osana jatkuvia palveluita (Lifecycle Services) ja pääasiallisesti service deskin toimintaan kuuluvat erilaiset ulkoiset ja sisäiset asiakaspalvelutehtävät. Erityisesti tikettien koordinointi- ja seurantatehtävät ovat merkittävä osa service deskin toimintaa, mikä vaikuttaa tarpeeseen tehdä tikettien käsittelystä mahdollisimman tehokasta ja sulavaa sekä asiakkaille että service deskin työntekijöille. Erityisesti kiireellisissä, ennalta ennustamattomissa tilanteissa, kuten asiakkaan ohjelmistoon liittyvissä kriittisissä häiriötilanteissa, tikettienkäsittelyn tehokkuus ja siihen käytetty aika ovat ratkaisevassa tekijässä esimerkiksi sen suhteen, kuinka paljon yrityksen ratkaisutiimille jää palvelutasosopimuksen mukaista ratkaisuaikaa alkaen siitä hetkestä lähtien, kun service desk on käsitellyt asiakkaan häiriöilmoituksen ja välittänyt sen ratkaisutiimille. Erityisesti palvelutasosopimusten puitteissa service deskin tikettienkäsittelytoimintaan käytetty aika on siksi merkityksellistä muille käsittelijöille.

Bilot Oyj:n jatkuvien palveluiden ensisijaisena tikettienhallintajärjestelmänä on vuodesta 2011 ollut Jira. Jiraa on jatkuvissa palveluissa käytetty tikettien hallintaan niissä tapauksissa, joissa asiakkaalla ei ole käytössä omaa tikettijärjestelmää ja myös osassa niissä tapauksissa, joissa asiakkaalla on oma tikettijärjestelmä, mutta tukipalveluilla ei ollut pääsyä asiakkaan järjestelmään. Yrityksessä Jiraa on käytetty palveluiden seurantaan, raportointiin ja hoidettujen tikettien dokumentaatiolähteenä. Jatkuvien palveluiden palvelutavoitteiden seuraamiseksi enemmän projektienhallintaan suunnattuun Jira Coreen on otettu käyttöön Time to SLA – lisäosa ja tyylytellympään sekä useammasta asiakaslähteestä saapuvien sähköpostien toimivampaan käsittelyyn ohjelmaan on lisätty lisäosa Email this issue.

Jira Service Management -ohjelmiston käyttöönottoprojektin alkaessa yritykseen oli yrityskauppojen myötä yhdistymässä toinenkin jatkuvia palveluita toteuttava palveluorganisaatio, jolla ei ollut Jiraa päivittäisessä käytössä. Kahden erilaisesta lähtökohdista tulevan ja erilaisilla toimintatavoilla toimivan organisaation palveluiden hallinnan yhdenmukaistamiseksi ja service desk -työn modernisoimiseksi yrityksessä päätettiin ottaa Jira Coren yhteyteen käyttöön erityisesti service desk -palveluihin suunnattu Jira Service Management -ohjelmisto. Projektinvetäjän

näkemyksellä oli, että uuden lisäosan käyttöönotto toisi yritykselle lisäarvoa, sulauttaisi toimintoja paremmin yhteen ja uusien toimintaprosessien myötä vapauttaisi service deskin työntekijöille aikaa keskittyä paremmin service deskin asiantuntijatehtäviin. Lisäosan sisältämien toimintojen myötä tikkien käsittely, tikkiohjeiden käsittely sekä tikkien seuranta voitaisiin tehdä jatkossa Jirasta käsin eikä hajautetusti, kuten tähän asti.

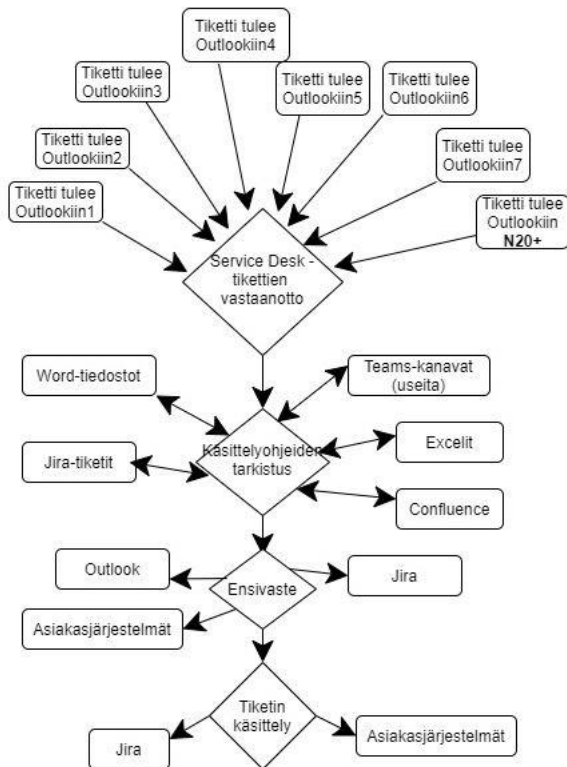
## 6.1 Yrityksen tikkien käsittelyprosessi

Ennen projektin toteutusta service desk otti vastaan asiakkaiden lähettämät palvelupyynnöt, häiriöilmoitukset ja muutospyyntöjä pääasiallisesti sähköpostin kautta. Eri asiakkaille oli omat sähköpostikanavat, joita seurattiin Microsoft O365 Outlookista käsin. Asiakkaiden oli mahdollista ottaa yhteyttä myös puhelimen ja Microsoft Teamsin kautta. Yhteydenoton saapuessa service desk otti viestin vastaan ja

- kävi asiakkaan järjestelmässä tallentamassa tikkia hoitavan henkilön tikkille ja laittoi tikkien siirtämistä kautta liikkeelle,
- hoiti automaattisesti Jiraan ohjautuneen tikkien eteenpäin ratkaistavaksi yrityksen omassa Jirassa ja vastasi asiakkaan yhteydenottoon sähköpostin kautta,
- kuittasi asiakkaan yrityksen Jiraan suoraan avaaman tikkien Jirassa vastaanotetuksi ja hoiti eteenpäin sitä kautta,
- avasi erikseen manuaalisesti tikkien johonkin asiakkaan järjestelmään ja kuittasi sitten sähköpostilla viestin vastaanotetuksi
- tai hoiti tulleen pyynnön suoraan sähköpostilla.

Tämä prosessi vaati useiden sähköpostilaatikoiden sekä Jirasta tulleiden ilmoitusten seuranta ja lisäksi tarvittaessa prosessiin kuuluvien ohjeiden etsintää erilaisista tallennetuista lähteistä (Confluence, Teams, Word, Excel).

Kuva 4 Yrityksen tikkettienkäsittelyprosessi

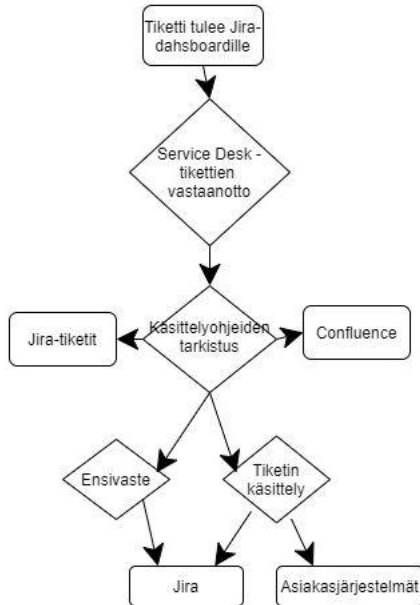


Sekä yrityksen Jirassa että asiakkaan tikkettijärjestelmissä toimiessa tikkettien käsittely vaati myös tikkettien päivitystehtäviä. Tikettiä vastaanottaessa tikkettityyppi määriteltiin ja muokattiin asiakkaan ilmoitusta vastaavaksi, tiketille lisättiin esimerkiksi SLA:n seuraamista tai asiakasraportointia varten erilaisia määritteitä, komponentteja sekä tiketti päivitettiin sovittuun statukseen jatkokäsittelyn aloittamista varten ja joissain tapauksissa ensivasteen SLA:n katkaisemiseksi. Ensivasteen lähettäminen kohdistetusti esimerkiksi asiakkaalle ja tallennettavaksi Jira-järjestelmään vaati tikkettinumeron manuaalista poimimista järjestelmästä ja lisäämistä avatun sähköpostiketjun otsikkoon Outlookissa, jotta jatkossa sekä asiakas että muut luodun sähköpostikeskusteluketjun käsittelijät saivat viestit kohdistettua oikealle tiketille vastaamalla suoraan viestiketjuun. Vanhassa prosessissa yksittäisen tiketin vastaanottoon ja eteenpäin saattamiseen tarvittiin käsittelyprosessissa vähintään kahta ohjelmaa, sähköpostia ja tikketoimintajärjestelmää. Jos service desk -työntekijä tarvitsi käsittelyä varten ohjeistuksia, hän joutui käyttämään vähintään yhtä lisäohjelmaa saadakseen käsiinsä asiaankuuluvat toimintaohjeet.

## 6.2 Projektin kulku

Päätös Jira Service Management -ohjelmiston hankkimisesta oli jo projektin alkaessa tehty, joten projektin toteutus aloitettiin keskustelemalla työn tilaajan näkemyksistä ja tavoitteista Jiraan tulevan lisäosan tuomista hyödyistä service desk -organisaation toiminnan suhteen. Tämän jälkeen kartoitettiin alustavasti palaverissa tulevien loppukäyttäjien ajatuksia tulevasta muutoksesta tikettienkäsittelyprosessissa ja pyydettiin käyttäjiä ilmaisemaan ajatuksia ja ideoita projektin suhteen. Projektin seuraavassa vaiheessa ohjelmisto hankittiin ja hankinnan jälkeen projektia varten konfiguroitiin pari testiprojektia mallintamaan tulevia käyttöönotettavia asiakasprojekteja. Testiprojektin toiminnallisuuksia ja käyttöä perehdytettiin service deskin vapaaehtoisille testaajille ja muille käyttäjille useassa vaiheessa ja testiprojektin avulla prosessia testattiin. Tämän jälkeen testaajia pyydettiin raportoimaan testiprojektin käyttökokemuksista ja kehitysehdotuksista. Saatujen kommenttien pohjalta testiprojekteja muokattiin vastaamaan korjaus- ja kehitysehdotuksissa ilmentyneisiin tarpeisiin. Testaamisen ohessa testikäyttäjiä perehdytettiin Jiran käyttöön useampaan otteeseen. Testaamisen jälkeen Jira Service Managementiin rakennettiin ensimmäiset asiakasprojektit pilottivaihetta varten. Pilottivaiheessa mahdollisesti ilmenevien ongelmien hallitsemiseksi pilottikäyttöön otettiin ainoastaan muutama tikettivolymimäärältään pieni asiakkuus. Pilottikäyttöön otettujen asiakasprojektien käyttäjiltä pyydettiin kommentteja käyttökokemuksista ja kehitysehdotuksia. Pilotointivaiheessa asiakasprojektien asetuksia muokattiin ja yksilöitiin asiakastarpeita vastaaviksi pieniltä osin. Pilotointivaiheen jälkeen Jira Service Managementin käyttöä alettiin laajentaa asteittain koskemaan kaikkia käsittelytavan puolesta mahdollisia jatkuvien palveluiden asiakasprojekteja, jotka olivat jo ennen käyttöönottoprojektia toimineet Jiran piirissä. Lähitulevaisuudessa käyttö on tarkoitus laajentaa myös asiakkuuksiin, joissa käytetään muita tikettijärjestelmiä.

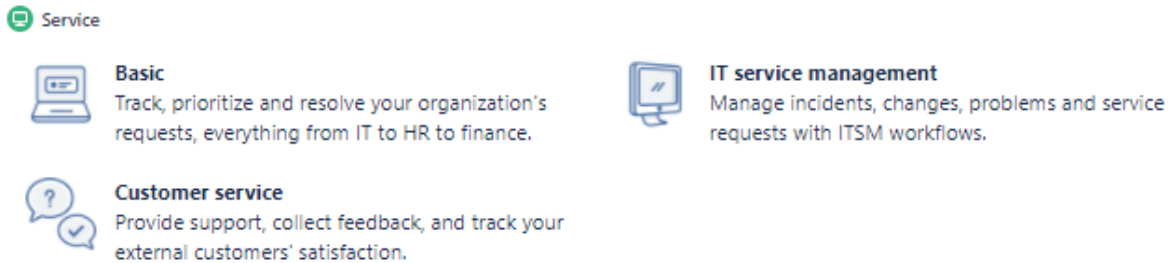
Kuva 5 Yrityksen tikkienkäsittelyprosessi projektin valmistuttua



### 6.2.1 Testiprojektin rakentaminen – projektityyppi

Jira Service Managementiin rakennettiin testausta varten testiprojekti, joka oli muodoltaan mahdollisimman paljon tulevien, käyttöönotettavien asiakasprojektien mukainen. Tätä testiprojektia oli tarkoitus käyttää pohjana kaikissa käyttöönotettavissa asiakasprojekteissa. Aluksi harkittiin rakentaa testiprojekti mahdollisimman yksinkertaisella valmispohjalla, mutta tämä ajatus hyljättiin liian suppeana vaihtoehtona ja lopulta testiprojekti päätettiin rakentaa valmispohjalla, joka olisi mahdollisimman kattava tulevaan asiakaskäyttöön. Jira Service Managementissa pystyy kuvan 6 mukaisesti avaamaan kolmenmuotoisia projekteja: Basic, Customer Service tai IT Service management.

Kuva 6 Jira Service Management -projektityypit



Testiprojektia pystytettäessä avattiin ensin kokeeksi sekä Basic-muotoinen että Customer service -muotoinen projektipohja, mutta molemmat olivat liian suppeita vaihtoehtoja yrityksen jatkuvien palveluiden käyttöön. Lopullinen pohja rakennettiin IT Service Management -pohjalla, jossa oli riittävän laaja tikettityyppivalikoima sekä riittävän laaja työnkulkukaavio, joiden pohjalta projektia pääsi muokkaamaan eteenpäin toimeksiantajan tarpeita vastaaviksi.

### 6.2.2 Tikettityypit

Jira Service Managementissa määritellään erikseen palvelupyynnötyypit (toimeksiantajayrityksessä tikettityypit), joita asiakas voi avata omaan projektiinsa kuuluvassa asiakasportaalissa.

Testiprojektin pohjaksi valitussa IT Service management -pohjassa valmiina olevia tikettityyppejä oli paljon, mutta ne koskivat kaikki IT-laitteiden hallintaa ja erilaisia käyttäjätuen ongelmatilanteita, joten valmiit tikettityypit poistettiin käytöstä ja luotiin uudet ITILin ja asiakassopimusten mukaiset tikettityypit, joita vastaavia ennen käyttöönottoprojektia oli ollut jo käytössä muilla Jira-projekteilla. Tikettityypeistä testiprojektille avattiin kuvan 7 tapaan incident, service request, change request, problem, task ja service order.

Kuva 7 Testiprojektin pyyntötyypit (Request types)

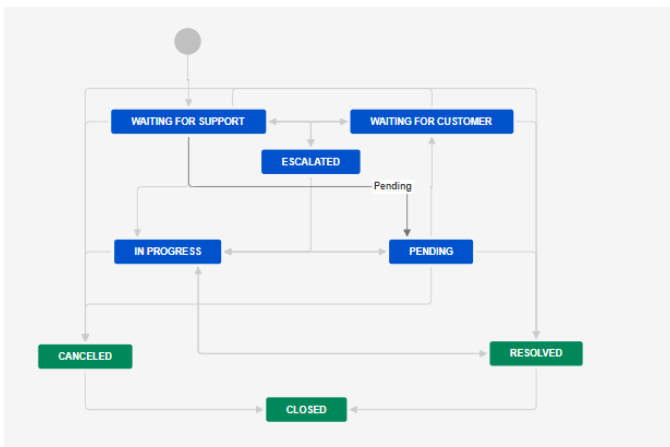
Icon	Request name	Issue type	Description (Optional)
		<input checked="" type="checkbox"/> Task	
	Request for incident investigation	<input checked="" type="checkbox"/> Incident	Something is not working as it should/ Having trouble with a system? Report system incident.
	Request for service	<input checked="" type="checkbox"/> Service Request	Need information, advice, access to a service?
	Change Request	<input checked="" type="checkbox"/> Change	Need adjustment/change to a system.
	Service Order	<input checked="" type="checkbox"/> Service Order	Service Ordered
	Problem	<input checked="" type="checkbox"/> Problem	Recurring incident. (Multiple incidents created before this problem.)

Service order -tikettityyppiä lukuun ottamatta samat tikettityypit tulevat IT Service management -projektipohjassa automaattisesti myös Issue Type Schemeen, johon Jirassa määritellään projektilla käytössä olevat tikettityypit. Tämän vuoksi skeemaa ei tarvinnut muokata runsaasti ja siihen lisättiin ainoastaan uutena tikettityyppinä service order -tikettityyppi.

### 6.2.3 Työnkulku

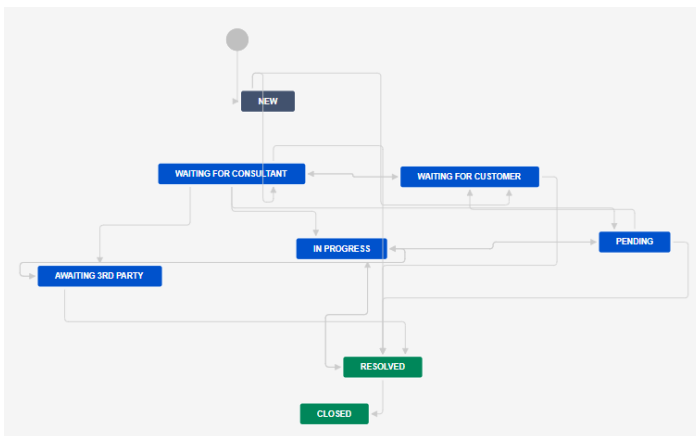
Jira Service Management luo työnkulkuluja (Workflow) jokaiselle avatulle projektille. Automaattisesti luoduista työnkuluista (kuva 8) Service Request Fulfillment workflow for Jira Service management vaikutti testiprojektille hyödyllisimmältä vaihtoehdolta, joten se otettiin pohjaksi testiprojektin rakentamiseen.

Kuva 8 Service Request Fulfillment -työnkulku Jirassa



Service deskin käsittelijöiden työprosessin selkeyttämiseksi testiprojektin työnkulkuun päivitettiin status New. Tätä statusta hyödyntämällä useampi käsittelijä saattaisi seurata projektien tilannetta ja tietää, että uusi tiketti olisi otettu käsittelyyn, kun se olisi päivitetty statukselta "New" seuraavalle statukselle. Alkuperäisessä työnkulussa ensimmäinen status oli Waiting for support. Tämä status nimettiin service deskin näkökulmasta selkeämmäksi nimeämällä se nimityksellä Waiting for consultant. Tämän lisäksi työnkulkuun lisättiin Awaiting 3rd party -status ja poistettiin turhana Canceled-status (kuva 9).

Kuva 9 Testiprojektin lopullinen työnkulku Jirassa







#### 6.2.4 Näytöt, kentät ja prioriteetit

Testiprojektiin automaattisesti Jiran luomat näytöt (Screen), jotka määrittelevät kuinka tiketin kentät näkyvät ja kentät (Field), jotka sisältävät tikettityypin kannalta olennaista tietoa olivat testiprojektin rakentamisen vaiheessa riittäviä, joten niitä ei projektin aikana muokattu. Sen sijaan Jira Service Managementin asennuksen myötä Jiraan ilmestyi uusia prioriteettiluokkia. Osan asennus lisäsi automaattisesti kaikille Jira Coressa käytössä olleille projekteille käyttöön uudet prioriteettityypit, joten uuden lisäosan käyttöönoton jälkeen uudet prioriteettityypit Blocker, Medium ja Minor poistettiin ensin vanhoilta projekteilta ja niille palautettiin käyttöön alkuperäiset perusprioriteettityypit: Critical, High, Moderate ja Low. Uudet Jira Service Managementin prioriteettityypit valittiin käyttöön testiprojektille ja prioriteettityypeistä luotiin tulevien projektien luomista helpottava valmis prioriteettiskeema, jonka voisi kopioida käyttöön myös muille service deskin käyttöön tuleville projekteille. Prioriteettivalikoima valittiin aikaisempia projekteja



mukailevasti neliportaiseksi ja prioriteettityypeiksi otettiin käyttöön Blocker vastaamaan aikaisempien projektien kriittistä prioriteettia, Medium vastaamaan Moderatea sekä High ja Low, jotka olivat jo ennestään käytössä (Kuva 10).

Kuva 10 Testiprojektin prioriteettityypit Jirassa

Icon and name	Description
 <b>Blocker</b>	The problem will block progress.
 <b>High</b>	Major loss of function.
 <b>Medium</b> (Default)	Has the potential to affect progress.
 <b>Low</b>	Cosmetic problem like misspelt words or misaligned text.

Pilottivaiheen jälkeen Service Management -projektien Blocker- ja Medium-prioriteettityypit vaihdettiin käyttäjien toiveesta selkeyden vuoksi Critical- ja Moderate-prioriteettityypeiksi muiden aikaisempien Jira-projektien mallin mukaan.

### 6.2.5 Asiakasoikeudet

Projektin haastavimmaksi osaksi muodostuivat asiakasoikeudet. Jira Service Management -projektilla voidaan määritellä kuka voi avata uusia pyyntöjä projektiin sekä kenelle pyynnön voi jakaa. Alun perin yritysten sisäiseen IT-tukeen suunniteltu Jira Service Management -projekti voidaan määritellä niin, että kuka tahansa voi avata projektille tikettejä asiakasportaalin kautta ja tikettejä voi jakaa kenelle tahansa. Tietoturvasyistä service deskin projekteille ei haluttu mahdollisuutta, että kuka tahansa voi avata tikettejä ja nähdä asiakasportaalin. Tästä syystä testiprojektille määriteltiin, että asiakkaat, joilla on tunnukset Jiraan ja oikeudet kyseessä olevalle projektille, voivat avata tikettejä ja tikettejä voi jakaa vain samaan organisaatioon liitetuille asiakkaille. Näiden määrittelyjen takia Jira Service Management ei kuitenkaan suostunut avaamaan asiakassähköpostin kautta lähetettyjä pyyntöjä, jos viestin lähettäjällä ei ollut olemassa olevia Jira-tunnuksia. Rajoituksen vuoksi ongelma päätettiin kiertää hyödyntämällä Jira Coressa käytössä olevaa Email This Issue -lisäosaa. Kyseisen lisäosan ja Jira Service Managementin integraatiolla projektin asetukset pystyttiin määrittelemään niin, että asiakassähköpostista tulevat viestit asiakkailta, joilla ei ole Jira-tunnuksia, ohjautuvat edelleen Email This Issuen kautta Jiraan ja niistä avautuu automaattisesti tiketti Jiraan, kuten Jira Coren projekteissa.

### 6.2.6 Asiakasilmoitukset

Jira Service Managementiin ei tarvita periaatteessa aiemmin Email This Issue -lisäosan kautta käytössä olleita viestipohjia, sillä lisäosassa voi määritellä jokaiselle projektille yksilöitävän asiakasviestinpohjan. Lisäosassa projektilla voi myös määritellä valmiista ilmoitussäännöistä, mitkä säännöt ovat voimassa tai pois käytöstä ja lisäksi lisäsääntöjä voi määritellä kunkin projektin automaattiosäännöistä. Yksittäisellä projektilla voi myös määritellä useita kieliä käyttöön, mutta toimeksiantajaryityksen käyttökieli on englanti, joten muita kieliä ei tässä vaiheessa lisätty asiakasviestipohjiin. Testiprojektiin ja avattavia asiakasprojekteja varten muokattiin Html-koodia ja CSS-koodia hyödyntämällä yhtenäinen, yrityksen visuaalista ilmettä mukaileva viestipohja projektin viestipohjaksi. Testausvaiheessa automaattisista ilmoitussäännöistä kytkettiin pois käytöstä ilmoitus ”Request Created”, koska haluttiin päättää tapauskohtaisesti ensimmäisten vastausviestin lähetystyyli ja -tapa, sekä ilmoitus ”Approval Required” toistaiseksi turhina ilmoitustyyppinä.

Pilotointi- ja käyttöönottovaiheissa Jira Service Managementin asiakasilmoituspohjat otettiin pois käytöstä useimmissa projekteissa, sillä lisenssisyistä ratkaisukonsulttien ja asiakkaiden välinen kommunikaatio ja siitä Jiran luomat ilmoitukset jouduttiin ohjaamaan Jira Service Managementista Jira Coreen ohjattujen tikettien kautta.

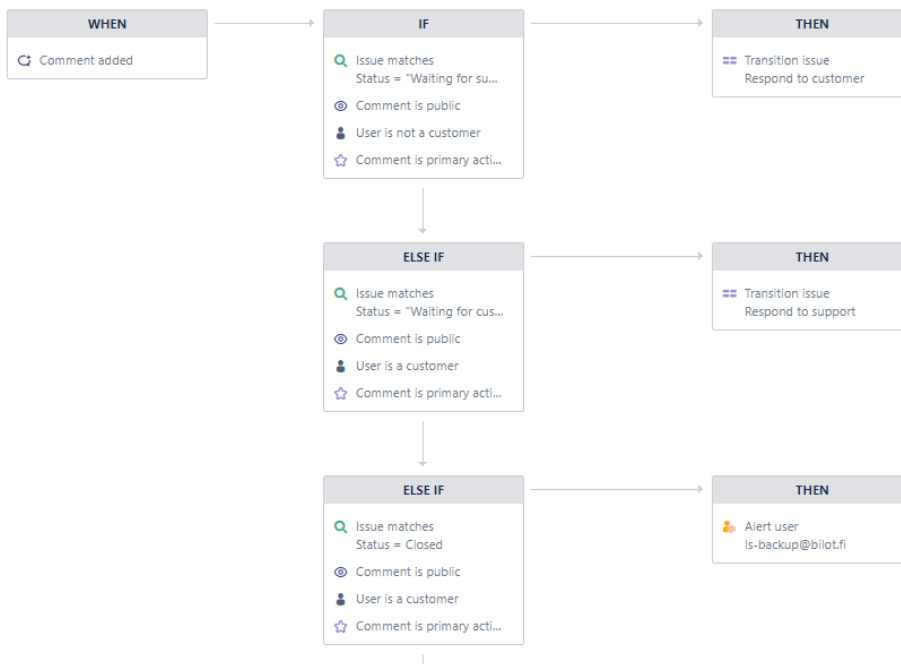
### 6.2.7 Automaatiot

Projekteille luotavat automaattiosäännöt ovat Jira Service Managementin huomattava parannus muihin Jira -tyyppihin verrattuna. Automaatioiden tarkoitus on vähentää tikettien käsittelijöiden käsin tehtäviä statuspäivityksiä ja esimerkiksi automaattisesti uudelleen avata kommentoituja suljettuja tikettejä, jotka saattavat normaalisti jäädä käsittelijöiltä huomaamatta. Automaatioilla saa määriteltyä myös varoituksia SLA-aikojen loppumisesta sekä esimerkiksi päivitettyä linkitettyjä tikettejä statusten vaihtuessa.

Testiprojektille luotiin automaatio varoittamaan SLA-ajan loppumisesta sekä automaattinen sulkeminen ratkaistuille tiketeille kahden viikon kuluttua tiketin ratkaisusta. Tämä vaihe on

aiemmin jouduttu tekemään käsin, joten automaatio oli suora parannus aikaisempaan prosessiin. Lisenssisyistä Jira Service Managementin service deskin hallinnassa olevat projektit joudutaan ainakin jonkin aikaa käyttöönoton jälkeen linkittämään ratkaisukonsulttien projekteihin vanhan Jira Coren puolella, joten projektille rakennettiin automaatio, joka huomauttaa tiketillä, jos linkitettyyn tickettiin on tullut päivityksiä. Tämän lisäksi luotiin kuvan 9 mukaisia automaattisia statussääntöjä, jotka muokkaavat tickettien statusta sen mukaan, miten linkitetty ticketti päivitetään, jotta Jira Service Managementin puolella olevaa tickettiä ei tarvitse erikseen päivittää.

Kuva 11 Osa testiprojektin "Comment added" -automaatiosääntökaaviosta



## 6.2.8 Tietämyskanta

Ohjeistusten saamiseksi Jiran yhteyteen helposti löydettäväksi Jira Service Managementiin kytkettiin valmiiksi yrityksen käytössä oleva Atlassian Confluencen dokumentaationhallintajärjestelmä. Jira Service Managementissa tietämyskanta (Knowledge Base) voidaan hyödyntää esimerkiksi asiakkaiden ohjeistusten tallentamiseen ja asiakkaiden itsepalvelun tukemiseen, mutta käyttöönottoprojektin päätarkoitus oli ottaa tietämyskanta ensisijaisesti käyttöön service desk -henkilökunnan tickettienkäsittelyohjeiden ja muiden

asiakkuuksiin liittyvien sisäisten ohjeiden tallentamiseen prosessin helpottamiseksi. Jiraan luotiin tietämuskannan perustamista varten erillinen käyttäjä, jolle myönnettiin Confluencessa riittävät oikeudet tietämuskantaa varten perustettujen Confluence-sivujen avaamiseen ja sisällön näkemiseen. Käyttäjä kytkettiin Jira-projekteille ja tietämuskanta aktivoitiin käyttöön. Asiakasprojektien suhteen service deskin käyttäjille määriteltiin riittävät oikeudet Confluencen dokumentaation hallintaan ohjeiden tallentamisen osalta.

### **6.2.9 Jonot ja Dashboard**

Jira Coren puolella yrityksessä oli ollut koko ajan erinäisissä yhteyksissä käytössä dashboard-toiminnallisuus, jolla erilaisilla Jiran tarjoamilla muokattavilla valmisohjelmapaloilla (Gadgets) oli mahdollista luoda yhdeltä sivulta seurattavia monitorointinäkymiä esimerkiksi saapuvien tikettien tai SLA:n piiriin kuuluvien tikettien seuraamiseksi. Projektin tilaajan näkemys oli siirtää saapuvien viestien seuraaminen yhteen keskitettyyn dashboard-näkymään, jolloin service deskin työntekijät vapautuisivat eri sähköpostilaatikoiden seuraamisesta saapuvien tikettien osalta. Jira Service Managementin tiketit saatiin mukaan käyttöönottoa varten rakennettuun dashboard-näkymään ja service deskin työntekijöille aktivoitiin testausvaiheen alkaessa näkymä seurattavaksi ja käyttötarpeiden mukaan muokattavaksi sekä rohkaistiin heitä rakentamaan ja kokeilemaan erilaisia dashboard-näkymiä tarpeen mukaan.

Jira Service Managementin puolella yksittäisillä projekteilla on dashboardin sijasta jononäkymä, johon pystytään erilaisia määritteitä lisäämällä tekemään sopivia jonoja esimerkiksi sisään tulevien, avoimien tai kriittisten tikettien seuraamiseen. Jononäkymiä päivitettiin käyttöönottoprojektin kannalta sopiviksi, mutta projektin tarkoitus oli vähentää service deskin työntekijöiden tarvetta seurata useita lähteitä saapuvien tikettien valvomiseksi ja sen vuoksi service deskin käyttöön muokattiin ensisijaisesti dashboard-näkymä usean jononäkymän sijasta.

### **6.2.10 Asiakastyytyväisyyskysely**

Yrityksellä ei ole aiemmin ollut käytössä yksittäisiin tiketteihin kohdistuvia asiakastyytyväisyysmittareita, joten Jira Service Managementin asiakastyytyväisyystoiminto oli

tarkoitus aktivoida heti käyttöön, mutta ominaisuus jouduttiin väliaikaisesti poistamaan käytöstä lisensseistä johtuvien tikettinäkömäärärajotusten vuoksi. Asiakastyytyväisyyskysely on kuitenkin tarkoitus aktivoida käyttöön projekteille sitä mukaa, kun asiakaskommunikaatio siirtyy pääosin Jira Service Managementin puolelle.

### **6.2.11 Asiakasportaali**

Jira Service Managementin asiakasportaali muokattiin ja brändättiin projektin aikana käyttövalmiiksi ja portaali on tällä hetkellä pilottikäytössä parissa asiakasprojektissa. Käyttöönottoprojektin päätavoite oli sisäisen käsittelyn toiminnoissa, joten muiden projektikohtaisten asiakasportaalien käytön aktivoiminen eri asiakkuuksissa jäi odottamaan tulevaisuuden tarpeita ja jatkokehitysvaihetta.

## **6.3 Pilotointivaihe**

Pilotointivaiheeseen valittiin muutama service deskin asiakkuus, joissa tikettivolyymit eivät olleet liian suuria uuden prosessin harjoittelemisen näkökulmasta. Muokattua ja testattua testiprojektia käytettiin pilottiprojektien mallina ja pilottiasiakkuuksille rakennettiin kullekin oma Jira Service Management -projekti. Projektit kytkettiin käytössä oleviin asiakassähköpostilaatikoihin, sähköpostilaatikot päivitettiin IT-tuen avulla oikeanmuotoisiksi Jira Service Managementin ja Email This Issue -applikaation kautta tapahtuvaan sähköpostien sisään lukuun ja sähköpostilaatikoiden vanha ohjaus Jira Coren puolelle kytkettiin pois käytöstä. Asiakasprojektien vastuuhenkilöt alkoivat testata avattuja projekteja käytössä sekä heitä pyydettiin myös antamaan palautetta ja kehitysideoita käytössä havaittujen ominaisuuksien pohjalta.

Pilotointivaiheessa tehtiin pieniä muokkauksia aktivoituihin asiakasprojekteihin ja kytkettiin asiakaskohtaiset SLA-laskurit päälle. Pilottivaiheen loppukäyttäjää pyrittiin antamaan mahdollisimman aktiivisesti palautetta käyttökokemuksista, puutteista ja kehitysehdotuksista. Valitettavasti pilotointiin valitut projektit olivat tikettivolyymeiltään hiljaisia ja loppukäyttäjien aikataulut kiireisiä, joten palautteen ja kokemusten kerääminen oli hidasta. Pilotointivaihe päättyi käytännössä siihen, että uusi Jiran lisäosa todettiin toimivaksi, uusi prosessi haltuun otetuksi eikä

vakavia puutteita havaittu. Lisenssi ongelmista johtuen erinomaista parannusta aikaisempaan prosessiin ei pilotin aikana vielä huomattu, mutta positiivista palautetta tuli sähköpostien käsittelyn ja lajittelun vähentymisen vuoksi. Alkuharjoittelun jälkeen loppukäyttäjät kokivat, että uutta järjestelmää oli helppo käyttää.

#### **6.4 Käyttöönottovaihe**

Pilottivaiheesta siirryttiin liukuvasti käyttöönottovaiheeseen, koska todettiin pilottivaiheen tikettivolyymien olevan sen verran vähäiset, että käyttöönottovaiheeseen oli paras siirtyä vähitellen. Käyttöönottovaiheessa asiakasprojekteja aktivoitiin käyttöön maltilliseen tahtiin loppukäyttäjien muihin aikatauluihin sovittaen ja aikataulu myöhästyi pääasiallisesti käyttöönottoresurssien puutteesta johtuen jonkin verran. Käyttöönottovaiheen aikatauluihin vaikuttivat myös kuhunkin asiakkuuteen liittyvät muut sisäiset vastuuhenkilöt, joille uutta prosessia esiteltiin. Osassa asiakkuuksissa päädyttiin viivästyttämään käyttöönottoa käyttöönottoprojektin ulkopuolisista syistä, resurssipulan vuoksi sekä sen vuoksi, ettei uuden prosessinkäyttöönotto toisi välttämättä merkittävää parannusta käytössä olevaan prosessiin. Näistä hidastuksista huolimatta asiakasprojekteja aktivoitiin vähän kerrallaan ja projekti eteni. Käytännössä miltei kaikki aktivoidut projektit saatiin toteutumaan määritellyn prosessin mukaisesti.

Käyttöönottovaiheen aikana eniten ongelmia ilmeni viestien sisäänotossa asiakkuudessa, jossa Jira tulkitse asiakkaan tikettijärjestelmästä lähetettyjä viestejä roskapostiksi tai automaattisiksi vastauksiksi. Tähän löytyi toistaiseksi ratkaisuksi vain kaikkien automaattisten viestien salliminen viestien sisäänotossa, joten kyseisen asiakkuuden suhteen päädyttiin pitäytymään alkuperäisessä tikettienkäsittelyprosessissa, kunnes järjestelmäpäivitysten tai jonkin muun ratkaisun myötä toisesta järjestelmästä lähetetyt viestit saataisiin ohjattua myös Jira Service Managementin puolelle.

Testaus- ja pilotointivaiheen aikana käytetyn työnkulun ohelle otettiin käyttöönottovaiheessa vielä yksinkertaistettu työnkulku statuksilla Open, In Progress ja Resolved niille asiakkuuksille, joissa tikettien käsittely tapahtui pääasiallisesti asiakkaan omassa tikettijärjestelmässä. Näissä

projekteissa Jira Service Managementin ensisijainen tarkoitus oli saapuvien tikkettien seurannan siirtäminen pois sähköposteista keskitetyille alustalle ja muut toiminnot tapahtuivat asiakkaan järjestelmässä, joten yksinkertaistettu työnkulku vähentäisi Jira Service Managementin puolella tapahtuvaa käsittelytarvetta.

Käyttöönottovaiheen lopussa kaikille alun perin suunnitelluille projekteille oli aktivoitu Jira Service Management -projekti lukuun ottamatta niitä asiakkuusprojekteja, joissa service deskin ulkopuoliset syyt olivat siirtäneet projektien käyttöönoton myöhäisempään aikaan. Siirrettyjen projektien osalta oli kuitenkin määritelty jo tavoiteaikataulu, jonka puitteissa projektit oli tarkoitus ottaa käyttöön ja käyttöönotto tulee tapahtumaan asiakkuustarpeiden ja loppukäyttäjien aikataulujen mukaisesti.

## **6.5 Jatko ja seuranta**

Lähitulevaisuudessa Jira Service Managementin käyttöä on tarkoitus laajentaa sovittujen asiakasprojektien käyttöön. Tämän lisäksi uusi lisäosa on tarkoitus ottaa käyttöön toimeksiantajan tarpeiden mukaan myös erilaisten kehitysprojektien käyttöön, jotta kehitysprojekteissa mukana olevat asiakkaat ja konsultit voivat hyödyntää asiakasportaalia ja muita Jira Service Managementin ominaisuuksia. Tällä hetkellä käytössä olevia Jira Service Managementiin tehtyjä asiakasprojekteja seurataan ja kehitystyötä tehdään tarpeen mukaan puutteiden ja kehitysideoiden ilmetessä ja asiakas- ja käyttövaatimusten muuttuessa. Käyttöön otettujen asiakasprojektien asiakasportaalit on tarkoitus aktivoida käyttöön ensisijaisesti asiakastarpeiden mukaan.

Tärkeimpänä jatkotoimenpiteenä on lisenssien laajennus isommalle käyttäjäkunnalle, jotta jatkuvien palveluiden tikketikommunikaatio pystytään siirtämään kokonaisuudessaan Jira Service Managementin pariin ja tarve luoda tuplatiketöintiä Jira Service Managementin ja laajemmin lisensoidun Jira Coren välillä loppuu. Tämän jälkeen loputkin Jira Service Managementin tällä hetkellä käytöstä poistetut toiminnallisuudet saadaan aktivoitua käyttöön projekteille ja service desk ja muut toimivat pystyvät siirtymään täysipainoisesti tikkettienkäsittelyyn Jira Service Managementin kautta.

## 7 Johtopäätökset ja pohdinta

Jira Service Management -ohjelmisto tilattiin yrityksen käyttöön vastaamaan havaittuihin epäkohtiin service deskin tikettienkäsittelyprosessissa tavoitteena kehittää ja parantaa prosessia. Projektin tilaajan aikaisemman kokemuksen perusteella Jira Service Managementista löytyisi ratkaisuja service deskin tikettienkäsittelyprosessissa havaittuihin ongelmakohtiin ja samalla pystyttäisiin yhtenäistämään kahden jatkuvien palveluiden organisaation tikettienkäsittelyprosessit yhteen uuteen prosessiin. Ajatuksena tavoite yhdenmukaistaa prosessit ja siirtää sekä tikettien käsittely että ohjeiden keskitys samalle alustalle oli työn toteuttajan näkökulmasta erittäin hyvä ja kannatettava. Projektia tehtäessä sai hyvän käytännön kokemuksen käyttöönottoprojektista ja etsintöjen jälkeen työhön löytyi myös työn tilaajan toivomaa teoriaa muutosprosessista, vaikka alkuun sopivaa teoriamateriaalia tuntui olevan hankala löytää. Samoin IT-käyttäjätukeen keskittyvistä service desk -organisaatioista löytyi teoriamateriaalia, mutta enemmän B2B-toimintaan keskittyvään service desk -malliin oli hankalampi löytää soveltuvaa teoriapohjaa. Teoriaosuus saatiin kuitenkin kasattua ja siihen liitettiin alun perin mukaan toivotut osiot Lean-teoriasta ja muutosvistarinnasta.

Käyttöönottoprosessia eli muutosprosessia lähdettiin toteuttamaan service deskin tiiminvetäjän aloitteesta ja loppukäyttäjiä pyrittiin saamaan sitoutumaan jo toteutettavaksi päätettyyn projektiin ennakkokeskusteluiden avulla. Palmerin muutosprosessiteorian mukaan muutosvistarinta ja humanistiset syyt ovat yleensä projektin epäonnistumistekijöitä. Palmerin teorian pohjalta organisaation kyky hyväksyä tuleva muutos sekä kapasiteetti tulevaan muutokseen olisi työn tekijän näkökulmasta pitänyt kartoittaa tarkemmin, sillä näiden tekijöiden epäonnistuminen heijastui projektissa suoraan testausaktivointiin, työntekijöiden sitoutumiseen, kehitysehdotusten ja palautteiden saamiseen sekä uuden järjestelmän käyttöönottohalukkuuteen. Palmerin muutosprosessiin nojaten yhteisen tarpeen luominen ja vision muotoileminen epäonnistuivat projektissa johtuen loppukäyttäjien voimakkaasta jo projektia edeltäneestä muutosvistarinnasta, joka näkyi projektiin sitoutumattomuutena ja osallistumishalukkuuden puutteena. Loppukäyttäjiä pyrittiin kuitenkin sitouttamaan projektiin mukaan rohkaisemalla testaamaan, kysymään ja kommentoimaan ja tekemällä mahdollisimman selväksi, että omalla vaikuttamisella voi vaikuttaa tulevien uusien prosessien lopputulokseen. Projektin jokaisessa



vaiheessa loppukäyttäjillä oli mahdollisuus ilmaista mielipiteensä ja esittää kysymyksiä. Lisäksi koulutusta järjestettiin ja luvattiin järjestää joka projektivaiheessa lisää tarpeiden ilmetessä ja kaikkia kannustettiin ilmaisemaan lisäkoulutuksen tarve ja uuteen prosessiin liittyvät ongelmat ja kehitysehdotukset heti niiden ilmetessä. Muutosmallin tapaan projektiin neutraalisti suhtautuvia koetettiin kannustaa ja heiltä koetettiin saada palautetta projektin muokkaamiseksi niin, että lopputulos olisi aidosti tiketinkäsittelyprosesseja helpottava. Näistä toimista huolimatta humanistiset tekijät ja muutosvastarinnasta johtuva passiivisuus vaikeuttivat projektin onnistumista ja vaikuttivat lopputuloksen aikatauluihin.

Koko käyttöönottoprojekti pyrittiin toteuttamaan Lean-arvomaailman mukaan. Projektin idea oli Leanin periaatteiden mukaan pyrkiä vähentämään service deskin työssä havaittua hukkaa ja parantaa prosessien virtausta muun muassa vähentämällä hukaksi koettuja työvaiheita ja turhia järjestelmien välisiä siirtymisiä. Hukan vähentämisen koettiin vaikuttavan suoraan asiakkaan arvoa lisäävästi, sillä tikettienkäsittelyprosessin sulavoitumisen lopputulemana service deskin asiantuntijoiden työaika vapautuisi enemmän varsinaiseen asiantuntijatyöhön ja prosessien kehittämiseen. Projektin tavoitteena oli Leanin perusteiden mukaan myös sitouttaa loppukäyttäjiä mukaan kehitysprojektiin ja sitä kautta hyödyntää sekä heidän osaamistaan ja saada käyttäjät kokeilemaan uutta prosessia ennakkoluulottomammin ja osallistumaan organisaation kehitykseen aiempaa itseohjautuvammin. Varsinaista itseohjautuvuuden ja sitoutumisen kasvua ei projektin aikana näkynyt, mutta alun perin neutraalisti muutokseen suhtautuvat antoivat projektin loppuvaiheissakin neutraalia ja positiivista palautetta uuden prosessin toimivuudesta ja kokivat, että siinä on hyviä ominaisuuksia loppukäyttäjien näkökulmasta verrattuna aikaisempaan prosessiin.

Opinnäytetyön teoriassa mainittua DMAIC-ongelmanratkaisuprosessia käytettiin runkona koko projektin ajan. DMAIC-ongelmanratkaisuprosessin vaiheet: määrittely, mittaus, analysointi, parannus ja ohjaus pätevät käytännön tasolla kaikkiin projektin vaiheisiin, joissa ratkaistiin esimerkiksi loppukäyttäjien ilmoittamia testihavaintoihin liittyviä ongelmia. Käytännön tekemisessä DMAIC kulki projektissa ongelmanratkaisuprosessina jatkuvasti mukana siinä, kuinka tehtyjä havaintoja tarkasteltiin, parannettiin ja loppukäyttäjiä opastettiin tehtyjen muutosten osalta, vaikka ongelmanratkaisuprosessia ei erityisemmin määritelty julkisesti DMAIC-

ongelmanratkaisuprosessiksi. Ainoa ero DMAIC-ongelmanratkaisuprosessiin oli varsinaisen mittausvaiheen puuttuminen, sillä projektissa ei ollut käytössä mittausvaiheeseen erityisiä mittaristoja eikä esimerkiksi tikettien käsittelyyn kuluva aikaa mitattu ennen eikä jälkeen projektin millään tavalla, vaan analysoinnin pohja perustui tiiminvetäjän ja loppukäyttäjien subjektiivisiin näkemyksiin. Käytännön toteutuksessa ongelmanratkaisuprosessissa mittaus- ja analysointivaihe siis sulautuivat yhteen.

Varsinainen käyttöönottoprojektin tekninen puoli sujui ongelmitta ja hyvin olemassa olevat rajoitukset huomioiden. Käyttöönottoprojektissa pyrittiin ottamaan uusi ohjelmisto käyttöön mahdollisimman laajalti ja hyödyntämään kaikkia siinä olevia ominaisuuksia. Kaikki ohjelman toiminnot saatiin työn tekijän toimesta kytkettyä käyttöön testauksessa ja samat toiminnallisuudet aktivoitiin myös käytettävissä asiakasprojekteissakin, mutta valitettavasti käytössä olevien lisenssien rajallisuus johti projektin osalta siihen, että Jiran uusi lisäosa pystyttiin ottamaan käyttöön sovitusti vain service deskillä. Koska deskin ulkopuoliset toimijat esimerkiksi ratkaisukonsultit eivät pystyneet käyttämään Jira Service Managementia lisenssien puutteessa, lisäosan hyödyt, kuten automaatiot, automaattiset asiakastytyväisyyskyselyt, asiakasportaali ja tietämuskantaintegraatio olivat vain deskin käytössä eikä niitä pystytty projektin aikana ottamaan laajempaan käyttöön. Automaatioiden ja muiden erityisominaisuuksien hyödyntäminen rajautui sen vuoksi ainoastaan service deskin ratkaisemiin tiketteihin ja tiimin ulkopuolisiin tiketteihin täytyi käyttää Jira Coren puolella olevia vanhoja asiakasprojekteja ja linkittää näihin service deskin tikettejä. Käytännössä siis tikettien käsittely pystyttiin projektin tavoitteiden mukaan siirtämään Jiraan, mutta tuplatiketöinnin vuoksi tikettien käsittely ei kokonaisuudessaan muuttuisi sulavammaksi tai tehokkaammaksi ennen lisenssiressurssien paranemista. Projektin mukana tulleita hyötyjä olivat kuitenkin useamman järjestelmän käytön tarpeen väheneminen ja saapuvien tikettien sulavampi seuraaminen keskitetysti dashboard-näkymää hyödyntämällä. Lisäksi service deskin ohjedokumentaatio pystytään projektin puitteissa siirtämään Confluenceen aikaisemmista lähteistä, joten ohjeistustenhakuprosessia voidaan tehostaa paremmin Jiran yhteyteen kytkeytyväksi.

Kiireellisimpänä jatkosuunnitelmaehdotuksena projektille on ehdottomasti lisälisenssien hankinta, jolloin Jira Service Managementissa toimivat projektit voidaan ottaa suoraan koko

tukiorganisaation käyttöön. Tällöin saadaan täysipainoisesti käyttöön lisäosan uudet ominaisuudet, joista haluttiin alun perin tehostusta olemassa oleviin prosesseihin. Tämän jälkeen ohjelman uusia ominaisuuksia voidaan alkaa kehittää täysipainoisesti ja testata, miten niistä saadaan paras mahdollinen lisäarvo asiakkuuksien hallintaan ja esimerkiksi asiakasportaaliominaisuus voidaan lanseerata tukiasiakkuuksien käyttöön.

## 8 Yhteenveto

Opinnäytetyön tutkimuskysymyksiin vastaaminen onnistui toivotusti. Tutkimuskysymyksiin vastattiin ja tutkimuskysymyksiä tukemaan löytyi teoriaa, vaikka pääosa löydetyistä service desk -teoriasta koskikin lähinnä IT-käyttötuen kaltaisia service desk -organisaatioita. Opinnäytetyön tilaajan toiveesta opinnäytetyöhön sisällytettiin teoriaa muutosvastarinnasta ja muutosprosessista, koska yksi projektin tärkeimmistä seikoista pohjimmiltaan oli työntekijöiden tehokkuuden lisäksi työhyvinvoinnin lisääminen ja kahden erilaisen organisaation toimintatapojen sulauttaminen yhdeksi uudeksi prosessiksi. Muutosprosessiteoria osoittautui opinnäytetyön toteutuksen kannalta keskeiseksi projektin hallinnan ja siihen liittyvien tahojen kommunikaation ymmärtämisen kannalta. Tutkimuskysymys Lean-mallista ja siihen liittyvä teoria tuki niitä huomioita, joiden mukaan projektia johdettiin pääasiassa teoriassa esitellyn Lean-mallin arvojen mukaisesti.

Projektin aikana työ toteutettiin alun perin toivottujen raamien puitteissa ja loppukäyttäjiä pyrittiin mahdollisimman aktiivisesti ottamaan osaa Jiran uuden lisäosan kehitykseen ja palautteen antamiseen. Projektia tehdessä työn toteuttaja oppi perusteita käyttöönottoprojektin ja muutosprosessin hallinnasta ja mahdollisista vaikeuksista, jotka kannattaisi ottaa huomioon tulevia projekteja tehtäessä. Löydetty teoria tuki hyvin ennen projektin aloitusta tiedettyä sen suhteen, miten suuri vaikutus ihmisillä on muutosprojektin läpiviennin onnistumiseen. Projektin kaikki vaiheet saatiin toteutettua, tekniset tavoitteet täytettyä ja jatkossa tikettien käsittelyä on mahdollista toteuttaa työn tilaajan alkuperäisen toiveen mukaan uusi lisäosa päätyökaluna.

Lähitulevaisuudessa projektin aikana asennetun ja käyttöönotetun lisäosan käyttöä laajennetaan ja siihen rakennettuja yksittäisiä asiakasprojekteja kehitetään yksilöllisesti lisää mahdollisesti ilmenevien muutostarpeiden tai asiakastarpeiden mukaan. Samoin tiedossa on, että lisäosaan kuuluvia asiakasportaaleja tullaan kysynnän mukaan laajemmin aktivoimaan ja lisäosan toistaiseksi sisäiseen service desk -käyttöön tarpeettomat ominaisuudet, kuten asiakastyytyväisyyskysely, tullaan käytön laajentuessa ottamaan mukaan asiakasprosesseihin.

## Lähteet

Aarnikoivu, H. (2005). Kilpailu kiristyy ja asiakkaiden vaatimukset kasvavat. Teoksessa H.

Aarnikoivu, Onnistu asiakaspalvelussa. Alma Talent Oy. <https://verkkokirjahylly-almatalent-fi.ezproxy.hamk.fi/teos/DAJBHXCTDG#kohta:1>

Alahyari, H., Gorschek, T. & Svensson, R. (2018). An exploratory study of waste in software development organizations using agile or lean approaches: A multiple case study at 14 organizations. Information and Software Technology, 08/2018. <http://gorschek.com/wp-content/uploads/2019/12/An-Exploratory-Study-of-Waste-in-Agile-Software-Development-Organizations.pdf>

Atlassian. (n.d.-a). Service desk vs help desk vs ITSM: What's the difference? Haettu 25.4.2021 osoitteesta <https://www.atlassian.com/itsm/service-request-management/help-desk-vs-service-desk-vs-itsm>

Atlassian. (n.d.-b). A brief history. Haettu 25.4.2021 osoitteesta <https://www.atlassian.com/company>

Atlassian. (n.d.-c). Introducing Jira Service Management Haettu 25.4.2021 osoitteesta <https://www.atlassian.com/blog/announcements/introducing-jira-service-management>

Atlassian. (n.d.-d). What makes up Jira Core. Haettu 25.4.2021 osoitteesta <https://confluence.atlassian.com/jiracoreserver/what-makes-up-jira-core-938846151.html>

Atlassian. (n.d.-e). Getting started with Jira Core. Haettu 25.4.2021 osoitteesta <https://confluence.atlassian.com/jiracoreserver/getting-started-with-jira-core-938846183.html>

Atlassian. (n.d.-f). See what's possible with Jira Core. Haettu 25.4.2021 osoitteesta <https://confluence.atlassian.com/jiracoreserver/see-what-s-possible-with-jira-core-938846153.html>

Atlassian. (n.d.-g). Haettu 25.4.2021 osoitteesta

<https://confluence.atlassian.com/jiracoreserver/how-do-i-build-the-workflow-i-want-938846175.html>

Atlassian. (n.d.-h). Haettu 25.4.2021 osoitteesta <https://support.atlassian.com/jira-service-management-cloud/docs/what-is-jira-service-management/>

Atlassian. (n.d.-i). Automation: Basics. Haettu 25.4.2021 osoitteesta

<https://www.atlassian.com/software/jira/guides/expand-jira/automation>

Atlassian. (n.d.-j). Managing Service project notifications. Haettu 25.4.2021 osoitteesta

<https://confluence.atlassian.com/servicemanagementserver/managing-service-project-notifications-939926348.html>

Atlassian. (n.d.-k). Setting up SLAs. Haettu 25.4.2021 osoitteesta

<https://confluence.atlassian.com/servicemanagementserver/setting-up-slas-939926373.html>

Atlassian. (n.d.-l). What are queues. Haettu 25.4.2021 osoitteesta

<https://support.atlassian.com/jira-service-management-cloud/docs/what-are-queues/>

Atlassian. (n.d.-m). Set up a knowledge base so customers can serve themselves. Haettu 25.4.2021 osoitteesta

<https://support.atlassian.com/jira-service-management-cloud/docs/set-up-a-knowledge-base-so-customers-can-serve-themselves/>

Atlassian. (n.d.-n). Configuring the customer portal. Haettu 25.4.2021 osoitteesta

<https://confluence.atlassian.com/servicemanagementserver/configuring-the-customer-portal-939926277.html>

Atlassian. (n.d.-o). Jira Service Management: Customer Satisfaction Surveys (CSAT). Haettu 25.4.2021 osoitteesta

<https://confluence.atlassian.com/confeval/jira-service-management-evaluator-resources/jira-service-management-customer-satisfaction-surveys-csat>

Atlassian. (n.d.-p). Managing access to your service project. Haettu 25.4.2021 osoitteesta <https://confluence.atlassian.com/servicemanagementserver0415/managing-access-to-your-service-project-1044115269.html?permissionViolation=true>

Atlassian. (n.d.-q). Collecting customer satisfaction csat feedback. Haettu 25.4.2021 osoitteesta <https://confluence.atlassian.com/servicemanagementserver/collecting-customer-satisfaction-csat-feedback-939926566.html>

Atlassian. (n.d.-r). About Confluence. Haettu 30.7.2021 osoitteesta <https://www.atlassian.com/software/confluence/guides/get-started/confluence-overview#about-confluence>

Atlassian. (n.d.-s). An example of the rule builder. Haettu 25.4.2021 osoitteesta <https://www.atlassian.com/software/jira/guides/expand-jira/automation>

Axelos. (2019). ITIL 4 Edition. Axelos Limited 2019. Stationery Office Books, Storytel. (Alkuperäinen teos julkaistu 2019)

McCullough, A. (21.1.2020). Service desk - the value creating one-stop-shop. Axelos. <https://www.axelos.com/news/blogs/january-2020/service-desk-the-value-creating-one-stop-shop>

Cobb, C. G. (2011). Making Sense of Agile Project Management – Balancing Control and Agility (John Wiley & Sons, Incorporated). <https://ebookcentral-proquest-com.ezproxy.hamk.fi/lib/hamk-ebooks/reader.action?docID=661483>

Harned, D. (2018). Hands-on Agile Software Development with Jira – Design and manage software projects using the Agile methodology. Pakt Publishing LTD. <https://ebookcentral-proquest-com.ezproxy.hamk.fi/lib/hamk-ebooks/reader.action?docID=5520880>

Hjelt, M. & Syynimaa, N. (2021). Quality Management in Service Desk - How Does Service Desk Managers Define and Measure Quality. Faculty of Information Technology, University of Jyväskylä.

<https://jyx.jyu.fi/bitstream/handle/123456789/60404/1/hjeltsyynimaaqualitymanagement.pdf>

Jäntti, M., Shrestha, A. & Cater-Steel, A. (2012). Towards an Improved IT Service Desk System and Processes: A Case Study. International Journal on Advances in Systems and Measurements, vol 5

no 3 & 4. [https://eprints.usq.edu.au/22735/3/Jantti\\_Cater-Steel\\_Shrestha\\_PV.pdf](https://eprints.usq.edu.au/22735/3/Jantti_Cater-Steel_Shrestha_PV.pdf)

Kohler, S. (2013). Atlassian Confluence 5 Essentials – Learn how to install configure, and manage Atlassian Confluence 5 to build an enterprise-grade collaboration platform. Packt Publishing Ltd,

UK. <https://ebookcentral-proquest-com.ezproxy.hamk.fi/lib/hamk-ebooks/reader.action?docID=1192657>

Morgan, J. & Brenig-Jones, Martin. (2012). Lean six sigma for dummies. Chichester: Wiley 2nd ed.

Palmer, B. (2004). Making the Change work – Practical Tools for Overcoming Human Resistance to Change. ASQ Quality Press Milwaukee, Wisconsin. <https://ebookcentral-proquest-com.ezproxy.hamk.fi/lib/hamk-ebooks/reader.action?docID=3002559>

Ponteva, K. (2010). Onnistu muutoksessa. Talentum Media.

<https://verkkokirjahylly-almatalent-fi.ezproxy.hamk.fi/teos/GAJBHXCTDG#kohta:14>

Torkkola, S. (2015). Lean ansiantuntijatyön johtamisessa. Talentum Pro 2015.

<https://verkkokirjahylly.almatalent.fi/teos/BAJBEXCTEB>

Six Sigma. (n.d.). Tätä on Lean. Haettu 20.4.2021 osoitteesta

<http://www.sixsigma.fi/index.php/fi/lean/lean/>

Stellman, A. & Greene, J. (2014). Learning Agile: Understanding Scrum, XP, Lean and Kanban.

O'Reilly Media, Incorporated. <https://ebookcentral-proquest-com.ezproxy.hamk.fi/lib/hamk-ebooks/reader.action?docID=1840882>



Sweeney, B. (2017). Lean Six Sigma QuickStart Guide. Clyde Bank Media LLC. Storytel.  
(Alkuperäinen teos julkaistu 27.2.2017)

Wheatcroft, P. (2014). Service Desk and Incident Manager. BCS The Chartered Institute for IT, BCS Learning & Development Ltd. <https://ebookcentral-proquest-com.ezproxy.hamk.fi/lib/hamk-ebooks/reader.action?docID=1713961&query=Service+Desk>

## **Liite 1: Aineistonhallintasuunnitelma**

Kehitysprojektin aikana pidetään päiväkirjaa Jira-projektiin liittyvistä löydöksistä, kysymyksistä, edistymisen seurannasta ja palavereista. Päiväkirjaan kerätään teknistä tietoa muun muassa ohjelinkkejä projektista. Tätä tietoa analysoidaan työtä tehdessä opinnäytetyötä varten ja päiväkirja toimii muistiinpanojen säilytysalustana. Päiväkirjaa säilytetään tekijän tietokoneen C-aseamalla, ja siitä tehdään varmuuskopioita yrityksen Jira-projektille luodulle tiketille. Päiväkirjaa säilytetään C-aseamalla ainakin vuoden verran opinnäytetyön valmistumisesta. Päiväkirjaa ei laiteta julkisesti nähtäville eikä tämän työn liitteeksi, koska se sisältää asiakastietoja. Säilytetty päiväkirja tuhotaan viimeistään siinä vaiheessa, kun työn tekijä siirtyy toisen työnantajan palvelukseen. Siihen asti se säilytetään mahdollisia muita Jira-kehitysprojekteja varten pääasiallisesti yksityisenä dokumentaationa, josta voi löytää ohjelinkkejä ja mietintöjä aikaisemmasta projektista ja sen toteutuksesta.

Päiväkirjan lisäksi projektiin liittyviä tietoja, kysymyksiä ja kommunikaatiota kerätään Jira Coreen luodulle Lifecycle Services -organisaation kehitystiketille. Kehitystiketille kerätään kuvankaappauksia rakennetuista ratkaisuista ja välitetään sähköpostitse tai muuten saapuneita kommentteja, että tiedot löytyisivät mahdollisimman keskitetysti yhdestä paikasta. Kehitystiketille luodaan tarvittaessa alitikettejä eri projektivaiheiden selkeämmän seurannan mahdollistamiseksi. Käyttöönottoprojektin päättyessä tiketti suljetaan ja se säilyy Jirassa ja tiketti poistetaan toimeksiantajayrityksen normaalien käytäntöjen mukaisesti.