



Asiakasportaalin käyttäjäkokemuksen parantaminen

Ammattikorkeakoulututkinnon opinnäytetyö
Tieto- ja viestintätekniikka, insinööri (AMK)
Syyslukukausi 2023
Walteri Mäntykoski

Tieto- ja viestintäteknikka

Tekijä Waltteri Mäntykoski

Työn nimi Asiakasportaalin käyttäjäkokemuksen parantaminen

Ohjaaja Toni Laitinen

Tiivistelmä

Vuosi 2023

Tämä opinnäytetyö käsittelee ServiceNow-järjestelmää ja sen sisältämää Employee Center-portaaliratkaisua sekä Service Catalog lomakepohjia. Työssä käydään läpi erilaisia järjestelmän mahdollistamia automatisointimahdollisuuksia erilaisten järjestelmän sisältämien työkalujen avulla ja miten sillä voidaan parantaa asiakkaan portaalikokemusta. Portaalin sekä lomakkeiden tavoitteena on nopeuttaa sisäisten tikkettien käsittelyä vähentämällä manuaalista työtä ja ohjaamalla tikit automaattisesti oikeaan työjonoon.

Portaali toteutettiin ServiceNow:n Employee Center-portaaliratkaisuun ja lomakkeet rakennettiin Service Catalogiin sisältyvällä Catalog Builder-lomaketyökalulla. Automatisointiin hyödynnettiin ServiceNow:n client script-ominaisuuksia, UI policyjä sekä Flow Designer-työkalua.

Työn lopputuloksena saatiin aikaan sisäinen asiakasportaali, josta käyttäjä löytää helposti tarvitsemansa lomakkeet. Lomakkeissa kysytään kaikki kyseisen asian hoitamiseen tarvittava tieto, ja automaatio hoitaa tikketin oikean työryhmän työjonoon ilman, että käyttäjän tarvitsee välttämättä tietää, kenelle pyyntö kuuluu.

Avainsanat Automaatio, portaali, järjestelmä

Sivut 19 sivua

Information and communication technology

Author Waltteri Mäntykoski

Subject Customer service portal user experience improvement

Supervisors Toni Laitinen

Abstract

Year 2023

This thesis covers the ServiceNow system and its Employee Center portal solution. The thesis covers various automation possibilities included in the system by utilizing various different tools and methods and how they can be used to improve the customer portal experience. The goal of the portal and its forms is to speed up the processing of internal support tickets by reducing manual work and by automatically assigning the tickets to the correct group's ticket queue.

The portal was implemented with ServiceNow's Employee Center portal solution and the form templates were built using the Catalog Builder form tool. Automation was done by utilizing ServiceNow's client scripts, UI Policies and the Flow Designer tool.

The thesis resulted in a customer portal, where the end-user can easily find the ticket forms they need. The forms ask all the necessary information for that specific request, and by utilizing automation the ticket is assigned to the correct assignment group's queue without the user needing to know which group the submitted request belongs to.

Keywords Automation, portal, system

Pages 19 pages

Sisällys

	Sanasto.....	1
1	Johdanto.....	1
2	ServiceNow:n esittely.....	1
3	Toteutus.....	3
3.1	Työjonon automatisointi	4
3.2	Lomakkeiden toteutus.....	4
3.3	Client scriptit	5
3.4	GlideRecord.....	6
3.5	UI Policy	7
3.6	Työnkulku	8
3.7	Flow Designer.....	8
3.8	Portaalisivu	10
4	Testaus.....	12
4.1	Update set	13
4.2	Automated Test Framework.....	14
5	Tulokset ja jatkokehitys.....	14
6	Pohdinta.....	15
	Lähteet	17

Sanasto

API	Application Programming Interface, eli ohjelmistorajapinta.
ATF	Automated Test Framework, ServiceNow:n työkalu, joka mahdollistaa järjestelmän testauksen automatisoinnin. (ServiceNow, 2023l).
CMDB	Configuration Management Database, tietokanta, joka sisältää tiedot yrityksen laitteistoista ja ohjelmistoista.
CSS	Cascading Style Sheets, käytetään yleisesti verkkosivujen tyylimäärittäisiin.
Dictionary	ServiceNow:n Dictionary Entry taulu, joka sisältää kaikki järjestelmään luodut taulut ja niiden kolumnien määrittelyt.
Employee Center	ServiceNow:n portaaliratkaisu, joka on tarkoitettu yrityksen henkilöstölle. (ServiceNow, 2023c).
Flow Designer	ServiceNow:n työkalu, joka mahdollistaa prosessien automatisointia. (ServiceNow, 2023f).
GlideRecord	ServiceNow:n sisältämä API, jota käytetään tietokantaoperaatioihin. (ServiceNow, 2023e).
HR	Human Resources, eli henkilöstöhallinta.
HTML	HyperText Markup Language, merkintäkieli, jota käytetään verkkosivujen tekemiseen.
Impersonointi	ServiceNow:n ominaisuus, jolla järjestelmänvalvojat voivat kirjautua testausta varten toisen käyttäjän profiiliin käyttäen kyseisen käyttäjän oikeuksia.
IT	Information Technology, yleinen tietotekniikkaa tarkoittava lyhenne.

SaaS	Software as a Service, eli jakelumalli, jossa palvelun tarjoaja ylläpitää ohjelmistoa palvelimillaan ja julkaisee sitä internetin välityksellä tilauksena.
ServiceNow	yhdysvaltalainen ohjelmistokehitysyritys, joka kehittää ja ylläpitää saman nimistä järjestelmää. (ServiceNow, 2022).
Update set	ServiceNow:n ominaisuus, jonka avulla kehittäjä voi siirtää tekemiään muutoksia instanssista toiseen. (ServiceNow, 2023j).
Instanssi	ServiceNow-alustan yksittäinen ilmentymä, esim. testaus- tai tuotantoinstanssi.
ITSM	IT-palvelunhallinta (IT Service Management).
ITIL	Information Technology Infrastructure Library. Joukko käytäntöjä IT-palveluiden suunnitteluun, toteutukseen ja ylläpitoon.

1 Johdanto

Tämä opinnäytetyö käsittelee toimeksiantajan projektia, jossa tavoitteena oli parantaa yrityksen sisäistä itsepalveluportaalia ottamalla käyttöön uusi portaaliratkaisu.

Toimeksiantajalla on käytössään SaaS-tyyppinen ServiceNow-järjestelmä ja käyttöön otetaan ServiceNow:n sisältämä Employee Center-portaaliratkaisu, joka on tarkoitettu yrityksen sisäisille käyttäjille.

Työssä oli tavoitteena luoda helppokäyttöisempi asiakasportaali yrityksen sisäisille asiakkaille. Tavoitteena oli luoda siisti ja helppokäyttöinen portaali, josta sisäinen käyttäjä löytää tarvitsemansa lomakepohjat erilaisten häiriö- tai palvelupyyntöjen tekemiseen.

Väärään työjonoon johtavia tai puutteellisilla tiedoilla täytettyjä pyyntöjä tai häiriöilmoituksia voidaan ehkäistä luomalla käyttötarkoitukseen spesifisiä lomakkeita, jotka pyytävät käyttäjää täyttämään aiheeseen liittyvät vaaditut tiedot ja yleisesti opastavat käyttäjää, jotta käyttäjä ei voi lähettää lomaketta ilman vaadittuja tietoja ja vikaa ratkaiseva ryhmä voi aloittaa työn mahdollisimman pian ilman jatkokysymyksiä.

Lomakkeet mahdollistavat myös niihin liittyvien prosessien automatisointia. Käyttäjä ei välttämättä tiedä mihin työjonoon hänen pyyntönsä tai vikailmoituksensa kuuluu. On myös mahdollista, että esimerkiksi erilaisten laitteiden tai järjestelmien ylläpitovastuussa olevat ryhmät vaihtuvat, jolloin lomakkeen dynaamisesti toimiva automatisointi varmistaa, että pyyntö menee automaattisesti mahdollisimman oikeaan työjonoon.

2 ServiceNow:n esittely

ServiceNow on ohjelmistokehittäjäyritys, joka on perustettu Yhdysvalloissa vuonna 2003. (ServiceNow, 2022).

ServiceNow tuottaa saman nimistä palvelunhallintajärjestelmää, joka sisältää useita palvelukokonaisuuksia, niihin sisältyy muun muassa: (ServiceNow, 2023).

- IT Service Management (ITSM)
- HR Service Delivery
- Customer Service Management (CSM)
- Employee Center

ServiceNow on pilvipohjainen SaaS palvelunhallinta-alusta, jonka avulla voidaan yhdistää organisaation eri liiketoiminnot saman järjestelmän alle. Yksinkertaisella tasolla järjestelmää voisi kuvata työkaluna, joka mahdollistaa pyyntöjen ja häiriöiden tiketöinnin ja seurannan sekä säännöllisten IT-palvelupyyntöjen käsittelyn ja katalogisoinnin. ServiceNow:n tikettityyppeihin sisältyvät mm.: (GetApp, 2023).

- Incident
- Change Request
- Problem
- Service Request

Nämä ovat ITIL-ohjesääntöjen mukaisia termejä erilaisille tikettityypeille. Incidentit ovat häiriötikettejä, eli incident avataan, kun jokin ei toimi halutulla tavalla. Change requestit ovat muutospyyntöjä, eli change request avataan, kun johonkin halutaan muutos. Tämä voi olla esim. uuden palvelun käyttöönotto tai poisto, tai esim. uuden ominaisuuden kehitys. Problem eli ongelmatiketti on hieman samanlainen kuin incident, mutta problemia käytetään lähinnä incidentin juurisyyn tiketöimiseen ja ratkaisemiseen. Service Request on palvelupyyntötiketti, jota käytetään yleisiin IT-palveluihin kuten esim. salasanan resetoimiseen tai laitteiston hankintaan.

IT Service Management on ServiceNow:n ITSM ratkaisu, joka on suunniteltu IT-palveluiden hallintaan organisaatiossa. ITSM-paketti tarjoaa monia työkaluja ja ominaisuuksia, jotka auttavat organisaatiota parantamaan IT-toimintansa tehokkuutta. ITSM-paketti sisältää tiketöintijärjestelmän muun muassa häiriöiden, muutosten ja palvelupyyntöjen hallintaan. (ServiceNow, 2023a)

Customer Service Management on Asiakaspalvelun hallinnan ratkaisu, joka sisältää erilaisia työkaluja asiakkaan palvelemiseen. CSM-moduulilla asiakas voi ottaa palvelupisteeseen yhteyttä parhaaksi katsomallaan tavalla muun muassa sähköpostitse, puhelimitse, chatin tai sosiaalisen median kautta millä tahansa laitteella. Älykäs työnkulku voi tarvittaessa ohjata asiakastiketit automaattisesti oikealle henkilölle/ryhmälle tiketin kontekstin perusteella. (ServiceNow, 2022b).

HR Service Delivery (HRSD) on ServiceNow:n moduuli, joka sisältää monia työntekijöille hyödyllisiä ominaisuuksia. HRSD:n avulla voidaan suoraviivaistaa erilaisia HR-palveluita yhdelle ja samalle portaaliratkaisulle. Sisältää esimerkiksi Employee Center-portaaliratkaisun, jota voidaan hyödyntää sisäisenä portaalina työntekijöille. (ServiceNow, 2023b).

Service Catalog on ServiceNow:n sovellus, joka mahdollistaa palvelukatalogien luomisen. Service Catalogin avulla voidaan luoda yksinkertaisia lomakepohjia, joita voidaan julkaista sekä sisäisille että ulkoisille asiakkaille ServiceNow:n erilaisten portaaliratkaisujen kautta tai suoraan ServiceNow:n käyttöliittymän kautta.

Employee Center on moniosainen ja dynaaminen portaalit yrityksen sisäisille käyttäjille. Portaalin keskeisiä ominaisuuksia ovat mm.:(ServiceNow, 2023c).

- Muokattavat portaalisivut
- Yhtenäistetty selaus kaikille sisältötyypeille käyttäen yhtenäistä taksonomiamallia
- Sisäänrakennetut widgetit ja sivut joilla, yritys voi räätälöidä sivut halutunlaiseksi
- Integroidut työnkulut, joilla automatisoidaan yleisiä prosesseja
- Mahdollisuus luoda kuratoituja sivuja suosituista aiheista
- Työntekijäkeskeinen navigointi

Employee Center parantaa työntekijöiden tuottavuutta tekemällä tietojen ja palveluiden etsimisestä helpompaa. Portaalin avulla tietojen ja palvelujen etsiminen on tehty yksinkertaiseksi.

3 Toteutus

Jotta portaalista saatiin toimiva kokonaisuus, tarvittiin siihen useita osa-alueita. Lähtötilanteessa häiriö- tai palvelupyyntötiketin lähettäminen itsepalveluna on ollut haastavaa. Käyttäjä ei välttämättä tiedä kaikkia tietoja, mitä tikettiä ratkaiseva ryhmä tarvitsee tiketin hoitamiseksi. Tiketin ratkaisu nopeutuisi merkittävästi, jos tiketin käsittelijän ei tarvitse esimerkiksi esittää jatkokysymyksiä vaan pääsee heti aloittamaan työskentelyn. Hyödyllistä olisi myös, jos tiketti menisi suoraan oikeaan työjonoon sille ryhmälle, joka kyseistä laitetta/järjestelmää tai muuta vastaavaa ylläpitää. Tämä on mahdollista toteuttaa automatisoinnilla.

Vanhassa järjestelyssä oli huomattu lukuisia parannuskohteita. Koska lomakkeita ei ollut räätälöity tiettyihin aihealueisiin, ei käyttäjältä voitu vaatia kovin spesifistä aiheeseen liittyvää tietoa vaan lomake koostui käytännössä otsikosta ja kuvauksesta, eikä lomake opastanut käyttäjää. Tämän seurauksena oli myös mahdotonta tunnistaa tiketin sisällöstä automaattisesti, että mihin pyyntö tai häiriö liittyy, jolloin tikettien käsittelijät joutuivat manuaalisesti siirtämään tikettejä oikeisiin joihin.

Tavoitteena oli myös vähentää tiketöinnin kynnystä, sillä käyttäjillä oli usein tapana lähettää sähköpostia tai viestitellä eri chattikanavilla ja pyytää niiden kautta apua erilaisiin ongelmiin. Tiketin teon tulisi tehdä sen verran yksinkertaiseksi, että se olisi käyttäjän ensisijainen kanava avunpyyntöön suht. pienissäkin asioissa. Tällä on lukuisia etuja, mm. se, että tiketistä jää järjestelmään jälki, jonka avulla voidaan tulevaisuudessa tarkistaa miksi tai miten jokin muutos on tehty ja nähdään suoraan, kuka ne on tehnyt. Näin voidaan varmistua, että muutoksiin liittyvä tieto ei katoa ja vältytään turhalta selvitystyöltä tulevaisuudessa. Tikettien käsittelijät myös kirjaavat työtuntejaan ServiceNow-tiketeille, joka voi olla hyödyllistä dataa tulevaisuudessa esimerkiksi erilaisten tikettien ratkaisuun käytetyn ajan selvittämiseen.

3.1 Työjonon automatisointi

Jotta käyttäjien pyynnöt voitaisiin ohjata mahdollisimman oikeaan työjonoon automaattisesti, piti keksiä keino, miten sellainen voitaisiin toteuttaa. Helpoin tapa olisi, jos käyttäjälle annettaisiin lomakkeille lista, josta valita minkä ryhmän työjonoon tiketti lähetetään. Tämä ei ole toimiva ratkaisu, koska laitteita, järjestelmiä ja niitä ylläpitäviä ryhmiä on aivan liikaa jokaisen työntekijän ulkoa muistettavaksi.

Organisaation laitteet, järjestelmät ja projektit ovat tallennettuna ServiceNow:n CMDB (Configuration Management Database), eli konfiguraatiohallintatietokantaan. Konfiguraatiohallinnan kautta laitetta tai järjestelmää ylläpitävä taho voi muuttaa kyseiseen kohteeseen liittyviä tietoja. Koska moniin lomakkeisiin liittyi järjestelmän tai laitteen valinta, vaikutti se hyvältä paikalta tallentaa myös työryhmätieto.

Jotta tiettyyn järjestelmään tai laitteeseen liittyvä tiketti menee automaattisesti oikeaan työjonoon, päätettiin luoda CMDB-tauluun uusi kolumni nimeltä "Primary assignment group", eli päätyöryhmä. Kentästä tehtiin referenssityyppinen, joka tarkoittaa, että kenttä viittaa toisessa järjestelmän taulussa olevaan tietoon. Kenttä laitettiin viittamaan järjestelmän käyttäjäryhmätauluun sys_user_group. Tämä mahdollisti sen, että jokaiselle CMDB-taulusta löytyvälle kohteelle voitiin valita oikea pääryhmä, ja tätä tietoa voidaan sitten hyödyntää lomakkeiden automatisoinnissa myöhemmin hakemalla kyseinen tieto laitteen tai järjestelmän kautta.

3.2 Lomakkeiden toteutus

Lomakkeiden toteutukseen käytettiin ServiceNow:n Catalog Builder työkalua, joka on Service Catalogiin sisältyvä editorityökalu lomakkeiden tekoon. Toimeksiantaja laati listan tarvittavista erilaisista lomakkeista sekä niissä tarvittavista kentistä. Lomakkeilla kysytään

kyseiseen aiheeseen liittyvät tarvittavat tiedot pakollisina kenttinä. Kuvan 1 lomake on hyvä esimerkki sellaisesta lomakkeesta, jonka Catalog Builderilla voi tehdä.

Kuva 1 Esimerkki luodusta lomakkeesta

Uuden järjestelmän käyttöönotto

Uuden järjestelmän käyttöönotto

* Otsikko

* Kuvaus


* Uuden järjestelmän nimi

* Järjestelmän selkokielineen kuvaus

* Järjestelmän omistava liiketoiminta

* Järjestelmävastaava

* Pääkäyttäjä

 Add attachments

Order Now

Required information

Otsikko Kuvaus

Uuden järjestelmän nimi

Järjestelmän selkokielineen kuvaus

Järjestelmävastaava Pääkäyttäjä

Lomakkeille luotiin myös käyttäjältä piilotettuja kenttiä, joita hyödynnetään automaatiassa. ServiceNow mahdollistaa Service Catalogissa skriptin liittämisen kenttiin Catalog Client Scripteillä. Client scriptit ovat ServiceNowssa käyttäjän puolella tapahtuvia muutoksia, eli luodut client scriptit ajetaan käyttäjän käyttöliittymässä eivätkä ServiceNow:n palvelimella.

Skriptaukseen ServiceNow käyttää ECMAScript5-standardin mukaista JavaScript-ohjelmointikieltä (ServiceNow,2023d). ServiceNow:lla on myös lukuisia dokumentoituja ohjelmointirajapintoja (API) joita skripteissä voidaan hyödyntää erilaisiin toiminnallisuuksiin.

3.3 Client scriptit

Client script voidaan käynnistää kolmella eri tavalla:

- onLoad
- onChange
- onSubmit

onLoad tyyppiset skriptit ajetaan siinä yhteydessä, kun käyttäjä avaa lomakkeen. Tämä on hyödyllinen esimerkiksi tilanteessa, jossa käyttäjän nimi, puhelinnumero tai sähköpostiosoite halutaan täyttää lomakkeelle automaattisesti hakemalla se kirjautuneen käyttäjän tiedoista. onChange tyyppisiä skriptejä käytetään tilanteissa, joissa lomakkeella halutaan tapahtuvan jokin muutos, kun client scriptille määritettyä kenttää muutetaan. onSubmit skriptit ajetaan sillä hetkellä, kun käyttäjä painaa lomakkeen tilauspainiketta.

Nämä kaikki ovat eri tavoin hyödyllisiä, mutta tässä käyttötapauksessa käytettiin onChange skriptiä lomakkeen mukaan joko järjestelmän tai laitteen valintakentässä. Kun käyttäjä valitsee listasta laitteen, onChange skripti käynnistyy ja suorittaa queryn, eli tietokantahaun järjestelmästä. Skriptissä hyödynnetään ServiceNow:n sisältämää GlideRecord ohjelmointirajapintaa. Kuvassa 2 on esimerkki client scriptistä, joka hakee häiriöilmoituslomakkeelle automaattisesti oikean työryhmän (assignment group) valitun järjestelmän perusteella.

3.4 GlideRecord

GlideRecord on ServiceNow:n sisältämä API, joka on tarkoitettu ServiceNow:n tietokantaan liittyviin operaatioihin. GlideRecord mahdollistaa tietojen hakemisen, päivittämisen ja lisäämisen ServiceNow:n tietokantaan ja on olennainen osa ServiceNow:n skriptausympäristöä. (ServiceNow, 2023e).

GlideRecordia hyödyntämällä voitiin rakentaa skripti, joka haki käyttäjän valitseman laitteen tai järjestelmän perusteella kyseiselle laitteelle tai järjestelmälle määritetyn pääryhmän. Jotta ryhmätieto saatiin talteen, tehtiin lomakkeille uusi käyttäjältä piilotettu muuttuja, johon skriptillä haettu ryhmä tallennettiin. Näin saatiin automaattisesti haettua oikea työryhmä lomakkeelle. Tätä piilotettua kenttää voidaan hyödyntää myöhemmin työnkulussa (ks. työnkulku) asettamaan lomakkeelle tallennettu ryhmä varsinaiselle tiketille, ja näin tiketti menee automaattisesti oikeaan työjonoon ilman, että käyttäjän tarvitsee itse tietää, että kenen hoidettavaksi tiketti kuuluu.

Kuva 2 esimerkki Catalog Client Scriptistä

New client-scripts are run in strict mode, with direct DOM access disabled. Access to jQuery, prototype and the window object are likewise disabled. To disable this on a per-script basis, configure this form and add the 'Isolate script' field. To disable this feature for all new globally-scoped client-side scripts set the system property 'glide.script.blockclient.globals' to false.

Name: Set SO and AG
 Applies to: A Catalog Item
 Active:
 UI Type: All

Application: Global
 Type: onChange
 Catalog Item: Ilmoita häiriöstä
 Variable name: system

Applies on a Catalog Item view:
 Applies on Target Record:

```

1 function onChange(control, oldValue, newValue, isLoading) {
2     if (isLoading || newValue == '') {
3         return;
4     }
5     //käyttäjän lomakkeelle valitsema järjestelmä
6     var sys = g_form.getReference('system', setCurrentValues);
7     var gr = new GlideRecord('u_cmdb_ci_system');
8     gr.addQuery('sys_id', sys);
9     gr.query();
10    function setCurrentValues(sys) {
11        //asetetaan järjestelmälle kuuluva projekti
12        g_form.setValue("service_offering", sys.u_service_offering);
13
14        //jos järjestelmälle ei löydy pääryhmää, laitetaan fallbackina IT:n jonoon
15        if(sys.u_primary_assignment_group == '')
16        {
17            g_form.setValue("assignment_group", '45493f17db22045061763412f596192b');
18        }
19        else{
20            //asetetaan pääryhmäksi määritetyn ryhmän tikettijonoon
21            g_form.setValue("assignment_group", sys.u_primary_assignment_group);
22        }
23    }
  
```

Koska voi olla myös tilanne, että oikeaa järjestelmää tai laitetta ei jostain syystä löydy ServiceNowsta, täytyy olla myös varajärjestely niille tapauksille. Tähän hyödynnettiin Service Catalogin Catalog UI Policyjä.

3.5 UI Policy

UI policyt eli käyttöliittymäsäännöt mahdollistavat lomakkeella olevien kenttien ominaisuuksien muuttamisen riippuen siitä, mitä käyttäjä on lomakkeella valinnut. UI Policyssä annetaan ehto, jolla sääntö toteutuu (When to apply) ja varsinaiset tapahtumat, jotka sääntö tekee, mikäli ehto toteutuu. Actioneissa valitaan lomakkeella oleva muuttuja (variable), joka voi olla esim. lomakkeella oleva kuvauskenttä tai valintalista, ja mitä muuttujalle halutaan tehdä. Vaihtoehtoina actioneissa on muuttaa kentän pakollisuutta, näkyvyyttä tai muuttaa sen muokattavuutta sekä määrittää että tyhjeneekö muuttujan arvo muutoksen yhteydessä. (ks. kuva 3).

Jotta käyttäjä voi silti poikkeustilanteessa lähettää tiketin, luotiin uusi "muu" valintaruutu sellaisille lomakkeille, jotka vaativat järjestelmä- tai laitevalinnan. Tähän valintaruutuun liitetty UI policyn ehto määritettiin niin, että jos "muu" valintaruutu on rastittu, niin järjestelmä piilottaa järjestelmä-/laitevalinnan ja antaa käyttäjän lähettää lomakkeen ilman valintaa. Koska työryhmän olisi tarkoitus täytyä automaattisesti järjestelmävalinnan perusteella, se ei tässä poikkeustilanteessa voi toimia ja tiketti ohjautuu oletuksena tiketin luonteen mukaan joko yrityksen service deskin tai IT-tuen jonoon.

Kuva 3 Esimerkki Catalog UI Policystä

UI policy = If "other" checkbox is checked

Applies to: A Catalog Item

Application: Global

* Catalog item: Ilmoita häiriöstä

* Short description: If "other" checkbox is checked

When to Apply

Script

Catalog UI policy actions are applied only if all the following conditions are met:

1. The catalog UI policy is Active
2. The items in the Conditions field evaluate to true
3. The field specified in the catalog UI policy is present on the specified catalog item

Catalog Conditions: Add Filter Condition Add "OR" Clause

other is true

Applies on a Catalog Item view

Applies on the Target Record

Apply the catalog UI policy actions when the form is loaded or when the user changes values on the form

On load

Reverse the effects of the catalog UI policy actions when the Conditions evaluate to false

Reverse if false

Update Delete

Related Links

[Add to Update Set](#)

[Run Point Scan](#)

UI policy = If "other" checkbox is checked

Name	Read only	Mandatory	Visible	Order
system	Leave alone	False	False	100

3.6 Työnkulku

ServiceNow mahdollistaa työnkulkujen eli workflowien tekemisen. Työnkulkuja voidaan hyödyntää monilla eri tavoilla asioiden automatisointiin ja esim. vähentää turhaa manuaalista työtä automatisoimalla toistuvia tehtäviä. Työnkulku koostuu triggeristä, eli ehdosta, jonka toteutuessa työnkulku käynnistyy, sekä actioneista eli tapahtumista jotka työnkulku tekee käynnistyessään.

Työnkululla voidaan asettaa valmiiksi tiettyjä arvoja, jotka pysyvät aina samana, tai esimerkiksi vaikka kopioida lomakkeen kentästä tieto tiketin kenttään (esim. käyttäjän kirjoittama selvitys voidaan kopioida tiketin kuvauskenttään (description)). Työnkulkuun voidaan myös asettaa ehtoja, jolloin tiketin kulku tekee eri asioita työnkulkuun määritettyjen ehtojen perusteella.

3.7 Flow Designer

Tässä projektissa portaalin lomakkeisiin liitettävä työnkulku toteutettiin ServiceNow:n sisältämällä Flow Designer-työkalulla. Flow Designer on ServiceNow:n ominaisuus, joka mahdollistaa työnkulkujen tekemisen visuaalisesti selkokielellä, ilman että käyttäjän tarvitsee

välttämättä koskea koodiin ollenkaan, mutta sisältää myös skriptausmahdollisuuden sekä muita ominaisuuksia edistyneemmille käyttäjille. (ServiceNow, 2023f).

Työnkulku Flow Designerissa koostuu kolmesta pääosasta:

- **Trigger**, eli ehto, jonka toteutuessa työnkulku käynnistyy.
- **Actions**, eli tehtävät mitä työnkulku suorittaa.
- **Error handler**, eli mekanismi, joka käynnistyy, jos työkulussa ilmaantuu virhe (error).

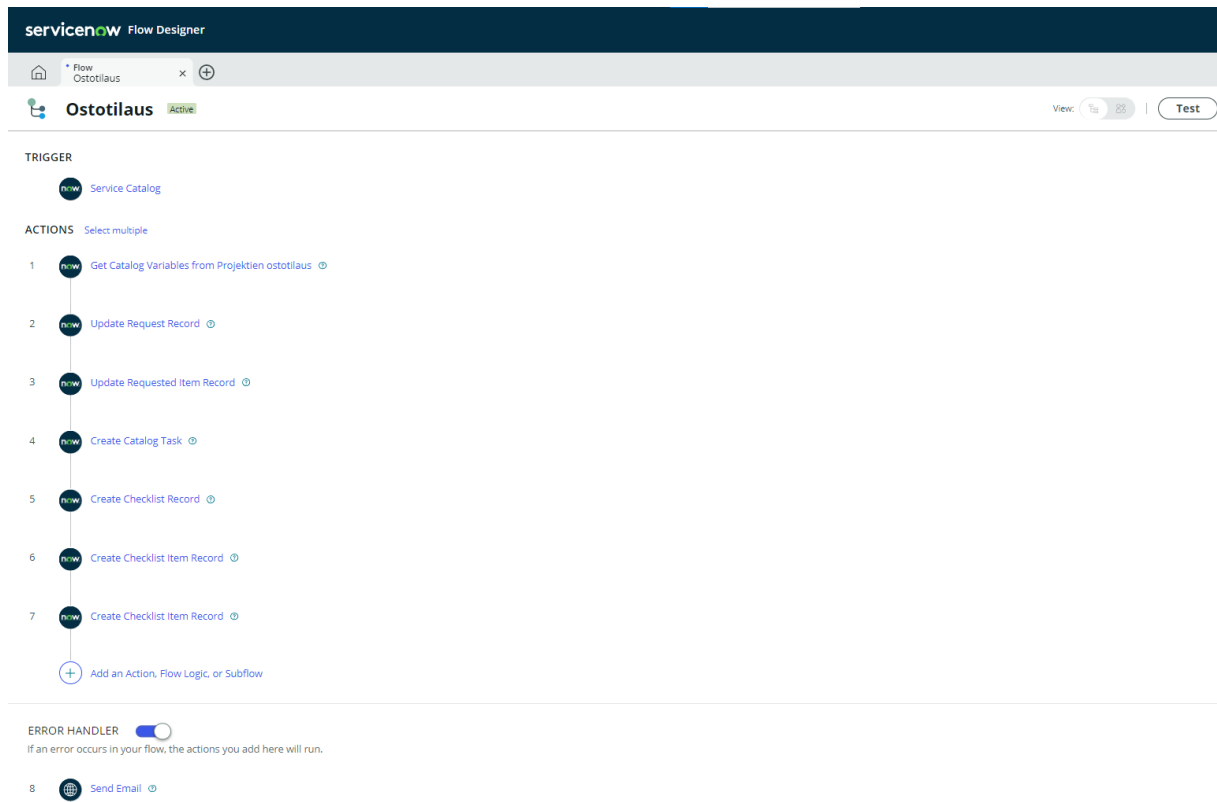
Triggerinä voidaan käyttää monia eri vaihtoehtoja, se voi olla esimerkiksi edellä mainitun lomakkeen (catalog item) lähettäminen. Tämän lisäksi triggerinä voidaan käyttää mm. järjestelmään saapuvaa sähköpostia, tai vaikka ajastaa työnkulku käynnistymään tietyn aikavälein tai tiettyinä kalenteripäivinä. Triggeriin voidaan myös määrittää lisäehtoja, joilla voidaan rajata työnkulun käynnistymistä. Esimerkiksi saapuvissa sähköposteissa voidaan määrittellä, että sähköpostin pitää tulla tietyistä domainista, tai viestin pitää sisältää jokin tietty avainsana. Triggerien kanssa pitää olla tarkkana, jotta niihin määritetyt ehdot eivät toteudu vahingossa muissa tilanteissa.

Tehtäviin määritetään kaikki tapahtumat mitä työnkulun halutaan suorittavan. Actioneilla voi mm. lähettää sähköposteja halutuille henkilöille, päivittää tiketin tietoja tai luoda alitikettejä (task). Työnkulkuun voidaan myös määrittää ehtoja erillisillä if(jos)-lausekkeilla, eli jossa työnkulkuun määritetty asia suoritetaan työkulussa vain, jos if-lausekkeeseen määritetty ehto toteutuu, ja jos ei niin työnkulku jättää kyseisen actionin tekemättä ja siirtyy työkulussa eteenpäin. Actioneihin on myös mahdollista luoda wait-tyyppinen ehto, jolla voidaan määrittää työnkulku odottamaan siihen asti, että määritetty ehto toteutuu. Työnkulkuun voidaan myös liittää alityönkulkuja (subflow), jolloin action suorittaa jonkin toinen kokonaan erillisen työnkulun. Kuvassa 3 on esimerkki yhdelle palvelupyntölomakkeelle laaditusta työkulusta ja esimerkki erilaisista actioneista, joita työkulkuun voidaan valita.

Error handleria käytetään mahdollisten työkulussa ilmenevien virheiden raportoimiseen. Error handleriin voidaan liittää actioneita samaan tapaan kuin itse työkulkuun, jonka avulla voidaan esim. lähettää sähköpostiviesti organisaation ServiceNow-ylläpitäjille tapahtuneesta virheestä lokitietoineen, jotta ongelma voidaan ratkaista. Error handlerin avulla voidaan varmistua, että vaikkapa asiakkaan lähettämä pyyntö menee onnistuneesti perille asti, eikä käy niin että asiakkaan pyyntö jää täysin tiketöimättä ja asiakas kyselee vihaissena perään, että mitä hänen lähettämälleen pyynnölle on tapahtunut. Esimerkiksi kuvassa 4 lähetetään

error handlerin lauetessa sähköpostia, joka tässä tapauksessa menee ylläpitotiimille ja sisältää virheen tiedot.

Kuva 4 Esimerkki Flow Designer työnkulusta



3.8 Portaalisivu

Portaaliin luotiin käyttäjälle helppokäyttöinen etusivu (dashboard) josta käyttäjä näkee silmäyksellä avaamansa tiketit ja niiden tilat, sekä löytää helposti tarvitsemansa lomakepohjat.

Kategoriointiin hyödynnettiin Employee Centerin **Taxonomy**-ominaisuutta. Taxonomy merkitsee ServiceNow:ssa kategoriaa tai luokittelua, jonka tarkoituksena on yhdistää erilaisia sisältötyyppejä samojen otsikoiden alle. Toimeksiantaja oli laatinut listan erilaisista aihealueista, joiden alle lomakkeet lisätään. Taxonomyihin lisättiin aihealueet pääotsikoittain, ja näiden alle alaotsikot, joiden alle varsinaiset lomakkeet tulevat.

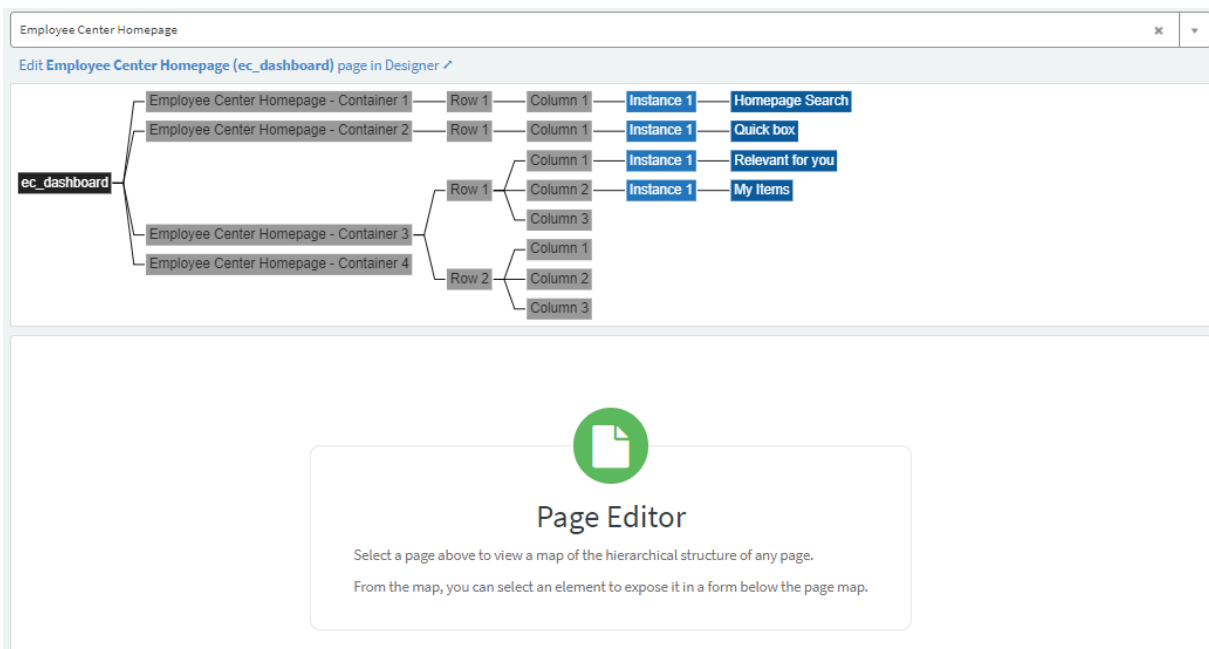
Etusivun muokkaamiseen käytettiin ServiceNow:n Service Portalin konfiguraatiotyökalua. Työkalu sisältää Brand Editorin, designerin, page editorin ja widget editorin.

Brand Editoria käytetään nimensä mukaan portaalisivujen brändäyksen muuttamiseen. Sillä voidaan asettaa portaalisivuille organisaation logo sekä määrittää sivujen värimaailmaa, joita käytetään oletuksena kaikilla portaalisivuilla. (ServiceNow, 2023g).

Designer on sivun sisällön muokkaamiseen tarkoitettu visuaalinen muokkaustyökalu, jolla portaalisivuja voidaan luoda ja muokata halutunlaisiksi. Designer mahdollista erilaisten widgettien lisäämisen sivulle hiirellä raahaamalla. (ServiceNow, 2023h).

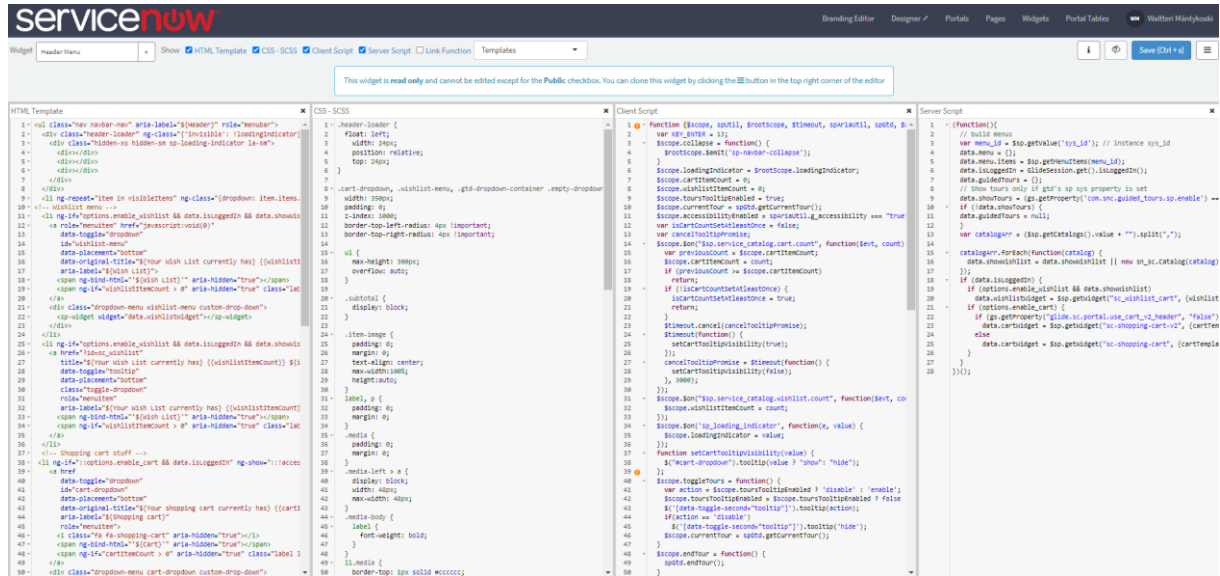
Page Editor antaa käyttäjälle hierarkkisen näkymän portaalisivusta ja sillä olevista elementeistä ja widgeteistä. Näkymässä voi klikata haluamaansa elementtiä ja pääsee helposti muokkaamaan siihen liittyviä asetuksia. Kuvassa 5 on esimerkki Page Editorin näkymästä. (ServiceNow, n.d.).

Kuva 5 Esimerkki Page Editorin hierarkianäkymästä



Widget Editor on widgettien eli pienoishjelmien tekoon tarkoitettu työkalu. Widget Editorista käyttäjä pääsee muokkaamaan widgettien HTML ja CSS arvoja sekä muokkaamaan widgettien client ja server scriptejä. ServiceNow sisältää paljon valmiiksi luotuja widgettejä, joita voi hyödyntää suoraan tai muokata omaa käyttöä varten, mutta järjestelmän omia widgettejä ei ole mahdollista muokata suoraan, vaan niistä täytyy tehdä kopio, koska oletuswidgetit ovat kriittisiä monissa paikoissa. (ServiceNow, n.d.b).

Kuva 6 Widget editorin muokkausnäky



Linkit lomakkeisiin lisättiin portaalisivujen yläreunassa sijaitsevaan navigointipalkkiin. Koska lomakkeille asetettiin kategoriat ja alikategoriat Employee Centerin taxonomyjä hyödyntäen, lisäys oli yksinkertaista ja lomakkeiden linkit olivat automaattisesti järjestyksessä eri otsikoiden alla portaalien yläpalkissa (ks. kuva 7).

Kuva 7 Navigointipalkki ja kategoriat



4 Testaus

Jotta lomakkeet saatiin testattua toimivaksi, hyödynnettiin lomakkeiden testauksissa ServiceNow:n imitointiominaisuutta (**impersonate**). Imitointi antaa järjestelmänvalvojen oikeuden kirjautua muille järjestelmän käyttäjille ja sen avulla järjestelmänvalvoja voi tarkistaa, millä järjestelmä näyttää tavallisen käyttäjän näkökulmasta. Tämä on todella hyödyllinen tilanteissa, joissa pitää varmistaa, että käyttäjällä on riittävät oikeudet jonkin uuden ominaisuuden käyttämiseen, tai jos halutaan varmistaa, että jokin tieto näkyy käyttäjälle oikein. (ServiceNow, 2023i).

Toimeksiantajalla oli käytössään ServiceNow:sta kehitys-, testi- sekä tuotantoinstanssit. Työtä tehtiin kehitysinstanssissa (dev), ja kun työ oli valmis, se siirrettiin testiympäristöön ServiceNow:n sisältämällä **update set**-ominaisuudella lopullista testaamista varten.

4.1 Update set

Update setit eli päivityspaketit ovat ServiceNow:n kehittäjille tarkoitettu ominaisuus, joka mahdollistaa tehdyn työn siirtämisen instanssista toiseen, esimerkiksi kehitysympäristöstä testi- tai tuotantoympäristöön, koska luonnollisesti mitään muutoksia ei tulisi tehdä suoraan tuotantoympäristöön siihen liittyvien riskien takia. Update set on eräänlainen lista, joka kerää muistiin kaikki käyttäjän järjestelmään tekemät muutokset, esimerkiksi erilaiset konfiguraatiomuutokset, jolloin ne voidaan replikoida toisessa instanssissa. Update setit osaavat myös tarkistaa muutokset mahdolliset konfliktien varalta, eli jos esimerkiksi testi- tai tuotantoympäristössä on eri versio jostain muutetusta tiedosta, jota update setissä on muutettu, update set ei anna tehdä kyseistä muutosta, kunnes käyttäjä on tarkistanut kyseisen konfliktin ja joko hypännyt kyseisen muutoksen yli, tai hyväksynyt muutoksen tekemisen. (ServiceNow, 2023j).

Update setit ovat erittäin hyödyllinen ominaisuus mutta siinä on myös pieniä riskejä. Koska käyttäjällä voi olla työn alla samaan aikaan useita eri projekteja, joille on luotu omat update settinsä, on mahdollista, että käyttäjällä on esimerkiksi väärä update set valittuna. Tämä voi johtaa siihen, että toiseen projektiin liittyviä muutoksia siirtyy väärän projektin mukana tai osa muutoksista jää puuttumaan. Kehittäjän on hyvä tarkistaa update setin sisältö ennen sen siirtämistä, mutta laajemmissa projekteissa voi olla satoja eri muutoksia, jolloin listasta tulee pitkä ja jotain muutoksia saattaa jäädä huomaamatta.

Tässä etuna on se, että toimeksiantajalla kehitys- ja testiympäristö ovat erillään. Koska update setti siirretään ensin testiympäristöön, voidaan varmistua, että se toimii siirron jälkeen testiympäristössä halutulla tavalla. Näin voidaan melko varmasti varmistua siitä, että siirretyt muutokset toimivat myös tuotantoinstanssissa toivotulla tavalla eikä update setin mukana siirtynyt mitään epätoivottua.

Update setit mahdollistavat myös muutosten perumisen, mikäli huomataan että tehty muutos ei jostain syystä toimi tai jokin menee pahasti pieleen, niin muutokset voidaan perua ja muutetut tiedot muuttuvat takaisin vanhoihin versioihinsa. (ServiceNow, 2023k).

4.2 Automated Test Framework

ServiceNow sisältää myös mahdollisuuden automaattisiin testeihin Automatic Test Framework (ATF) työkalulla. Automaattitestejä ei hyödynnetty tässä projektissa, mutta niistä on ollut apua aikaisemmissa työstämissäni projekteissa.

Automaattitestin avulla käyttäjä voi määrittää askel kerrallaan, mitä käyttöliittymässä pitää testata. Käytännössä käyttäjä määrittelee askel kerrallaan, mitä järjestelmän pitäisi syöttää mihinkin kenttään, ja mitä painikkeita tulisi painella, ja voi pyytää järjestelmää varmistamaan, että esim. luotu tiketti löytyy lähettämisen jälkeen järjestelmästä tai tieto tallentuu oikein. Jos automaattitesti ei syystä tai toisesta pääse etenemään seuraavaan vaiheeseen tai jokin varmistus ei mene läpi, ilmoittaa testi virheestä. Tämä on hyödyllinen ominaisuus esimerkiksi järjestelmän versiopäivitysten yhteydessä, jotta voidaan varmistua, että kaikki kriittiset järjestelmän toiminnot toimivat yhä toivotulla tavalla, eikä järjestelmän ylläpitäjien tarvitse aina käydä kaikkia asioita käsin yksitellen läpi. Automaattitestejä käyttämällä varmistuu myös, että testi tapahtuu aina täysin samalla tavalla. (ServiceNow, 2023I).

Kuva 8 Esimerkki automaattitestin askeleista

Display name	Description	Table	Execution order	Active	Notes
Create a User	Create and impersonate user with first name 'Auto' and last name 'Testaaja' With roles: itil		1	true	
Open a Catalog Item	Open 'Firewall request form' catalog item		2	true	
Set Variable Values	Answer the questions on the form as follows: Caller = Guest and Contact = Testi Testaaja and Short description = testi and Description = testi		3	true	
Order Catalog Item	Order item and confirm that ordering is successful		4	true	
Record Query	Validate there is at least one record in 'sc_request' matching query: Sys ID starts with [[Step 4: Order Catalog Item.Request]] With a failure timeout of 5 Seconds	Request [sc_request]	5	true	
Record Query	Validate there is at least one record in 'sc_task' matching query: Request = [[Step 4: Order Catalog Item.Request]] With a failure timeout of 5 Seconds	Catalog Task [sc_task]	6	true	
Open a New Form	Open a new 'Time Worked' form	Time Worked [task_time_worked]	7	true	
Set Field Values	Set the following values on the form: Comments = test and Type = Asentaja and Date = 30.10.2020 and Task = [[Step 6: Record Query,First record]] and Start = 14:00:00 and End = 15:00:00	Time Worked [task_time_worked]	8	true	
Submit a Form	Submit the current form and confirm form submitted to server		9	true	

5 Tulokset ja jatkokehitys

Työ eteni mielestäni hyvin sujuvasti, eikä työhön tarvinnut tehdä matkan varrella kovin isoja muutoksia. Tämä projekti ei ole ensimmäinen työ, jota olen toimeksiantajalle tehnyt, joten

monet käytännöt ovat jo aikaisemmin vakiintuneet ja sen ansiosta työhön liittyvät vaatimukset olivat jo melko selkeästi listattuna aikaisemmin totuttuun tapaan.

Portaalin etusivusta saatiin mielestäni helppokäyttöinen ja eri aihealueisiin liittyvät lomakkeet ovat kategorisoitu järkevästi ja loogisesti. Lomakkeiden sisältö ja ominaisuudet vastaavat toimeksiantajan toiveita, ja niiden ympärille rakennettu automaatio helpottaa huomattavasti tikettien lähettämistä.

Aiheesta saatiin toteutettua toimiva kokonaisuus ja käyttäjät ovat aloittaneet uuden portaaliratkaisun käytön. Rakentavaa palautetta on jo tullut ja pieniä korjauksia on jo tehty. Portaalia tullaan jatkokehittämään palautteiden perusteella ja lomakkeita voidaan myös lisätä helposti käyttäjien tarpeiden mukaan, ja samoja tekniikoita voidaan hyödyntää tulevaisuudessa pienillä muutoksilla myös ulkoisilla asiakkailta hyödyntäen esim. ServiceNow:n CSM-moduulia.

6 Pohdinta

Työ vaati useiden eri ServiceNow:n osa-alueiden ja työkalujen käyttöä ja oli mielestäni hyvä tapa päästä ServiceNow:n perusasioihin käsiksi. Järjestelmänä ServiceNow on hyvin monipuolinen ja kehittyy jatkuvasti uusien versiopäivitysten myötä. Olen saanut työskennellä järjestelmän parissa toimeksiantajayrityksellä jo muutaman vuoden, ja opin silti joka päivä järjestelmästä jotain uutta. Järjestelmän käyttäjien määrästä päätellen uskon, että ServiceNowiin perehtymällä saa aikaan pitkän uran saman järjestelmän parissa.

ServiceNow on järjestelmänä sen verran laaja, että sitä on yllättävän vaikea kuvata lyhyesti. Aloitin aikanaan järjestelmän parissa työskentelyn täysin järjestelmästä tietämättömänä, kun toimeksiantaja oli ottamassa järjestelmää käyttöön ja järjestelmään on täytynyt perehtyä hiljalleen yksi ominaisuus kerrallaan.

ServiceNow:lla on erittäin kattava dokumentaatio, jota he ylläpitävät itse, ja se helpottaa järjestelmän opiskelua merkittävästi. Tästä konkreettisena esimerkkinä vaikkapa se, että opinnäytetyön lähdeluettelo koostuu lähes täysin ServiceNow:n omasta dokumentaatiosta. Järjestelmä on myös maailmalla hyvin suosittu ja täten sisältää aktiivisen community-foorumin, jossa kehittäjät kyselevät neuvoja toisiltaan ja usein ongelmiin löytää myös hakukoneiden kautta ratkaisun, sillä joku on todennäköisesti törmännyt ennen vastaavanlaiseen ongelmaan ja kysynyt siitä foorumeilla. ServiceNow:lla on myös kattava kurssivalikoima järjestelmän opiskeluun ja kursseilta voi saada järjestelmän ylläpitoon sekä kehittämiseen liittyviä erilaisia sertifikaatteja.

ServiceNow:ssa työskentelyyn ei tarvitse juuri ollenkaan ulkoisia työkaluja, vaan lähes kaikki kehittäminen tapahtuu järjestelmän omassa käyttöliittymässä siellä olevilla työkaluilla. Järjestelmän omat työkalut toimivat mielestäni hyvin ja niillä on mukava tehdä kehitystyötä, ja järjestelmään tulee jatkuvasti uusia työkaluja ja ominaisuuksia, jotka helpottavat työskentelyä.

Lähteet

GetApp. (2023). *IT Support Ticket Types: A Comprehensive Guide*

<https://www.getapp.com/resources/what-are-various-service-desk-ticket-types/>

ServiceNow. (2022). *In the beginning: How ServiceNow was born*

<https://www.servicenow.com/blogs/2022/how-servicenow-was-born.html>

ServiceNow. (2023). *What is ServiceNow?*

<https://www.servicenow.com/what-is-servicenow.html>

ServiceNow. (2022b). *Customer Service Management*

<https://docs.servicenow.com/bundle/tokyo-customer-service-management/page/product/customer-service-management/concept/customer-service-bus-unit-overview.html>

ServiceNow (2023a). *IT Service Management*

https://docs.servicenow.com/bundle/vancouver-it-service-management/page/product/it-service-management/reference/r_ITServiceManagement.html

ServiceNow. (2023b). *HR Service Delivery*

<https://docs.servicenow.com/bundle/vancouver-employee-service-management/page/product/human-resources/concept/hr-service-delivery.html>

ServiceNow. (2023c). *Employee Center*

<https://docs.servicenow.com/bundle/vancouver-employee-service-management/page/product/employee-center/concept/employee-center-landing-page.html>

ServiceNow. (2023d). *Introduction to ServiceNow Scripting*

https://developer.servicenow.com/dev.do#!/learn/learning-plans/vancouver/new_to_servicenow/app_store_learnv2_scripting_vancouver_introduction_to_servicenow_scripting

ServiceNow (2023e). *GlideRecord – Global*

https://docs.servicenow.com/bundle/vancouver-api-reference/page/app-store/dev_portal/API_reference/GlideRecord/concept/c_GlideRecordAPI.html

ServiceNow. (2023f). *Flow Designer*

<https://docs.servicenow.com/bundle/vancouver-build-workflows/page/administer/flow-designer/concept/flow-designer.html>

ServiceNow. (2023g). *Configure portal branding*

https://docs.servicenow.com/bundle/vancouver-platform-user-interface/page/build/service-portal/concept/c_BrandingEditor.html

ServiceNow. (n.d.). *Page Editor*

https://developer.servicenow.com/dev.do#!/learn/learning-plans/vancouver/new_to_servicenow/app_store_learnv2_serviceportal_vancouver_page_editor

ServiceNow. (2023h). *Create and edit a page using the Service Portal Designer*

https://docs.servicenow.com/bundle/vancouver-platform-user-interface/page/build/service-portal/task/t_ConfigureAPage.html

ServiceNow. (n.d.b). *Widget Editor*

https://developer.servicenow.com/dev.do#!/learn/learning-plans/utah/servicenow_application_developer/app_store_learnv2_serviceportal_utah_widget_editor

ServiceNow (2023i). *Impersonate a user*

https://docs.servicenow.com/bundle/vancouver-platform-administration/page/administer/users-and-groups/concept/c_ImpersonateAUser.html

ServiceNow. (2023j). *Update set transfers*

<https://docs.servicenow.com/bundle/vancouver-application-development/page/build/system-update-sets/reference/update-set-transfers.html>

ServiceNow. (2023k). *Back out an update set*

https://docs.servicenow.com/bundle/vancouver-application-development/page/build/system-update-sets/task/t_BackOutUpdateSet.html

ServiceNow. (2023l). *Test your apps with the ATF*

<https://docs.servicenow.com/bundle/vancouver-application-development/page/administer/auto-test-framework/concept/automated-test-framework.html>