



SAVONIA

OPINNÄYTETYÖ - AMMATTIKORKEAKOULUTUTKINTO
TEKNIIKAN JA LIIKENTEEN ALA

LAITEREKISTERIN PERUS- TAMINEN TERVEYDEN- HUOLLON LAITTEILLE JA TARVIKKEILLE

TEKIJÄ: Teemu Turunen

Koulutusala Tekniikan ja liikenteen ala	
Koulutusohjelma/Tutkinto-ohjelma Tietotekniikan koulutusohjelma	
Työn tekijä(t) Teemu Turunen	
Työn nimi Laiterekisterin perustaminen terveydenhuollon laitteille ja tarvikkeille	
Päiväys	Sivumäärä/Liitteet
Ohjaaja(t) Lehtori Keijo Kuosmanen, Lehtori Jussi Koistinen	
Toimeksiantaja/Yhteistyökumppani(t) Suomen aivotutkimus- ja kuntoutuskeskus Neuron	
Tiivistelmä <p>Opinnäytetyön tavoitteena oli kehittää laiterekisteri Suomen aivotutkimus- ja kuntoutuskeskus Neuronille. Laiterekisterin tarkoituksena oli terveydenhuollon laitteiden tietojen sekä huoltotapahtumien tallentaminen. Laiterekisteri kehitettiin SharePoint-alustalle.</p> <p>Kehitysvaihtoehdoksi valittiin app-malli, joka oli uusi keino kehittää SharePointin pilviympäristölle. App-mallista valittiin SharePoint-hosted vaihtoehto, joka on kevyempi tapa kehittää sovelluksia. Sallitut tekniikat ovat HTML, CSS, Javascript ja SharePointin osat.</p> <p>Työn lopputuloksena oli sovellus, joka mahdollisti laitteiden tiedonhallinnan. Sovellus tosin pohjautui paljon SharePointin oletus käyttöliittymälle, jolla sai toteutettua kaiken tarvittavan. Työn loppuvaiheilla tuli esille tarpeita toteuttaa käyttöliittymä toisella tavalla, mutta sille ei ollut enää riittävästi aikaa.</p>	
Avainsanat SharePoint, Laiterekisteri, Add-in, App	

Field of Study Technology, Communication and Transport			
Degree Programme Degree Programme in Information Technology			
Author(s) Teemu Turunen			
Title of Thesis Establishing Device Registry for Medical Devices			
Date	17 December 2019	Pages/Appendices	
Supervisor(s) Mr. Keijo Kuosmanen, Senior Lecturer and Mr. Jussi Koistinen, Senior Lecturer			
Client Organisation /Partners Finnish Center for Brain Research and Rehabilitation Neuron			
<p>Abstract</p> <p>The subject of this thesis was an equipment register for the Finnish Center for Brain Research and Rehabilitation Neuron. The purpose of the equipment register was to store information about health care equipment and their maintenance records. The equipment registry was developed using the SharePoint platform.</p> <p>Different development options were considered. The chosen option was the app model. It was the new way of developing for the SharePoint cloud environment. Out of the two app model options, SharePoint-hosted option was chosen to be used. For its' lighter way to develop for SharePoint. The work started by creating the lists for storing the necessary data. Then the frontpage was made using HTML, which tied together the lists.</p> <p>The result of this thesis was an application which allowed information management of the equipment. However, the application was mostly based on SharePoint's default user interface. At first it accomplished everything that was needed, but later there were some needs to implement the user interface in another way. But in the end, there was not enough time to start developing it in a new way.</p>			
Keywords SharePoint, Equipment register, Add-in, App			

ESIPUHE

Haluan kiittää toimeksiantajaa Neuronian yhteistyöstä opinnäytetyön kanssa. Lisäksi kiitos kaikille Savonia-ammattikoulun lehtoreille opintojeni aikana.

This work is dedicated to Linh.

Kuopiossa 17.12.2019

Teemu Turunen

SISÄLTÖ

1	JOHDANTO	6
1.1	Toimeksiantaja	6
1.2	Lain määritelmä	6
2	SHAREPOINT	7
2.1	Kehitysvaihtoehdot.....	8
2.1.1	Out-of-the-box-ratkaisut.....	8
2.1.2	Kustomisointi	9
2.1.3	Sandboxed Solutions	9
2.1.4	SharePoint Hosted Add-in.....	10
2.1.5	Provider Hosted Add-in.....	10
2.1.6	Farm Solutions.....	10
2.2	SharePoint App-mallin Kehitysympäristö	10
3	TYÖN TOTEUTUS	12
4	YHTEENVETO JA POHDINTA	25
	LÄHTEET	26

1 JOHDANTO

Tämän opinnäytetyön tarkoituksena on toteuttaa laiterekisteri. Laiterekisteri tulee olemaan keskeinen osa terveydenhuollon laitteiden ja tarvikkeiden seurantajärjestelmää, jonka tavoitteena on edistää ja ylläpitää terveydenhuollon laitteiden ja tarvikkeiden turvallisuutta niiden elinkaaren ajan.

1.1 Toimeksiantaja

Työn toimeksiantaja on Suomen aivotutkimus- ja kuntoutuskeskus Neuron. Se on vuodesta 1996 lähtien toiminut neurologisten kuntoutuspalveluiden tuottaja Kuopion Kortejoella. Sen asiakkaat tulevat kautta Suomen ja on mahdollistanut vuosien aikana tuhansien ihmisten kuntoukseen takaisin kotiin, työhön ja harrastuksiin. Neuron on tuonut neurologiseen kuntoutuskäytäntöön useita uusia, asiakaslähtöisiä, toimintakykyä edistäviä kuntoutusmenetelmiä. Nämä ovat myös levinneet käyttöön ympäri Suomen. Vuonna 2018 Vetreä -konserni osti Neuronin kuntoutustoiminnan ja lokakuussa 2019 siirtyi Kuopion Lippumäkeen. (Vetreä 2019)

1.2 Lain määritelmä

Terveydenhuollon laitteiden ja tarvikkeiden seurantajärjestelmä on määritelty laissa: Laki terveydenhuollon laitteista ja tarvikkeista (629/2010). Lain tarkoituksena on edistää ja ylläpitää terveydenhuollon laitteiden ja tarvikkeiden turvallisuutta. (Laki terveydenhuollon...2010, §1, §24-26.)

2 SHAREPOINT

SharePointtia voidaan määritellä jakamalla se kolmeen eri osaan:

- Yhteistyö
- Yhteentoimivuus
- Alusta.

SharePointin yksi päätavoitteista on tuoda ihmiset yhteen, tarjoamalla erilaisia keinoja käyttäjille, jotta he voivat tehdä yhteistyötä. Näitä keinoja ovat muun muassa ryhmäsivustot, sisällön hallinta työkalut, sosiaaliset työkalut, ihmisten ja heidän kykyjen löytäminen sekä paljon muuta. (Fox, Johnson ja Follette 2013, 37.)

SharePoint lisäksi mahdollistaa eri teknologioiden yhteentoimivuuden. Data on integroitavissa eri työkalujen välillä. Microsoftin sisäisesti, ulkopuolinen liiketoiminta data sekä muut verkon teknologiat. Dataa voi laajasti käyttää yhdessä ja niiden pohjalta luoda sovelluksia pilvipalveluihin. Tämä luo paljon mahdollisuuksia valmistaa käyttäjille uusia tehokkaita työkaluja. (Fox ym. 2013, 37.)

Viimeisenä SharePointia voidaan määritellä alustana. Tämä tarkoittaa mahdollisuutta muokata ja laajentaa sen sisältöä perusteellisesti. Tyypillisesti tämä on tarkoittanut SharePointin muokkausta ja ASP.NET tai Javascript kehitystä, mutta pilviteknologiaan siirtyminen on tuonut paljon uusia vaihtoehtoja. 2013 vuoden versio toi mukanaan uuden app-mallin, joka mahdollisti sovellusten liittämisen Sharepointtiin kolmannen osapuolen palvelimelta. (Fox ym. 2013, 37.)

2.1 Kehitysvaihtoehdot

SharePoint tällä hetkellä tarjoaa laajan valikoiman eri kehitysvaihtoehtoja (Taulukko 1). alkaa. Lisäksi sille voi tuottaa täysin uusia ominaisuuksia eri ohjelmointimalleilla. Farm Solutions oli ensimmäinen malli, jolla pystyi monipuolisesti kehittämään On-Premise ympäristössä. Tämän jälkeen siirryttiin malleihin, joiden tarkoitus oli mahdollistaa kehittäminen SharePointin pilviympäristössä.

Taulukko 1 SharePointin kehitysvaihtoehdot

Tekniikka	Ominaisuudet	Kehitys
OOTB Solution	SharePointin oletus ominaisuudet ja työkalut	Suppea
Customizations	Ulkopuolisten sovellusten käyttäminen muokkaamiseen	
Sandboxed Solutions	Osittain tuettu, mutta poistumassa käytöstä. Selaimen puolen koodi	
SharePoint Hosted App	Selaimen puolen koodi ja SharePoint ominaisuudet	
Provider Hosted App	Selain ja palvelin koodi. Koodi ulkopuolisella palvelimella.	
Farm Solutions	Selain ja palvelin koodi. Täydet oikeudet. SharePoint On-Premises	Laaja

2.1.1 Out-of-the-box-ratkaisut

SharePointin mukana tulevat ominaisuudet ja työkalut, jotka eivät vaadi koodia tai kolmannen osapuolen ohjelmia. Nämä usein pystyvät itsessään tarjoamaan ratkaisut organisaation liiketoiminta tarpeille. SharePoint tarjoaa oletuksena laajat työkalut sivustojen hallintaan, dokumenttien organisointiin ja projektien järjestämiselle. Vaikka nämä ovat kuitenkin rajoitetut, niiden käytöstä voi olla huomattava hyöty. (Zelfond 2016-3-12.)

Ominaisuuksien toteuttaminen näillä työkaluilla, tulee monessa tapauksessa halvemmaksi ja nopeammaksi kuin räätälöidyn sovelluksen tuottaminen. Ne vaativat huomattavasti vähemmän työtä sekä erityisosaamisen palkkaamista. Peruskäyttäjät voivat laajalti toteuttaa asioita, pelkästään verkosta löytyvän materiaalin avulla. SharePointin omilla työkaluilla toteutetut ominaisuudet ovat myös suorituskyvyn ja päivittämisen myötä turvallinen valinta. Niiden kanssa voi olla varmempi, että ne eivät aiheuta ongelmia ja tulevat toimivaan uusien päivityksien myötä. (Zelfind 2016-3-12.)

2.1.2 Kustomisointi

SharePointin kustomointi on laaja käsite ja sitä voi tehdä pienessä mittakaavassa tai hyvinkin laajasti. Tämä kustomointi yleensä tehdään ulkopuolisen sovelluksen avulla, jotka Microsoft tarjoaa. Aiemmin käytetyt työkalut olivat SharePoint Designer ja InfoPath, mutta nykyään tuki siirtyy uusille työkaluille; Microsoft PowerApps, Microsoft Flow ja Microsoft Power BI.

Microsoft SharePoint designer on alun perin 2006 vuonna julkaistu ilmaisohjelma kehittämään SharePoint sivustoja. Viimeisin versio on SharePoint Designer 2013, eikä sitä enää päivitä 2016 vuoden jälkeen ja sen tuki lopetetaan lopulta vuonna 2026. Huomattavimmat ominaisuudet ovat datan yhdistäminen SharePointin sisältä ja ulkoa, tiedonsiirron kulku käyttämällä workflow työkalua sekä sivuston ulkonäön muokkaus.

Uusissa työkaluissa huomattavaa on siirtyminen työpöytäsovelluksista verkkoselain pohjaisiin sovelluksiin. Microsoft PowerApps on sovellus, jolla käyttäjä pystyy toteuttamaan liiketoimintasovelluksia ilman ohjelmointi osaamista. Nämä sovellukset ovat selain- tai mobiilipohjaisia ja niitä voidaan liittää SharePoint sivustoon. Microsoft Flow on työkalu, jolla automatisoidaan tiedonsiirto sovellusten välillä. Automatisaatio tapahtuu luomalla workfloweja, jotka voidaan asettaa aktivoitumaan eri ehtojen perusteella. Microsoft Power BI on työkalu, joka käsittelee organisaation dataa. Se on suunniteltu peruskäyttäjille, mahdollistamalla näkymien ja raporttien luomisen ilman erityisosaamista. (Fajdiga 2019-2-11.)

Nämä työkalut yhdessä luovat alustan, jolla voidaan kehittää yksinkertaisesti hyvinkin laajoja liiketoimintasovelluksia. Näitä sovelluksia voidaan suorittaa joko mobiilisti PowerApps-sovelluksella tai selaimella joko Microsoft Dynamics 365 tai SharePoint Online palveluissa. SharePointin kanssa nämä työkalut ovat yhteensopivia SharePoint 2013 ja uudempien versioiden kanssa.

2.1.3 Sandboxed Solutions

Sandboxed Solutions oli uusi kehitysvaihtoehto, joka julkaistiin SharePoint 2010 mukana. Se oli ensiaskel pilvipalveluihin perustuvaan sovelluskehittämiseen, ottamalla käyttöön 2007 versiossa julkaistun Farm Solutions tekniikan. Pilvipalveluun siirtyminen vaati sen, että sovelluksen vapautta piti rajoittaa. Aiemmin käytetty Farm Solutions tekniikka asennettiin koko sivuston laajuudelle ja sille annettiin täydet oikeudet. Microsoft ei voinut antaa täysiä oikeuksia heidän palvelimelleen ladattavalle koodille ja Sandboxed Solutions oli siihen ratkaisu. Koodi sen sijaan ajettiin suojatussa ympäristössä, jossa huonosti toimiva ja resursseja kuluttava tai suorastaan haitallinen koodi ei pysty hidastamaan tai haavoittamaan koko järjestelmän toimintaa. Tämä suojattu ympäristö rajoitti kehitys mahdollisuuksia. (Microsoft 2017-2-2.)

Tällä hetkellä Sandboxed Solutions tuki on keskeytetty ja se on vanhentunutta tekniikkaa. Ainoastaan No-Code versio siitä on tuettu. Se tarkoittaa, että siinä voi olla ainoastaan selaimen puolen koodia, eli Javascriptiä. Lisäksi se voi käyttää SharePointin Site Templates, Lists ja Content Types. No-Code versio ei silti ole suositeltu vaihtoehto ja on parempi käyttää tekniikoita, joita Microsoft tukee tällä hetkellä. (Ferreira 2017-2-19.)

2.1.4 SharePoint Hosted App

SharePoint Hosted App, on SharePoint 2013 version mukana julkaistu kehitysvaihtoehto. Se jatkaa samaa linjaa Sandbox Solution kanssa, siirtymällä pilvipalveluihin ja ajetaan SharePointin yhteydessä, mutta add-in sisältää ainoastaan selaimenpuolen ohjelmointia.

2.1.5 Provider Hosted App

SharePoint 2013 version mukana julkaistu kehitysvaihtoehto. Se on pilvipalveluihin perustuva ratkaisu ja sen on tarkoitus mahdollistaa farm solution tyyppisen sovellus toteutuksen pilviympäristössä. Huomattava ero on se, että koodia ei ajeta SharePointin puolella. Sen sijaan koodi on asennettu kolmannen osapuolen palvelimilla ja siihen otetaan yhteyttä SharePointista. Tämä vaatii authorisoinnin sovelluksen ja sivuston välillä.

2.1.6 Farm Solutions

Farm solution on alkuperäinen tapa toteuttaa sovelluskehitystä SharePoint On-Premise ympäristössä. Se mahdollistaa full stack kehittämisen ja vaatii täydet oikeudet sivustoon. Edelleen suosituin tapa toteuttaa sovelluksia, mutta ei ole mahdollista SharePoint-online ympäristössä.

2.2 SharePoint App-mallin Kehitysympäristö

Appejä kehitetään käyttämällä Visual Studio työkalun 2013 versiota tai uudempaa. Josta löytyy valmiiksi projektipohja App-mallille. Projektissa määritellään, onko kyseessä SharePoint-hosted vai Provider-hosted projekti. Kehitys vaatii lisäksi SharePoint-sivuston, johon otetaan yhteys Visual Studiassa. Sivuston voi toteuttaa pääsääntöisesti kahdella eri tavalla. (Microsoft 2017-3-11.)

Ensimmäinen vaihtoehto on luoda oma paikallinen SharePoint-sivusto. SharePoint 2013 version asennus tukee ainoastaan Windows Server 2008 R2 Service Pack 1 tai Windows Server 2012 käyttöjärjestelmiä. Tämä ympäristö kuvastaa tyypillistä On-Premise SharePoint-sivustoa. Virtuaalikoneiden käyttäminen on yleinen keino helpottaa ympäristön toteutusta sekä ongelmatilanteissa sen palautusta.

Toinen vaihtoehto on käyttää Office 365 kehittäjä sivustoa verkossa. Kehittäjä sivusto on käytettävissä vain tietyissä Office 365 -palvelupaketeissa, mutta sille on myös kokeiluversio. App-malli suunniteltiin käytettäväksi SharePointin Online-versiossa. Tästä johtuen sen kehitys onnistuu ilman laajempaa On-Premise ympäristöä.

3 TYÖN TOTEUTUS

Työn tekeminen alkoi toukokuussa 2015. Sen sisällöstä, toteutustavasta ja aikataulusta sovittiin aloituspalaverissa, Neuronin tiloissa Kortejoella. Työ oli tarkoitus toteuttaa käyttämällä SharePointin app-tekniikkaa kolmessa kuukaudessa kesän aikana. Työn teko tapahtui itsenäisesti kotona.

Aloituksen jälkeen käytiin vielä palaverissa, keskustelemassa laiterekisterin sisällöstä tarkemmin. Lisäksi työn edetessä asiakkaalle lähetettiin app-paketti, jonka sai helposti asennettua asiakkaan SharePoint ympäristöön. Asiakkaat pystyivät kokeilemaan sovellusta ja tarvittavista muutoksista käytiin työn aikana vielä useammassa palaverissa.

Työn aikana kohdattiin muutamia ongelmia sovelluksen toiminnallisuuksien kanssa. Tämän johdosta perehdyttiin myös muihin kehitysvaihtoehtoihin, jotka voisivat auttaa sovelluksen toteuttamisen kanssa tai kartoittaa toisenlaista lähestymistapaa työn toteuttamisen kannalta.

3.1 Tietokanta

Vaikka SharePointin pohjalla on SQL Server, joka ylläpitää sivuston tiedonhallinnan rakennetta, käyttäjät kuitenkin hallitsevat dataa listoilla. Lista voi aluksi vaikuttaa vain perinteiseltä taulukkolaskelmaohjelmalta, mutta sen rooli SharePoint sivustossa on paljon laajempi ja se sisältää paljon toiminnallisuutta. Kalenterit sekä dokumenttikirjastot ovat esimerkiksi pohjimmiltaan listoja.

SharePoint listoja luodessa, niiden sisältö tulee rakentaa käyttämällä site column ja content type työkaluja. Site column määrittelee listan sarakkeen sisällön ja sen metadatan. Content type on taas koelma site column -sarakkeita. Site Column ja content type ovat molemmat koko sivuston laajuisia, noudattaen kuitenkin periytymisperiaatetta. Tämä tarkoittaa, että niitä voidaan käyttää sen sivuston alisivustoissa, jossa ne on otettu käyttöön. Ne eivät kuitenkaan näy ylemmillä sivustoilla tai toisissa sivustohaaroissa. Näiden työkalujen käyttäminen oikein, helpottaa SharePoint-sivuston datan yhteisyyden säilyttämisen, sekä vähentää toistuvan työn tekemistä.

3.2 Listan luominen Visual Studiassa

Listaa aloitetaan tekemällä site column, johon lista tulee perustumaan. Site column on xml tiedosto, jossa sarakkeet ilmaistaan Field elementeillä. Kuvassa nimetään huoltolistan laitenumero ja laitenimi, jotka toteutetaan lookup tyyppisinä. Lisäksi määrää aikaishuolto sarake, joka on RadioButton tyyppinen (Kuva 1).

```
<Elements xmlns="http://schemas.microsoft.com/sharepoint/">
  <Field Type="Lookup"
    Name="LaitenumeroLookup"
    ID="{2A688022-84AF-4C51-84F7-8E1B6025F104}"
    List="Lists/Laitelista"
    ShowField="laitenumero"
    DisplayName="Laitenumero"
    Required="true"
    Group="Custom Site Columns"/>
  <Field Type="Lookup"
    FieldRef="2A688022-84AF-4C51-84F7-8E1B6025F104"
    Name="laitenimiLookup"
    ID="{F0AF8CA5-EA71-46F7-9536-036539BA8C5C}"
    List="Lists/Laitelista"
    ShowField="laitenimi"
    DisplayName="Laitenimi"
    Required="false"
    Group="Custom Site Columns"
    ReadOnly="TRUE" />
  <Field
    ID="{539C0D45-5F45-43F4-B683-4B7655E92181}"
    Name="huoltotyyppi"
    DisplayName="Määräaikaishuolto"
    Type="Choice"
    Format="RadioButtons"
    Required="FALSE"
    Group="Custom Site Columns">
    <Default>Ei</Default>
    <CHOICES>
      <CHOICE>Ei</CHOICE>
      <CHOICE>Kyllä</CHOICE>
    </CHOICES>
  </Field>
```

Kuva 1 Site Column XML koodi

Seuraava luodaan content type, joka on kokoelma sarakkeita. Siinä voidaan käyttää sarakkeita useammasta eri site columnista tai SharePointin tarjoamista oletus sarakkeista. Content Type-välilehdessä voi asettaa vielä muutamia metadata asetuksia (Kuva 2).

Huoltolistassa (Kuva 3) tallennetaan laitteille uusia huoltotoimenpiteitä. Huoltotoimenpiteiden kirjaaminen on yksi keskeinen osa sovellusta. Tässä listassa käytetään muokattua näkymää, normaalin sijaan. Data ei ole riveittäin, huoltotapahtuma kerrallaan. Sen sijaan data näytetään ryhmittelemällä.

	Laitenimi	Määräaikaishuolto	Toimenpide	Päivämäärä	Huollon tekijä
Laitenumero : 1 (1)					
Laitenumero : 2 (1)					
Laitenumero : 3 (1)					
Laitenumero : 4 (2)					
Laitenumero : 5 (1)					
Laitenumero : 6 (3)	Testiläite	Ei	Lorem ipsum dolor sit amet, consectetur adipiscing elit. Vestibulum rhoncus quis magna sit amet blandit. Maecenas sit amet justo mollis, elementum nisl tincidunt, pharetra quam. Ut id faucibus massa, non consequat nisi. Pellentesque nec malesuada nunc, non commodo turpis.	23.8.2016	TT
	Testiläite	Ei	Donec nec libero nec felis varius interdum. Donec et nulla a mi volutpat venenatis. Vestibulum iaculis odio a laoreet tincidunt. Aliquam erat volutpat. Duis dignissim neque odio, eget vehicula nisl tincidunt nec. Ut tortor lectus, pellentesque nec mollis vitae, tristique rhoncus justo.	18.5.2016	TP
	Testiläite	Kyllä	Sed dignissim nunc sed est hendrerit varius. Nullam egestas quam sed dolor tempor, venenatis posuere tortor mattis. Nunc lorem ex, porttitor eu enim sit amet, vehicula pharetra elit. Praesent pulvinar massa nisl, id consectetur eros molestie ac. Curabitur sed sagittis diam, eget consectetur augue.	17.5.2016	NT
Laitenumero : 7 (1)					
Laitenumero : 111 (1)					
Laitenumero : 333 (2)					

Kuva 3 Huoltolista

Huoltolistan sisältö on seuraava:

Laitenumero (Lookup): Laitteen numero, haetaan laitelistasta.

Laitenimi (Lookup): Laitteen nimi, haetaan laitelistasta.

Määräaikaishuolto (Choice): Määritetään onko kyseessä määräaikaishuolto.

Toimenpide (Multiple Lines of Text): Laitteelle suoritetun huoltotoimenpiteen kuvaus.

Päivämäärä (Date and Time): Suoritetun huollon päivämäärä.

Huollon tekijä (Single Line of Text): Suorittaneen huollon tekijän nimi.

Numero	Laitteen nimi	Merkki ja malli	Sarjanumero	Laiteryhmä	Yksikkö	Tila	Kuva	Hankintapaikka	Hankinta pvm.	Takuu	Käyttöohje	Lisätiedot	Huoltoväli	Määräaikaishuolto
1	Testilaitte	Testi merkki ja malli	Testi sarjanumero	Testiryhmä 1	Kuntoutusosasto	Testi tila	https://www.google.fi		1/1/2015	Testi takuu	Testi ohje	Lisätieto	30	2/2/2016
2	Testilaitte	Testi merkki ja malli	Testi sarjanumero	Testiryhmä 1	Kuntoutusosasto	Testi tila	https://www.google.fi		1/1/2015	Testi takuu	Testi ohje	Lisätieto	30	2/13/2016
3	Testilaitte	Testi merkki ja malli	Testi sarjanumero	Testiryhmä 1	Kuntoutusosasto	Testi tila	https://www.google.fi		1/1/2016	Testi takuu	Testi ohje	Lisätieto	30	1/22/2016
4	Testilaitte	Testi merkki ja malli	Testi sarjanumero	Testiryhmä 1	Palvelukoti	Testi tila	https://www.google.fi		1/2/2016	Testi takuu	Testi ohje	Lisätieto	30	2/22/2016
5	Testilaitte	Testi merkki ja malli	Testi sarjanumero	Testiryhmä 2	Palvelukoti	Testi tila	https://www.google.fi		1/1/2016	Testi takuu	Testi ohje	Lisätieto	30	2/22/2016
6	Testilaitte	Testi merkki ja malli	Testi sarjanumero	Testiryhmä 1	Kuntoutusosasto	Testi tila	https://www.google.fi		1/1/2016	Testi takuu	Testi ohje	Lisätieto	30	2/22/2016
7	Testilaitte	Testi merkki ja malli	Testi sarjanumero	Testiryhmä 1	Palvelukoti	Testi tila	https://www.google.fi		1/1/2016	Testi takuu	Testi ohje	Lisätieto	30	2/22/2016
8	Testilaitte	Testi merkki ja malli	Testi sarjanumero	Testiryhmä 1	Terapiatoiminta	Testi tila	https://www.google.fi		1/1/2016	Testi takuu	Testi ohje	Lisätieto	30	2/22/2016
9	Testilaitte	Testi merkki ja malli	Testi sarjanumero	Testiryhmä 1	Avoterapia	Testi tila	https://www.google.fi		1/1/2016	Testi takuu	Testi ohje	Lisätieto	30	2/22/2016
10	Testilaitte	Testi merkki ja malli	Testi sarjanumero	Testiryhmä 1	Ravintokeskus	Testi tila	https://www.google.fi		1/1/2016	Testi takuu	Testi ohje	Lisätieto	30	2/22/2016
11	Lisää laitteita	Testi merkki ja malli	Testi sarjanumero	Testiryhmä 1	Avoterapia	Testi tila	https://www.google.fi		1/1/2016	Testi takuu	Testi ohje	Lisätieto	30	1/11/2016
12	Lisää laitteita	Testi merkki ja malli	Testi sarjanumero	Testiryhmä 1	Avoterapia	Testi tila	https://www.google.fi		1/1/2016	Testi takuu	Testi ohje	Lisätieto	30	1/11/2016
13	Lisää laitteita	Testi merkki ja malli	Testi sarjanumero	Testiryhmä 1	Palvelukoti	Testi tila	https://www.google.fi		1/1/2016	Testi takuu	Testi ohje	Lisätieto	30	1/11/2016
14	Lisää laitteita	Testi merkki ja malli	Testi sarjanumero	Testiryhmä 1	Terapiatoiminta	Testi tila	https://www.google.fi		1/1/2016	Testi takuu	Testi ohje	Lisätieto	30	1/11/2016
15	Lisää laitteita	Testi merkki ja malli	Testi sarjanumero	Testiryhmä 1	Terapiatoiminta	Testi tila	https://www.google.fi		1/1/2016	Testi takuu	Testi ohje	Lisätieto	30	1/11/2016

Kuva 4 Laitelista

Laitelistan sisältö on seuraava:

Numero (Number): Laitteelle annettava numero.

Laitteen nimi (Single Line of Text): Laitteen nimi.

Merkki ja malli (Single Line of Text): Laitteen merkki ja malli.

Sarjanumero (Single Line of Text): Laitteen sarjanumero

Laiteryhmä (Lookup): Laitteen ryhmä, johon se kuuluu. Vaihtoehdot haetaan laiteryhmälistasta.

Yksikkö (Lookup): Yksikkö johon laite kuuluu. Vaihtoehdot haetaan yksikkölistasta.

Tila (Single Line of Text): Tila, jossa laitetta käytetään.

Kuva (Single Line of Text): Mahdollisuus lisätä hyperlinkki laitteen kuvan sijaintiin.

Esimerkiksi SharePointin kuvakirjastoon voidaan lisätä kuva laitteesta ja liittää kuvan hyperlinkki listaan.

Hankintapaikka (Single Line of Text): Paikka, josta laite on hankittu.

Hankinta pvm. (Date and Time): Laitteen hankinta päivämäärä.

Takuu (Single Line of Text): Tiedot laitteeseen liittyvästä takuusta.

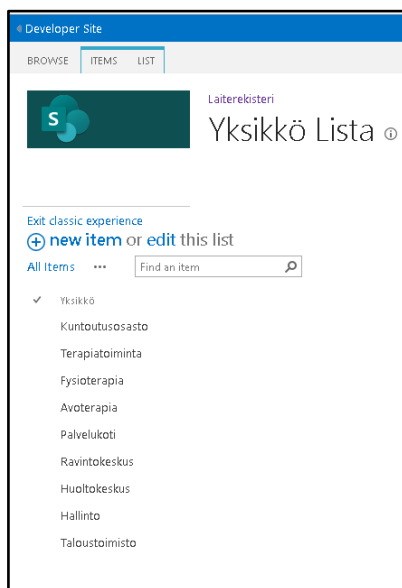
Käyttöohje (Single Line of Text): Mahdollisuus lisätä hyperlinkki laitteen käyttöohjeeseen.

Lisätiedot (Single Line of Text): Yleisiä lisätietoja laitteesta.

Huoltoväli (Single Line of Text): Laitteelle tehtävän määräaikaishuollon huoltoväli.

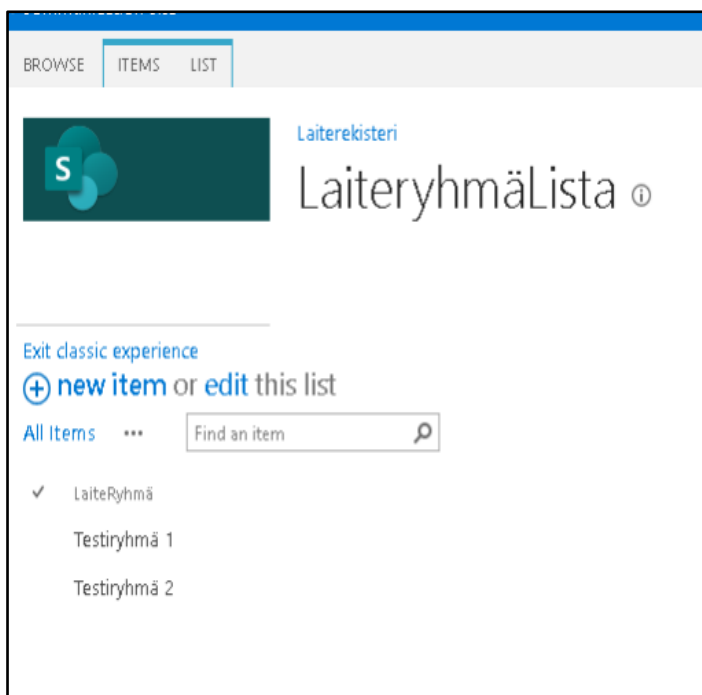
Määräaikaishuolto (Date and Time): Laitteen viimeisin määräaikaishuolto.

Laitelista hakee tästä listasta (Kuva 5) sen vaihtoehdot yksikkösarakeeseen. Tämän näkymän kautta pystytään lisäämään, muokkaamaan tai poistamaan yksiköitä tarpeen mukaan.



Kuva 5 Yksikkö lista

Laiteryhmälista toimii samalla tavalla kuin yksikkölista ja laitelista hakee siitä sen vaihtoehdot laiteryhmä-sarakeeseen. Listan (Kuva 6) sisältöä voi muokata tarpeen mukaan.

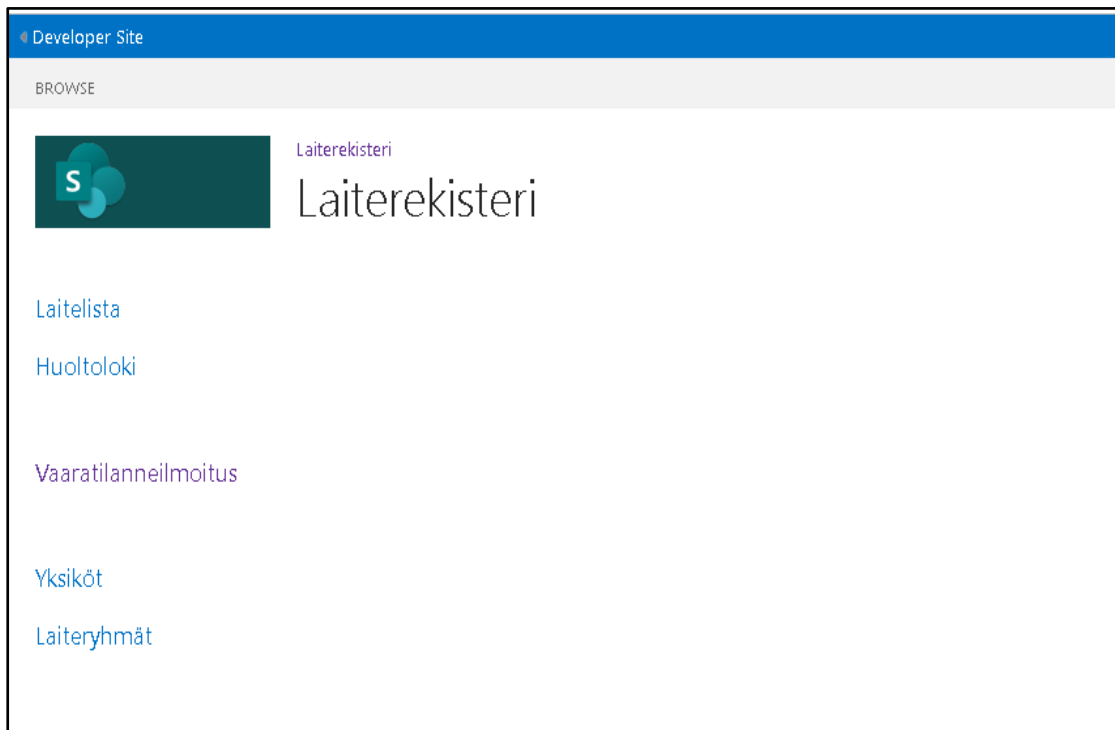


Kuva 6 Laiteryhmä lista

3.3 Etusivu

Listojen näkymää voidaan muokata sovelluksen asennettua SharePoint-sivustoon.

Laiterekisterin etusivulla näytetään linkit sovelluksen sisältöön (Kuva 7). Sivulla on linkit sovelluksen listoihin sekä linkki valviran vaaratilanneilmoitus kaavakkeeseen.



Kuva 7 Sovelluksen etusivu

Sivu on rakenteeltaan .ASPX tyyppinen ja käyttää HTML sekä ASP.NET Web Forms syntaxia. Kuvan (Kuva 8) koodissa luodaan linkit laite ja huolto listoihin, lisäksi linkki vaaratilanneilmoitukseen.

```
<div>
  <p style="font-size:130%"><asp:HyperLink runat="server"
NavigateUrl="JavaScript:window.location = _spPageContextInfo.webAbsoluteUrl + '/Lists/HuoltoLista/AllItems.aspx';"
Text="Huoltoloki" /></p>
</div>
<br />
<div>
  <p style="font-size:130%"><asp:HyperLink runat="server"
  NavigateUrl="https://tlt.valvira.fi/tltvaara/index.html"
  Text="Vaaratilanneilmoitus" /></p>
</div>
<div>
  <br />
  <p style="font-size:130%"><asp:HyperLink runat="server"
NavigateUrl="JavaScript:window.location = _spPageContextInfo.webAbsoluteUrl + '/Lists/YksikkoLista/AllItems.aspx';"
Text="Yksiköt" /></p>
</div>
```

Kuva 8 Etusivun koodia

Etusivun vaaratilanneilmoitus-hyperlinkki vie Valviran sähköiseen käyttäjän vaaratilanneilmoitus lomakkeeseen (Kuva 9.) Terveystieteiden ja tarvikkeiden ammattimaisilla käyttäjillä on ilmoittamisvelvollisuus niiden aiheuttamista vaaratilanteista.

← → ↻ 🔒 <https://tit.valvira.fi/titvaara/index.html>

Käyttäjän vaaratilanneilmoitus

Valvira
 Sosiaali- ja terveysalan lupa- ja valvontavirasto
 Lääkinnälliset laitteet
 Mannerheimintie 103b PL 210 00281 Helsinki

Tähdellä (*) merkityt kentät ovat pakollisia.

A. ILMOITTAJAN TIEDOT

Laitoksen nimi (*)	<input type="text"/>
Osoite (*)	<input type="text"/>
Postinumero (*)	<input type="text"/>
Postiosoite (*)	<input type="text"/>
Puhelinnumero (*)	<input type="text"/>
Faxnumero	<input type="text"/>
Yhteyshenkilö (*)	<input type="text"/>
Asema	<input type="text"/>
Ilmoituksen tekijän nimi (Mikäli eri kuin yhteyshenkilö)	<input type="text"/>
Sähköposti	<input type="text"/>
Puh	<input type="text"/>

B. LAITTEEN/TARVIKKEEN TIEDOT

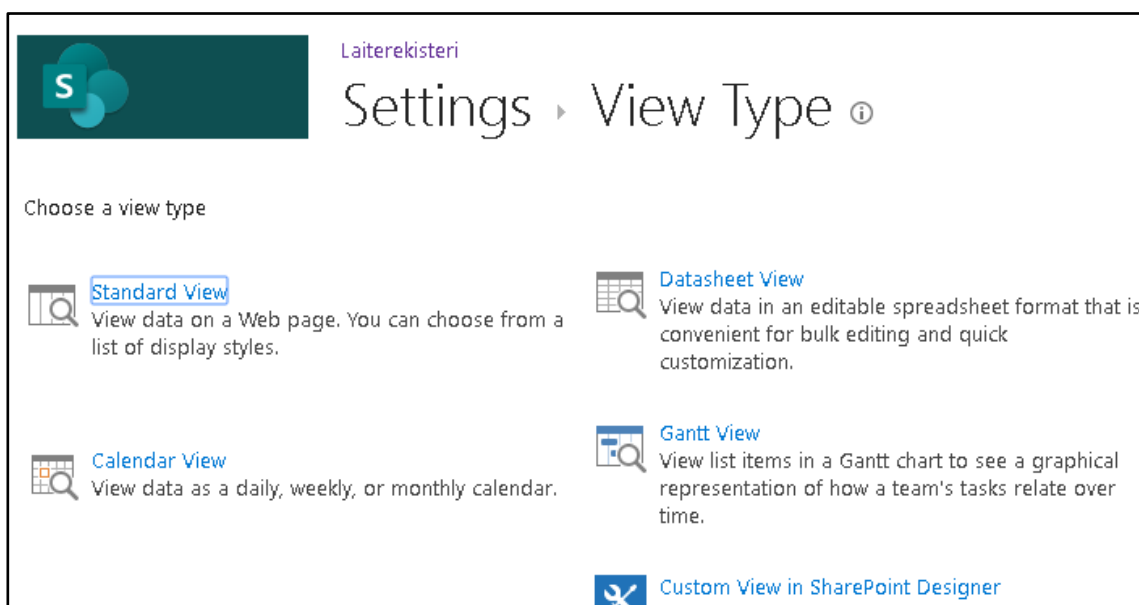
Valmistaja (*)	<input type="text"/>
----------------	----------------------

Kuva 9 Valviran vaaratilanneilmoitus-lomake

3.4 Listojen käyttöliittymä

SharePointissa on valmiiksi tarjolla käyttöliittymä listoille, joka ulkonäöltään muistuttaa perinteistä taulukkolaskentaohjelmaa. Tämä käyttöliittymä mahdollistaa kaikki tyypilliset toiminnallisuudet tietokannalle. Sisältöä voi lisätä, lukea, muokata ja poistaa. Tästä johtuen laiterekisterin toteuttaminen oli suhteellisen yksinkertainen tehtävä.

Sovelluksen sisältöä kehitettiin käymällä asiakkaan kanssa toistuvasti palavereissa, joissa sovellusta testattiin ja annettiin palautetta. Käyttöliittymä oli sopiva asiakkaan tarkoitukseen, mutta sille tuli työn edetessä muokkaus ehdotuksia. Tämän johdosta perehdyttiin tarkemmin siihen, mitä käyttöliittymällä pystyi tehdä ja kuinka sitä voi muokata.



Kuva 10 Listan näkymän luominen

SharePointissa on näkymä ominaisuus listoille (Kuva 10), joka mahdollistaa jonkin verran käyttöliittymän muokkausta. Listassa näytettävää tietoa voidaan rajoittaa ja järjestää uudelleen. Lisäksi listan voi esittää eri muodossa. Itse App-mallissa, käyttämällä Visual Studiota, on muutamia keinoja vaikuttaa listan toiminnallisuuteen. Site columnin XML tiedostoa voidaan muokata tai listalle voi lisätä CSS tyylittely tiedoston. Nämä keinot olivat kuitenkin rajalliset ja työn edetessä alkoi kertyä ominaisuuksia sekä muutoksia, joita ei pystytty tekemään. Tästä johtuen selvitettiin myös toissijaista vaihtoehtoa käyttöliittymän toteutukselle.

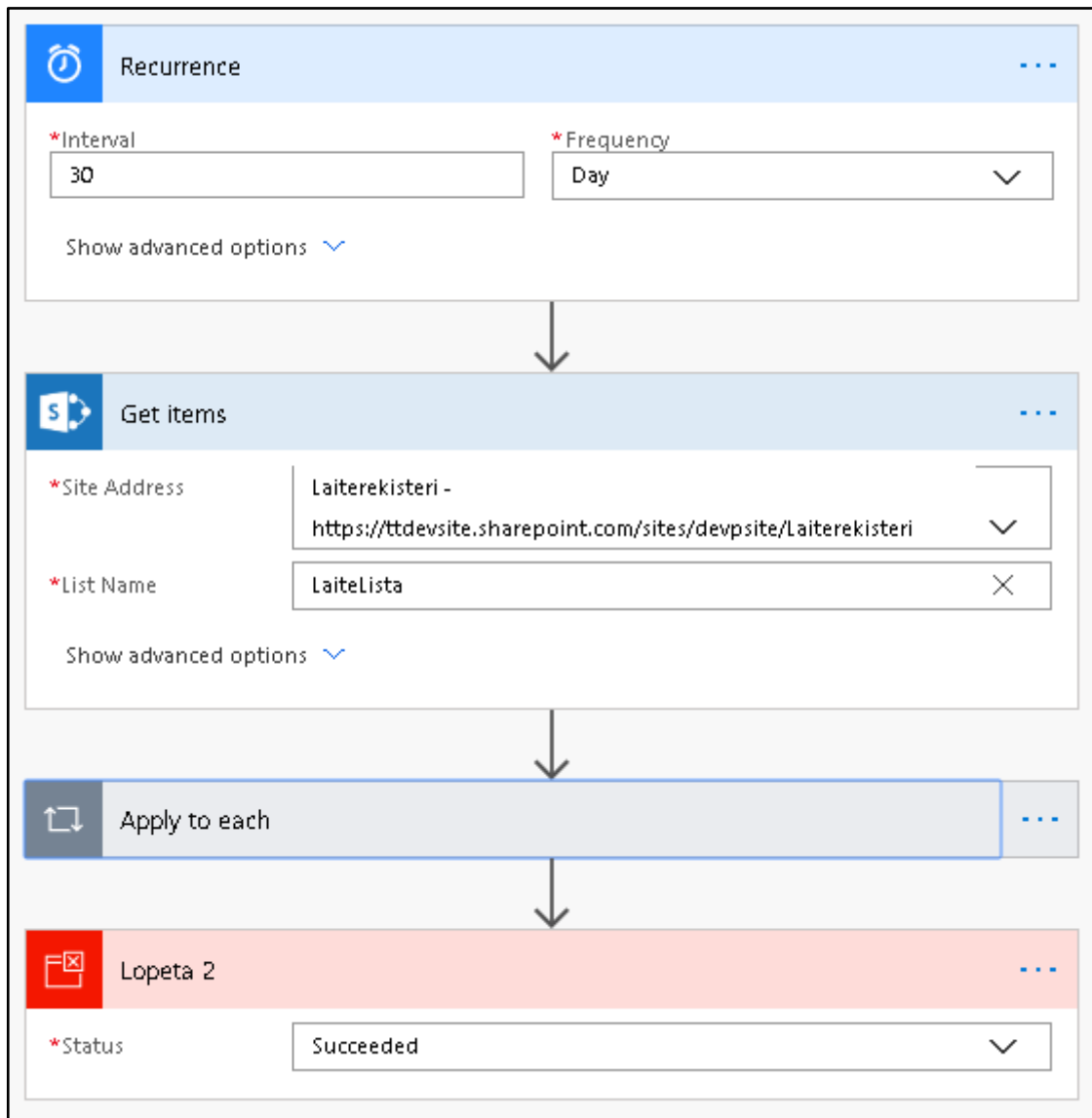
3.5 Huoltomuistutus

Yhtenä työn vaatimuksena oli automaattiset muistutukset laitteiden määräaikaishuolloille. Tämä ominaisuus ei perustu lainsäädäntöön. Sen tarkoitus on pelkästään helpottaa huoltoryhmän työtä ylläpitää laitteiden määräaikaishuoltoja. SharePoint-ympäristössä tällaisen ominaisuuden saa toteutettua käyttämällä työnkulkuja.

Työnkulut ovat pienoishjelmia, joilla voi automatisoida yleisimpiä työtehtäviä. App-mallin projektiin on mahdollista liittää ja muokata työnkulkuja. Työkalut niiden tekemiselle ovat suhteellisen laajat Visual Studiossa, mutta niiden käynnistys vaihtoehdot ovat hieman rajalliset. Työnkulku on joko sivuston laajuinen tai liitetty tiettyyn listaan. Listaan liitetty työnkulku käynnistyy, kun valitun listan sisältöä muokataan tai siihen lisätään uutta. Muuten työnkulut täytyy käynnistää manuaalisesti käyttäjän toimesta. Huoltomuistutusten oli tarkoitus olla automaattisia, joten tämä vaihtoehto ei sopinut tilanteeseen. Lopulta kyseisen ominaisuuden tekeminen jäi syrjään.

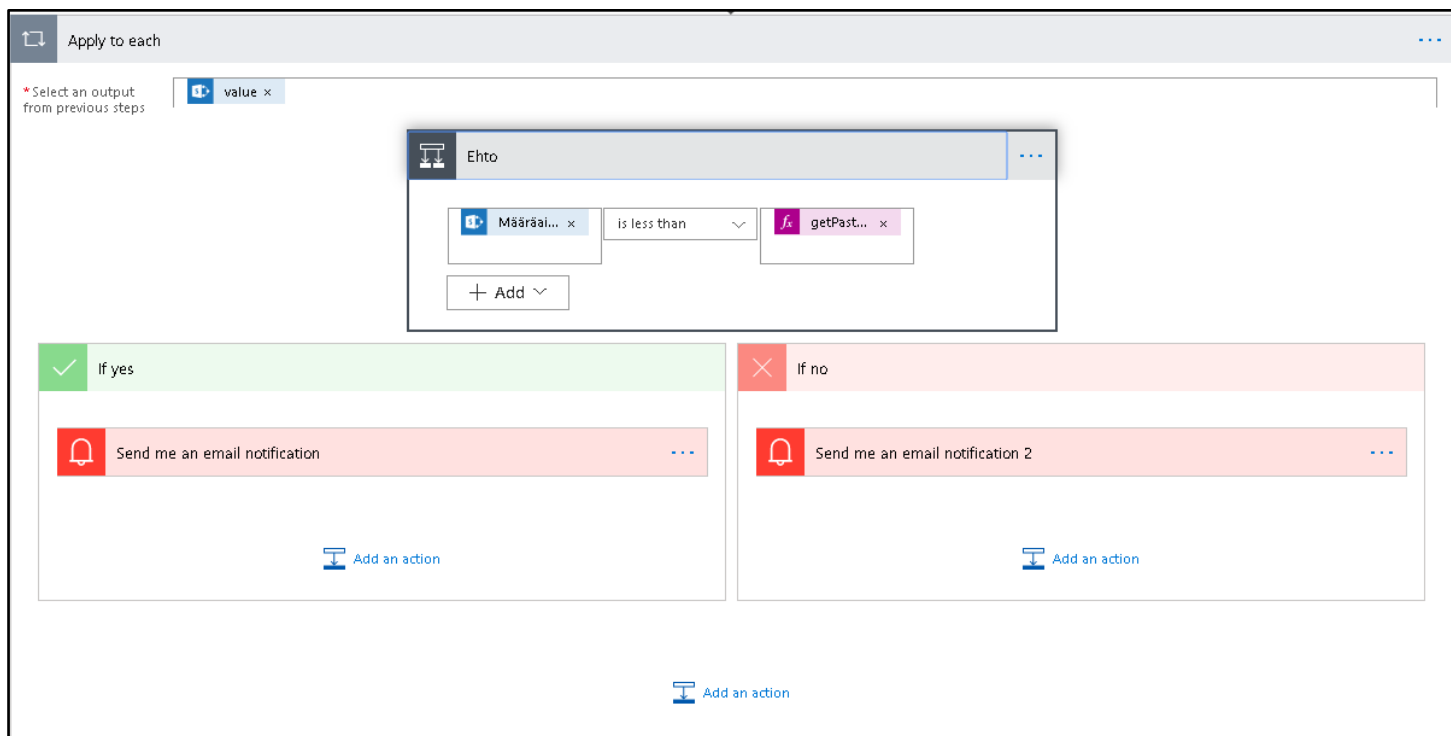
Nykyään kyseisen ominaisuuden saa helpoiten toteutettua käyttämällä uutta Microsoft Flow työkalua. Siihen voi siirtyä suoraan SharePoint-sivustosta, käyttäen samaa käyttäjätunnusta. Sivuston sisältö on saman tien käytettävissä työnkulkuja luodessa ja se toimii app-mallilla luotujen sovellusten kanssa.

Työnkulun tekeminen Flow:ssa alkaa luomalla ehto, josta sen suorittamisen käynnistyy. Tässä tapauksessa ehtona on ajoitettu toisto kolmenkymmenpäivän välein. Seuraavana annetaan ensimmäinen toiminto Get Items, joka tässä tapauksessa hakee kaikki laitteet laitelistasta. Tiedon käytön helppous näkyy kuvassa (Kuva 11). Tarvitaan vain linkki sivustoon ja listan nimi.



Kuva 11 Työnkulku

Kuvassa (Kuva 122) nähdään, kuinka haettua tietoa käytetään. Apple to each-vaiheessa käydään läpi kaikki listan laitteet ja tarkistetaan, onko tarvetta lähettää huoltomuistutusta. Ehto-vaiheessa määritellään, minkä perusteella huoltomuistutus lähetetään. Flow mahdollistaa ehdon luomisessa yhdistelmän dynaamista sisältöä ja funktioita. Lopuksi suoritetaan toiminto, jos ehto on täyttynyt. Tässä esimerkissä lähetetään sähköposti muistutus, mutta mahdollista olisi myös luoda huollosta tehtävä SharePoint-sivustoon.



Kuva 12 Työnkulun ehto

3.6 JSOM käyttäminen listan kanssa

Vaikka listojen oletus käyttöliittymä oli tarpeeksi monipuolinen ja sillä sai tehtyä kaikki tarvittavat asiat, tuli lopulta tarvetta pienille muutoksille. Tämä oli ongelmallista, koska sitä ei käytännössä voinut muokata ollenkaan. Koodi, joka määrittelee käyttöliittymän ulkonäön ja sen toiminnallisuuden, on SharePointin palvelimilla. App-mallissa sitä ei pysty muokkaamaan.

Sovellukseen voi kuitenkin rakentaa listan käyttämiseksi oman käyttöliittymän. Sen luomisessa voi käyttää selaimen puolen koodia, jota on HTML, CSS ja Javascript. Data listasta saadaan käyttämällä SharePointin Javascript Object Modelia. Se on rajapinta, jolla listalle voidaan suorittaa kaikkia yleisimpiä toimintoja. Tämä mahdollistaa kustomoidun käyttöliittymän kehittämisen, jossa listaa käytetään taustalla vain tietokantana.

```

function retrieveListItems() {
    var clientContext = new SP.ClientContext(siteUrl);
    var oList = clientContext.get_web().get_lists().getByTitle('LaitteetList');

    var camlQuery = new SP.CamlQuery();
    camlQuery.set_viewXml(
        '<View><Query><Where><Eq>' +
        '<FieldRef Name=\'Numero\'/>' +
        '</Eq></Where></Query></View>');
    this.collListItem = oList.getItems(camlQuery);
    clientContext.load(collListItem);

    clientContext.executeQueryAsync(Function.createDelegate(this, this.onQuerySucceeded),
    );
}

function onQuerySucceeded(sender, args) {
    var html = '<table>';
    var listItemEnumerator = collListItem.getEnumerator();

    while (listItemEnumerator.moveNext()) {
        var oListItem = listItemEnumerator.get_current();
        html += '<tr><td>';
        html += oListItem.get_item('Numero');
        html += '<tr><td>';
        html += oListItem.get_item('Laitteen nimi');
        html += '</tr></td>';
    }
    html += "</table>";
}

```

Kuva 13 Listan lukeminen JSOM:lla

Kuvan (Kuva 133) Javascript funktio hakee listan sisällön SharePoint-sivustosta. SP.ClientContext() on objecti, joka mahdollistaa yhteyden sivustoon. Seuraavaksi määritellään hakukysely, mitä tietoa listasta haetaan. Haku suoritetaan ja siirrytään onQuerySucceeded funktioon, jossa listan sisältö voidaan esittää käyttämällä HTML kieltä.

4 YHTEENVETO JA POHDINTA

Opinnäytetyön tavoitteena oli toteuttaa laiterekisteri SharePoint-ympäristössä. Laiterekisterin tarve perustui lainsäädäntöön terveydenhuollon laitteiden seurannalle. Työn keskeiset vaatimukset olivat laitteiden tietojen ja huoltotapahtumien tallennus sovellukseen sekä määräaikaishuollon muistutukset. Toimeksiantaja toimi Suomen aivotutkimus- ja kuntoutuskeskus Neuron.

Lopputuloksena saatiin aikaan sovellus, johon pystyi tallentamaan laitteiden tiedot sekä huoltotapahtumat. Listojen oletus käyttöliittymä teki tämän helpoksi ja siihen keskityttiin hieman liika työn aikana. Tämä johti siihen, että sovellusta ei pystynyt räätälöimään juuri asiakkaalle sopivaksi. Työn lopussa kuitenkin perehdyttiin, kuinka tiettyjä ominaisuuksia olisi voinut toteuttaa paremmin. Microsoft Flown käyttö työkuluissa ja erityisesti Javascript object modelin käyttö, mikä olisi laajentanut kehitysmahdollisuuksia huomattavasti.

Ongelmia työn kanssa, olisi voinut välttää perehtymällä valittuun kehitysvaihtoehtoon sekä muihin kehitysvaihtoehtoihin tarkemmin. Työn alussa ei ollut selvää näkemystä siitä, mitä SharePoint-hosted app-mallilla pystyi saamaan aikaan. Tästä johtuen kului paljon aikaa hukkaan yrittäen toteuttaa asioita, jotka olivat lopulta mahdottomia. Vähäinen tietämys SharePoint kehittämisestä teki myös tiedon hakemisesta paljon vaikeampaa.

LÄHTEET

LAKI TERVEYDENHUOLLON LAITTEISTA JA TARVIKKEISTA. L 2010/629. Finlex. Lainsäädäntö. [Viitattu 2019-12-14.] Saatavissa: <https://www.finlex.fi/fi/laki/alkup/2010/20100629#Lidp448107856>

FAJDIGA, Renato 2019-2-11. Optimizing business with Microsoft power platform – Power Apps, Microsoft Flow and Power BI explained. [verkkoaineisto]. [viitattu 2019-12-14.] Saatavissa: <https://www.adacta-group.com/news-center/optimizing-business-with-microsoft-power-platform-power-apps-microsoft-flow-and-power-bi-explained>

FERREIRA, Joao 2017-2-19. Can you still use sandboxed solution in 2017? [verkkoaineisto]. [viitattu 2019-12-14.] Saatavissa: <https://sharepoint.handsontek.net/2017/02/19/can-you-still-use-sandboxed-solutions-in-2017/>

FOX, Steve, JOHNSON, Chris ja FOLLETTE, Donovan 2013. Beginning SharePoint 2013 development. 1. painos. Indianapolis, Wiley.

MICROSOFT, 2017-2-2. Sandboxed solution considerations. [verkkoaineisto]. [viitattu 2019-12-14.] Saatavissa: <https://docs.microsoft.com/en-us/visualstudio/sharepoint/sandboxed-solution-considerations?view=vs-2019>

MICROSOFT, 2017-3-11. Tools and environments for developing SharePoint Add-ins. [verkkoaineisto]. [viitattu 2019-12-14.] Saatavissa: <https://docs.microsoft.com/en-us/sharepoint/dev/sp-add-ins/tools-and-environments-for-developing-sharepoint-add-ins>

VETREÄ, 2019. VetreäNeuron, Kuopio [verkkoaineisto]. [viitattu 2019-12-14.] Saatavissa: <https://www.vetrea.fi/yksikot/vetreaneuron-kuopio/>

ZELFOND, Gregory 2016-3-12. 6 reasons to use SharePoint Out of the Box functionality. [verkkoaineisto]. [viitattu 2019-12-14.] Saatavissa: <https://sharepointmaven.com/six-reasons-use-sharepoint-out-of-the-box/>