

Jarkko Tornberg

Intranetpalvelun personointi

Metropolia Ammattikorkeakoulu

Insinööri (AMK)

Mediatekniikan koulutusohjelma

Insinööriytyö

15.11.2017

Tekijä Otsikko Sivumäärä Aika	Jarkko Tornberg Intranetpalvelun personointi 34 sivua 15.11.2017
Tutkinto	Insinööri (AMK)
Koulutusohjelma	Mediatekniikka
Ohjaajat	Vanhempi kouluttaja Hannu Kauremaa Yliopettaja Kari Aaltonen
<p>Insinööriyön tarkoituksena oli tutkia Microsoft SharePoint Framework -sovelluskehitysmenetelmän soveltumista SharePoint-pohjaisen intranetpalvelun personointiin. Työssä kartoitettiin Office 365 -palvelussa syntyvän käyttäjäkohtaisen sisällön hyödyntämistä intranetin personoinnissa. Office 365 -palvelussa syntyvään käyttäjäkohtaiseen dataan on mahdollista päästä käsiksi käyttämällä Microsoftin tarjoamia rajapintoja. Työssä tutkittiin tämän mahdollisuuden hyödyntämissä SharePoint Framework -sovelluksissa.</p> <p>SharePoint Framework -sovellus on ns. asiakaspuolen sovellus, joka suoritetaan SharePoint-sivustolla omassa verkko-osassaan. Sovellus voi sisältää HTML-, CSS- ja JavaScript-koodia, joka suoritetaan käyttäjän selaimessa, käyttäjän oikeuksilla. SharePoint-sivulle voidaan sijoittaa kerrallaan useita SharePoint Framework -sovelluksia. Pienet, uudelleen käytettävät SharePoint Framework -komponentit mahdollistavat SharePoint Online -pohjaisen intranetpalvelun toiminnallisuuksien helpon laajentamisen. SharePoint Framework -sovelluskehitys toteutetaan ilmaisilla, avoimen lähdekoodin työkaluilla, joihin kuuluvat mm. Node.js, npm, Yoeman ja Gulp. Sovelluksen koodaukseen käytetään TypeScript-ohjelmointikieltä, joka ohjelman paketoitavaiheessa käännetään JavaScript-koodiksi.</p> <p>Työn aikana vastaan tuli kaksi isoa haastetta. Ensimmäinen haaste liittyy modernin, responsiivisen sivupohjan käyttämiseen SharePoint -sivustokokoelman mallipohjana. Kirjoitushetkellä sivustokokoelman luonnissa oli mahdollista käyttää mallipohjana suoraan vain klassisia, ei responsiivisia sivupohjia. Toinen haaste liittyy rajapintoihin ja käyttöoikeuksiin. Rajapintojen käyttöoikeuksienhallinta toteutetaan Azure AD -palvelussa OAuth 2.0 -autenttointikehyksen kautta. Yksittäiselle sivustolle voidaan sijoittaa useita SharePoint Framework -sovelluksia, mutta sivustolla voi olla voimassa vain yhden käyttöoikeudet rajapintaa kohti.</p> <p>Lokakuun lopussa Microsoftin Ignite 2017 -konferenssista julkaistujen tietojen perusteella näyttää kuitenkin siltä, että kumpikin edellä mainituista ongelmista on ratkeamassa kuluvan vuoden 2017 lopulla.</p>	
Avainsanat	Intranet, SharePoint Online, SharePoint Framework

Author Title Number of Pages Date	Jarkko Tornberg Personalization of the intranet 34 pages 15 November 2017
Degree	Bachelor of Engineering
Degree Programme	Media Technology
Instructors	Hannu Kauremaa, Senior Consultant Kari Aaltonen, Principal Lecturer
<p>The purpose of this research was to study personalization possibilities of SharePoint-based intranet service using Microsoft SharePoint Framework applications. In services like Office 365 users create a lot of personal data that can be used to personalize the intranet experience. This data can be used by applications through application programming interfaces (API). Office 365 provides many different APIs that can be utilized by programs like SharePoint Framework. During the research, I tested a few SharePoint Framework applications that could fetch user's personal data from Office 365 and presented it on SharePoint site. As a background to intranet personalization, I also investigated the shift towards digital workplace in organizations and the current intranet trends in general.</p> <p>SharePoint Framework application is a client side app running on user's browser with the rights of the current user. SharePoint Framework app consist of HTML, CSS and JavaScript code that is run on its own web part. You can add multiple SharePoint Framework web parts on a single SharePoint page. Small, re-usable components allow easy customization of SharePoint Online-based intranet solutions. SharePoint Framework application development is done with free, open source tools like Node.js, npm, Yeoman and Gulp. Application is coded with JavaScript superset, TypeScript. TypeScript code is converted to JavaScript in application build stage.</p> <p>During the research, I faced two major challenges. At the time of writing it is not possible to use modern SharePoint pages with SharePoint Framework applications as a template for a SharePoint site collection. A SharePoint intranet site collection cannot be built based on modern SharePoint experience. The other challenge is API and usage rights related. In Office 365 the authorization of rights through an API is handled with OAuth 2.0 framework. On a single SharePoint page, there can be multiple SharePoint Framework apps that use the same API. But the framework can only authorize one app at a time on a single page.</p> <p>SharePoint Framework is a new, evolving technology, therefore some remarks made in this study may not be valid later. According to information given by Microsoft in Ignite 2017 conference both challenges mentioned will hopefully be a thing of a past by the end of this year.</p>	
Keywords	Intranet, SharePoint Online, SharePoint Framework

Sisällys

1	Johdanto	1
2	Intranet murroksessa	2
2.1	Intranetin yleisiä kehitystrendejä	2
2.2	Digital workplace -toimintamalli	3
2.3	Moderni intranet	5
3	Organisaation työkalut pilvipalveluna	6
3.1	Office 365 -verkkosovelluspalvelu	6
3.2	SharePoint Online -sisällönhallintajärjestelmä	7
4	SharePoint Online -palvelun arkkitehtuuri	9
4.1	Sivustokokoelmat, sivustot ja sivut	9
4.2	SharePoint-kirjastot ja -listat	10
4.3	Verkko-osakomponentti	11
4.4	SharePoint Framework -ohjelmointirajapinta	11
5	SharePoint Online intranetratkaisuna	12
5.1	Intranetsivuston luominen SharePoint Online -palveluun	12
5.2	Intranetratkaisun laajentaminen	12
6	SharePoint Framework -sovelluskehitys	13
6.1	SharePoint Framework -sovelluskehityksen työkalut	14
6.2	Kehitysympäristön rakentaminen	16
6.3	SharePoint Online -palvelun asetukset sovellustestausta varten	16
6.4	SharePoint Framework -sovelluksen luominen	17
6.5	Sovelluksen jakeluversion luominen	18
6.6	Sovelluksen testaaminen	19
6.7	Sovelluksen asentaminen SharePoint Online -palveluun	19
6.8	Resurssitiedostojen jakelu	20
6.9	Sovellusten hyödyntäminen organisaatiossa	21
7	Intranetpalvelun personointi SharePoint Framework -sovellusten avulla	22
7.1	Office 365 -palveluiden hyödyntäminen personoinnissa	22
7.2	Hakemisto-, autentikointi- ja auktorisointipalvelut	23
7.3	Auktorisointi OAuth 2.0 -kehiksen avulla	23
7.4	Sovelluksen rekisteröiminen Azure AD -palveluun	25
7.5	Käyttöoikeuksien myöntäminen sovellukselle	27
7.6	Rajapintojen hyödyntäminen asiakassovelluksessa	28
7.7	Office 365 -palvelussa käytettävissä olevat rajapinnat	29

7.8	Haasteet rajapintojen käytössä	31
8	Loppupäätelmät	31
	Lähteet	33

Lyhenteet

SPFx	SharePoint Framework
CDN	Content Delivery Network, sisällön jakeluverkko
SPO	SharePoint Online
O365	Office 365
AAD	Azure Active Directory hakemistopalvelu
SaaS	Software as a Service, verkkosovelluspalvelu

1 Johdanto

Insinööriyön tarkoituksena on kartoittaa keväällä 2017 julkaistun SharePoint Framework -sovelluskehityksen toimivuutta osana intranetratkaisun käyttäjäkohtaista sisällön personointia SharePoint Online -ympäristössä. SharePoint Framework on uusi, kehittyvä teknologia, joka mahdollistaa SharePoint-sovelluskehityksen avoimen lähdekoodin työkaluilla. SharePoint Framework -sovellukset ovat itsenäisiä, uudelleenkäytettäviä komponentteja, joita voidaan lisätä sekä moderneille että klassisille SharePoint-sivustoille. Työn tavoitteena on löytää ratkaisuja, joka mahdollistaisivat modernin, personoidun intranetpalvelun rakentamisen käyttämällä hyväksi SharePoint Framework -sovelluksia.

Insinööriyön asiakkaana on IT-palvelutalo Edera Partners Oy. Edera Easy on Edera Partners Oy:n kehittämä konsepti, jossa asiakasyrityksen intranetratkaisu toteutetaan Microsoftin SharePoint Online (SPO) -palvelun avulla (1). Edera Easy -intranetpalvelun on tarkoitus hyödyntää asiakasyrityksen Office 365 -palvelussa syntyvää dataa ja näin löytää SharePoint Online -palvelun pohjalle rakentuva malli intranetpalvelun käyttäjäkohtaisen sisällön personoinnin toteuttamiseen.

Työssä kartoitetaan SharePoint Framework -sovelluskehityksen soveltuvuutta asiakkaan tarpeisiin. Työssä tutkitaan valmiiden SharePoint Framework -esimerkkisovellusten avulla, miten intranetpalvelun käyttäjäkohtainen sisällön personointi voitaisiin toteuttaa ja mitkä ovat ratkaisun mahdolliset haasteet. Tämän lisäksi työssä kartoitetaan yleisellä tasolla intranetpalveluiden personointia ja niiden nykytrendejä.

2 Intranet murroksessa

Intranet on organisaation sisäiseen käyttöön tarkoitettu suojattu verkkopalvelu, johon on pääsy vain organisaation jäsenillä (2). Intranet voidaan rakentaa käyttämällä sisällönhallintajärjestelmiä, kuten esimerkiksi Drupal tai erityisesti intranetpalveluiden rakentamista varten tarkoitettuja sovelluksia, kuten SharePoint, Liferay tai Igloo.

2.1 Intranetin yleisiä kehitystrendejä

Perinteisesti intranetpalvelua on käytetty yrityksen sisäiseen viestintään ja dokumenttien jakamiseen. Intranetin sisältö on ollut organisaation tuottamaa, virallista sisältöä. Intranetin luonne on kuitenkin muuttunut mm. sosiaalisen median palveluiden yleistymisen myötä. Nielsen Norman Groupin tutkimus vuodelta 2014 osoittaa, että sosiaalisen median palveluiden integroiminen intranettiin, sosiaalisen median palveluiden kaltaiset toiminnallisuudet ja intranetin personointi olivat selkeitä nousevia trendejä (3).

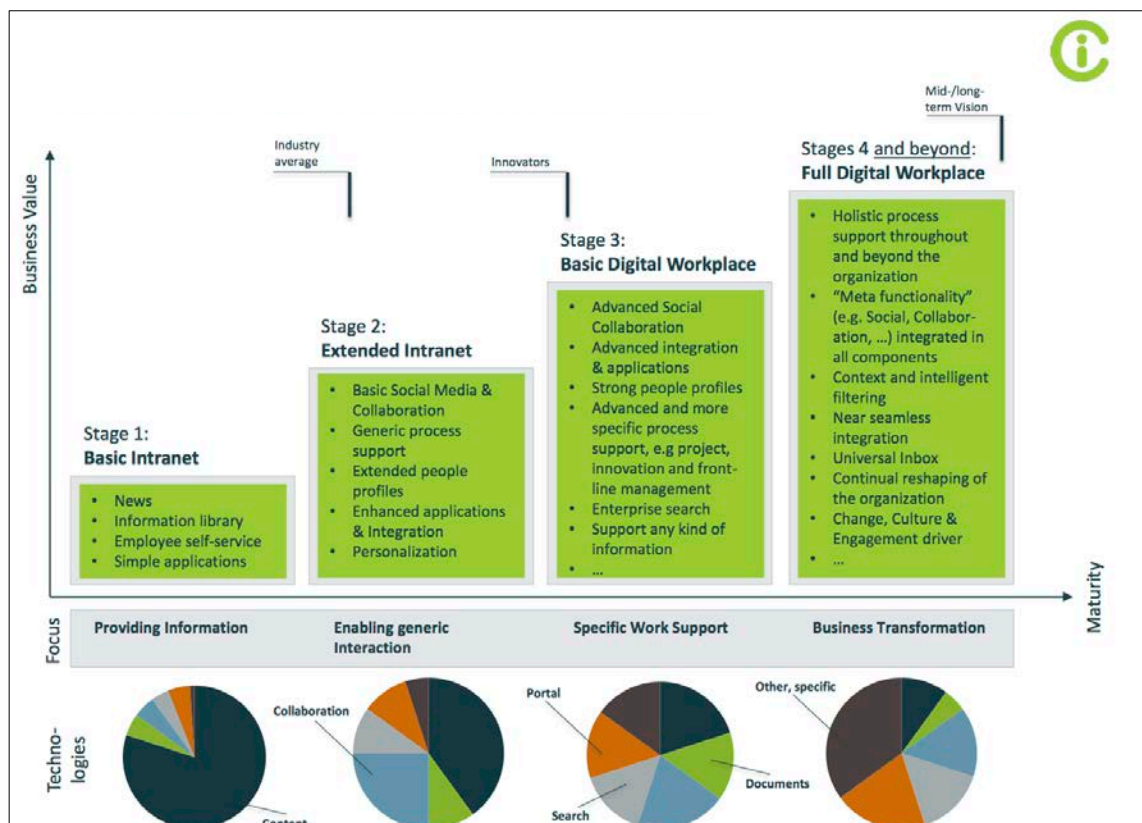
Nielsen Norman Groupin vuonna 2014 julkaiseman raportin mukaan 86 prosenttia uusista intranetsivustoista oli rakenteeltaan tehtäväpohjaisia (task-based) (4). Intranetpalveluissa olevat sovellukset, jotka auttavat käyttäjää suoriutumaan omassa työssään vetoavat käyttäjiin ja saavat heidät tulemaan takaisin intranetsivustolle (3). Myös käytettävyys mobiililaitteilla on tärkeä osa moderneja intranetratkaisuja. Norman Groupin toimittamassa 2017 Intranet Design Annual -julkaisussa arvioituja, parhaita intranetsivustoja yhdistivät mm. responsiivisuus ja käytettävyys eri päätelaitteilla. Tämän lisäksi voittajatöitä yhdistävät edellä mainitut sosiaaliset toiminnallisuudet, palvelun personointimahdollisuus ja päivittäin käytettävien työkalujen löytyminen intranetpalvelusta (5). Intranetpalvelun trendeiksi voidaan siis laskea käytettävyys mobiililaitteilla, sosiaalisen median kaltaiset toiminnallisuudet, tehtäväpohjaisuus ja personointi.

Amy Schade huomauttaa, että personoinnin voi toteuttaa joko käytettävä järjestelmä tai itse käyttäjä (6). Intranetpalvelun personoinnilla tarkoitetaan Digital Workplace Groupin julkaisun mukaan sitä, että palvelu huomioi, kuka palvelua käyttää ja esimerkiksi mistä sijainnista palvelua käytetään (7). Personointi voi intranetpalvelussa käytännössä tarkoittaa esimerkiksi sitä, että palvelu esittää käyttäjälle häntä koskevat kalenterimerkinnot, uudet sähköposti- ja pikaviestit, käyttäjän viimeksi käyttämät tiedostot ja häntä ehkä kiinnostavia uutisaiheita. Personoinnin ajatuksena on näyttää käyttäjälle sellaisia sisältöjä, joita hän tarvitsee tai joista hän mahdollisesti on kiinnostunut (6). Käyttäjät ovat tottuneet palveluiden personointiin käyttäessään kuluttajapuolen palveluita ja odottavat tätä myös organisaation tarjoamilta palvelulta.

2.2 Digital workplace -toimintamalli

Intranetpalveluiden murroksen voi katsoa liittyvän osaksi suurempaa, yritysten toimintakulttuuria koskevaa muutosta. Siirtyminen mobiiliin, liikkuvaan työhön ja digitaalisiin ratkaisuihin on pakottanut yritykset muuttamaan toimintakulttuuriaan. Yrityksissä on 2010-luvulla ollut käynnissä siirtymä kohti ns. digital workplace -toimintamallia. Gartnerin mukaan digital workplace on yrityksen liiketoimintastrategia, joka mahdollistaa uusia ja tehokkaita työskentelytapoja, parantaa työntekijöiden sitoutumista ja hyödyntää kuluttajaympäristöstä tuttuja toimintamalleja ja tekniikoita (8).

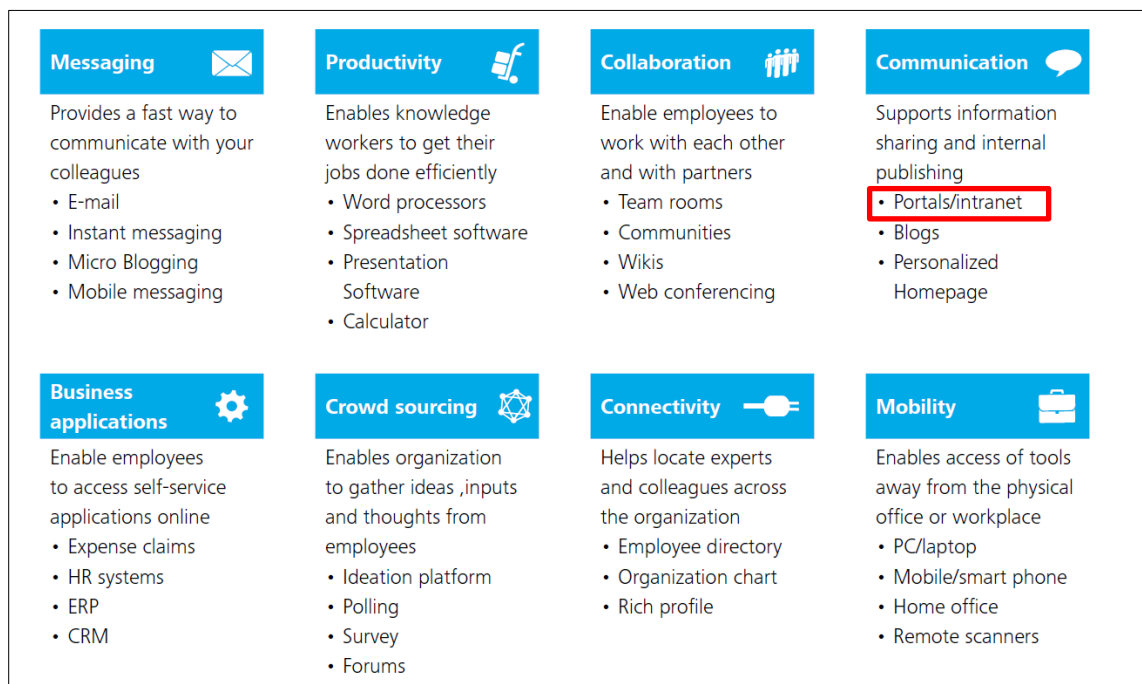
Stephan Schillerwein esittää Digital workplace -toimintamallin eräänlaisena siirtymänä pois intranetpohjaisesta lähestymistavasta (9). Schillerwein ei kuitenkaan näe digital workplace -toimintamallia intranetin uutena inkarnaationa. Schillerweinin kuvan 1 kaaviossa hyppäys intranetpohjaisesta mallista digital workplace -malliin merkitsee perinteisen intranetin sisällön (uutiset, dokumentit, tiedotteet jne.) merkityksen vähentymistä samalla kun mm. sosiaalisen yhteistyön ja portaali-tyyppisen toiminnallisuuden osuus organisaation käyttämissä teknologisissä ratkaisuissa kasvaa.



Kuva 1. Siirtyminen kohti kokonaisvaltaista, digitaalista liiketoimintamallia (9).

Forbesissa vuonna 2014 julkaistun artikkelin mukaan olennaisia elementtejä uuden digital workplace -työskentelymallin luomisessa ovat työkalujen käytettävyys eri päätelaitteilla, sujuvan kommunikaation mahdollistaminen ja liiketoiminnan sovellusten käytettävyys ajasta ja paikasta riippumatta (10). Digital workplace -mallin yhteydessä puhutaan usein ns. digitaalisesta työkalupakista (digital workplace toolbox), joka sisältää kaikki ne työkalut ja teknologiat, jotka auttavat yrityksen työntekijöitä selviytymään päivittäisistä työtehtävistään. Deloitteen mukaan digitaalinen työkalupakki sisältää kaikki yrityksen liiketoiminnassa käyttämät sovellukset lähtien yrityksen henkilöstöhallinnan sovelluksista, päätyen pikaviestintätyökaluihin (11).

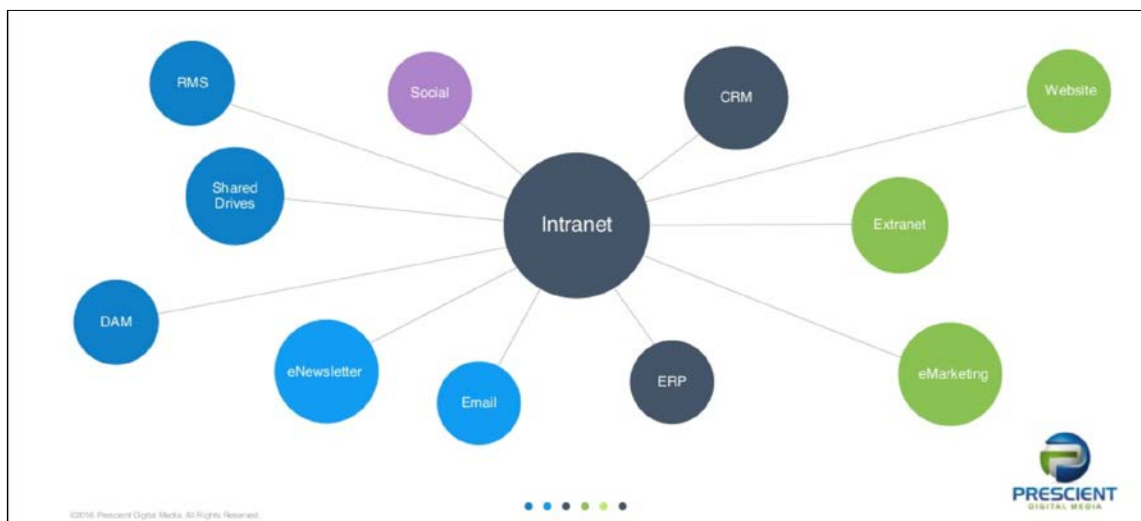
Deloitteen näkemyksessä (11) intranet näyttäytyy vain yhtenä pienenä osana yrityksen digitaalista työkalupakkia. Deloitteen jaottelussa (kuva 2) on erotettavissa eri sovelluskategorioita, mm. kahdenväliseen ja organisaatiotason viestintään liittyvät sovellukset, perinteiset toimistosovellukset, yrityksen henkilöstöhallintaan ja liiketoimintaan liittyvät sovellukset sekä yhteistyöhön, informaation jakamiseen ja sisäiseen viestintään liittyviä sovellukset ja palvelut, joista yhtenä voi toimia yrityksen intranet.



Kuva 2. Yrityksen digitaalinen työkalupakki Deloitteen mukaan (11).

2.3 Moderni intranet

Myös intranetin täytyy sopeutua työelämän murrokseen. Se ei voi enää olla pelkästään organisaation dokumenttien ja tiedotteiden jakamiseen tarkoitettu kanava, vaan monipuolinen digitaalinen työkalu, joka integroituu osaksi muita organisaation käyttämiä työkaluja ja palveluita. Intranetin ei tarvitse eikä se voi täyttää kaikkia organisaation tarpeita. Intranetin ei tarvitse myöskään korvata jo olemassa olevia sovelluksia ja palveluita. Se voi toimia portaalina yrityksen muihin digitaalisiin työkaluihin. Prescient Digital Media yrityksen mukaan (kuva 3) tämä voi tarkoittaa esimerkiksi intranetpalvelun aloitussivun toimimista sijaintina, josta käyttäjä voi helposti siirtyä muihin organisaation tarjoamiin työkaluihin ja palveluihin (12).



Kuva 3. Intranet portaalina (12).

Intranet voi yhdistää perinteisen organisaation tiedotuskanavan personoituun kokemukseen, jossa käyttäjä voi tarkastella itseään kiinnostavia resursseja. Intranetpalveluun voidaan tuoda sosiaalisesta mediasta tuttuja toiminnallisuuksia ja panostaa yhteistyön ja kommunikaation helpottamiseen. Modernin intranetpalvelun tärkeimmiksi ominaisuuksiksi voidaan laskea

- yhteistyön ja jakamisen mahdollistaminen
- monipuoliset viestintämahdollisuudet, myös sosiaalinen media ja sosiaalisen median kaltaiset toiminnallisuudet
- pääsy tarvittaviin resursseihin ajasta ja paikasta riippumatta
- palvelun käytettävyys mobiilisti ja kaikilla päätelaitteilla
- palvelun personointimahdollisuus
- tehokas, koko palvelun kattava hakutoiminto.

3 Organisaation työkalut pilvipalveluna

Digital workplace -mallin mukaan organisaation henkilöstön tarvitsemien ohjelmistojen ja muiden resurssien tulisi olla käytettävissä ajasta ja paikasta riippumatta. Käytännössä tämä usein tarkoittaa yrityksen käyttämien ohjelmistojen ja dokumenttien siirtämistä pilvipalveluihin, jossa ne ovat käytettävissä päätelaitteesta riippumatta siellä, missä yrityksen työntekijällä on verkkoyhteys. Samalla kun yrityksen sovellukset siirtyvät pilvipalveluihin, on sovellusten kautta syntyvää dataa mahdollista käyttää hyväksi yrityksen intranetpalvelun personoinnissa.

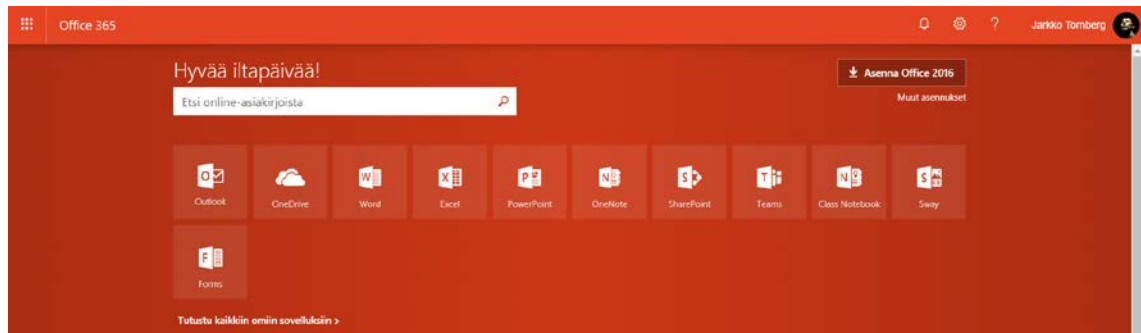
Yritysten perinteiset toimistosovellukset voi siirtää pilveen käyttämällä esimerkiksi Microsoftin Office 365- tai Googlen G Suite -pilvipalveluita. Pilvipalveluiden muita käyttökohteita yrityksissä ovat mm. tiedoston tallennus- ja jakamispalvelut (esimerkiksi Google Drive, Dropbox ja OneDrive for Business), sähköposti-, kalenteri- ja pikaviestipalvelut (mm. Gmail ja Outlook Web App) sekä yrityksille suunnatut yhteistyö- ja yhteisöpalvelut (esimerkiksi Slack tai Yammer). Yrityksille on tarjolla myös palveluita, kuten Amazon Web Services, Google Cloud Platform tai Microsoft Azure, jotka mahdollistavat yritysten omien ja kolmannen osapuolen sovellusten siirtämisen pilveen.

3.1 Office 365 -verkkosovelluspalvelu

Office 365 (O365) on Microsoftin tarjoama ns. verkkosovelluspalvelu (Software as a Service). Verkkosovelluspalvelu on ohjelmistojen tilauspohjainen lisensiointi- ja jakelumalli, jossa palveluntarjoaja tarjoaa asiakkaalleen valmiin sovelluspaketin pilvipalveluna. Palvelun tilaajan huoleksi jää vain käytettävien sovellusten räätälöinti (13, s. 3–4). O365-palvelun tarjoamat sovellukset voidaan jakaa toiminnallisesti kolmeen kategoriaan: viestintä, dokumenttien hallinta ja yhteistyöpalvelut. Kategoriat vastaavat Deloitteen vuonna 2011 julkaisemassa artikkelissa määriteltyjä Digital workplace -mallin avainajatuksia: tee yhteistyötä (collaborate), kommunikoi (communicate) ja yhdistä (connect) (11).

O365 koostuu useista erilaisista palveluista, joita ovat mm. Office Online -sovellukset, Microsoft Teams, Yammer, Skype for Business, Outlook, OneDrive for Business ja SharePoint Online. O365 mahdollistaa Microsoftin Azure AD -palvelun kautta sekä organisaation omien että kolmannen osapuolen sovellusten liittämisen palveluun niin, että kaikkiin sovelluksiin pääsee suoraan Office 365 -palvelusta. Sovellusten käyttäminen on usein mahdollista toteuttaa kertakirjautumisen avulla, jolloin O365-palveluun ja kaikkiin siihen liitettyihin palveluihin voidaan kirjautua yhdellä käyttäjätunnuksella ja salasanalla.

Office 365 -palvelu (kuva 4) voi näin toimia portaalina kaikkiin organisaation käyttämiin sovelluksiin ja palveluihin.



Kuva 4. Office 365 -palvelun aloitussivu.

Office 365 -palveluiden kautta syntyy hyvin paljon yksittäiseen käyttäjään liittyvää dataa. Käyttäjä saa henkilökohtaista sähköpostia ja erilaisia pikaviestejä, käyttäjän tiimeissä käydään keskusteluita, jaetaan tiedostoja jne. Microsoftin vuoden 2017 loppupuolella julkaistun raportin mukaan O365-palveluilla oli 120 miljoonaa kuukausittaista aktiivista käyttäjää (13), joista esimerkiksi Outlook-sähköpostipalvelun käyttäjät lähettävät päivittäin neljä biljoonaa sähköpostia. Kaikkea tätä käyttäjien itse tuottamaa dataa voidaan tarvittaessa hyödyntää personoitaessa SharePoint Online -palvelun päälle rakennettua intranetpalvelua käyttämällä Microsoftin tarjoamia O365-rajapintoja.

3.2 SharePoint Online -sisällönhallintajärjestelmä

SharePoint on Microsoftin tarjoama verkkopohjainen sisällönhallintajärjestelmä ja yhteistyö- ja ryhmätyökalu, joka mahdollistaa mm. sisällön jakamisen, ryhmätyöskentelyn ja dokumenttien yhteiskäytön (14). Se on tarjolla sekä Microsoftin ylläpitämänä SharePoint Online -pilvipalveluna että paikallisesti ylläpidettävänä SharePoint Server -palveluna (15). Lisäksi Microsoft tarjoaa hybridimallia, jossa yrityksen käytössä ovat sekä SharePoint Online että paikallisesti ylläpidettävä SharePoint Server (16, s. 8). SharePoint-palvelua kehitetään Microsoftin mukaan painottaen SharePoint Online -pilvipalvelun kehittämistä ja panostamalla palvelun mobiilikäyttöön (14).

SharePoint Online (SPO) on Microsoftin tarjoama kuukausimaksullinen verkkosovelluspalvelu, joka voidaan hankkia joko itsenäisenä palveluna tai osana Office 365 -palvelupakettia (16). SPO koostuu useasta toiminnallisesta komponentista, jotka yhdessä muodostavat palvelun ominaisuudet, joita ovat mm. sivustokokoelmat ja sivustot sekä listat ja kirjastot (17). SharePoint Online on varteen otettava vaihtoehto intranetpalvelun alustaksi silloin, kun organisaatiolla on jo käytössään Office 365 -palvelu.

SPO on rakennettu toimimaan yhdessä muiden Office 365 -sovellusten ja toiminnallisuuksien kanssa. SharePoint-dokumenttikirjastossa olevia Office-dokumentteja voidaan avata suoraan selaimessa Office Online -sovelluksilla. Office 365 -ryhmät ja -tiimit saavat automaattisesti käyttöönsä oman ryhmä- ja/tai tiimikohtaisen SharePoint-tiimisivuston. Lisäksi olemassa oleva SharePoint-tiimisivusto voidaan liittää Office 365 -ryhmään tai tiimiin (18). Nämä ja muut vastaavat ominaisuudet helpottavat SharePoint Online -palvelun integroimista osaksi muita organisaation käyttämiä digitaalisia työkaluja.

SharePoint-palvelun kehityskaaressa on havaittavissa selkeä siirtymävaihe. Siirtymävaiheen voi määritellä johtuvan kolmesta eri seikasta:

- siirtymisestä kohti liikkuvaa työtä ja mobiilikäyttöä
- siirtymisestä pois perinteisestä intranetistä
- SharePoint-sovelluskehityksen murroksesta

SharePoint-sivustojen sujuva mobiilikäyttö vaatii sivustolta responsiivisuutta. Tämän vuoksi SharePoint-palveluun on julkaistu uusi ns. moderni käyttökokemus, joka tukee sivujen responsiivisuutta. SharePoint tarjoaa näin loppukäyttäjälle kahta erilaista käyttökokemuskäyttöä, klassisen ja modernin. Jako klassiseen ja moderniin käyttökokemukseen koskee SharePoint-listoja ja -kirjastoja, -sivustoja ja -sivutyyppejä sekä sivuille oletuksena lisättävissä olevia verkko-osakomponentteja (19). Moderni käyttökokemus mahdollistaa responsiivisten sivujen toteuttamisen käyttövalmiina, ilman oman responsiivisen sivumallipohjan luomista. Mobiilikäyttöön panostaminen näkyy myös Microsoftin julkaisemassa, mobiililaitteille tarkoitettussa SharePoint-mobiilisovelluksessa, joka mahdollistaa SharePoint-intranetpalvelun käytön esimerkiksi matkapuhelimen avulla (20).

Useat perinteisesti intranetistä löytyvät palvelut on O365-palvelussa korvattu erillisillä palveluilla. Esimerkiksi tiedoston tallennus hoidetaan Office 365 -palvelussa usein käytämällä OneDrive for Business -palvelua, SharePoint tiimit on ”korvattu” Office 365 -ryhmillä ja Microsoft Teams -palvelun tiimeillä. SharePoint-intranetpalvelusta on tullut vain yksi osa yrityksen digitaalista työkalupakkia. Intranetpalvelun olisi hyvä toimia yhdessä muiden organisaation käyttämien sovellusten ja palveluiden kanssa. SharePoint Online on integroitu hyvin osaksi muita Office 365 -palveluja. Näiden muutosten lisäksi myös käyttäjien yleiset odotukset intranetpalveluita kohtaan ovat muuttuneet. Käyttäjät odottavat palvelun personointia, joka voidaan toteuttaa esimerkiksi hakemalla käyttäjiä koskevaa, personoitua Office 365 -palvelussa syntynyttä tietoa käyttämällä SharePoint Framework -sovelluksia.

Kolmas muutos koskee SharePoint sovelluskehitystä. Perinteisesti SharePoint-sovelluskehitys on ollut palvelun heikkoja kohtia (22, s. 36–41). Microsoft on SharePoint Framework -sovelluskehityksen julkistuksen myötä muuttanut lähestymistapaansa ja mahdollistanut SharePoint-sovelluskehityksen avoimen lähdekoodin työkalujen avulla. Microsoft on julkisesti ilmoittanut käyttävänsä itse samoja työkaluja omaan SharePoint -sovelluskehitykseensä (21).

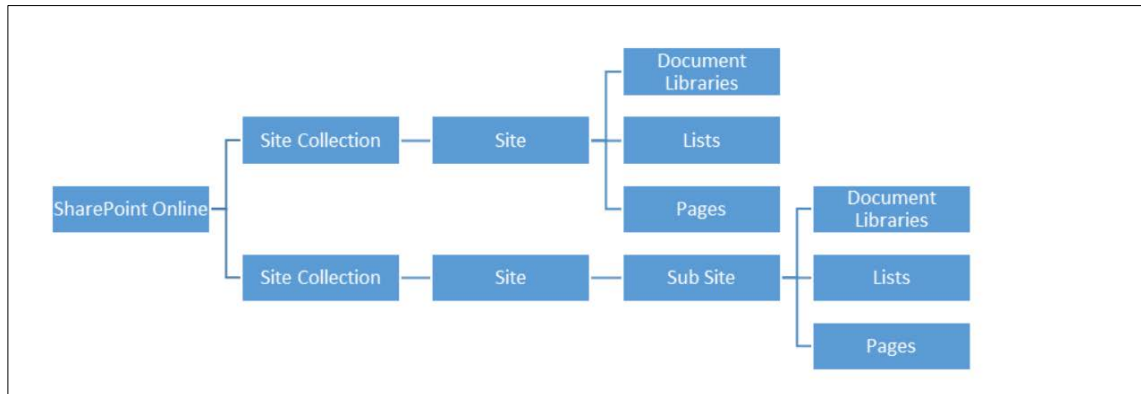
4 SharePoint Online -palvelun arkkitehtuuri

Organisaatiot käyttävät SharePoint-palvelua luomalla palveluun verkkosivuja ja -sivustoja, jotka sisältävät, esittävät ja keräävät yhteen SharePoint-palvelun sekä tarvittaessa muiden verkkopalveluiden tuottamaa sisältöjä. SharePoint-sivuja voidaan käyttää organisaation sisällä useisiin eri käyttötarkoituksiin, ei ainoastaan intranetsivustona. SharePoint-sivuja voivat luoda palvelun järjestelmänvalvojat ja heidän niin salliessaan myös palvelun peruskäyttäjät. Peruskäyttäjän luomat SharePoint-sivut ovat aina ns. moderneja sivuja, ja ne luodaan yleensä yksittäisen käyttäjän tai tiimin tarpeisiin, eikä niitä ole tarkoitettu koko yrityksen intranetpalvelun rakennuspalikoiksi. Peruskäyttäjän on vaikea ja osittain myös mahdotonta muokata luomiinsa sivuihin liittyviä asetuksia niin, että ne voisivat palvella koko organisaation intranetsivustona. SharePoint-sivusto syntyy automaattisesti aina myös silloin, kun O365-palveluun luodaan uusi Office 365 -ryhmä tai tiimi (22). Nämä SharePoint-sivustot on tarkoitettu pääasiassa tietyn ryhmän tai tiimin käyttöön.

4.1 Sivustokokoelmat, sivustot ja sivut

SharePoint-palvelun ylläpitäjän luomat sivustot koostuvat sivustokokoelmista (site collection), niiden sisältämistä sivustoista ja alisivustoista (site/subsite) sekä yksittäisistä sivuista (page). Sivustokokoelma on hallinnollinen yksikkö, jonka alle ylläpitäjä voi luoda yhden tai useampia sivustoja tai alisivustoja. Sivustokokoelma pitää sisällän omat sivustokokoelmakohtaiset asetukset. Niiden kautta voidaan hallita esimerkiksi sivustokokoelman sisältämien sivustojen, kirjastojen jne. oikeuksia ja niiden periytymistä, käytössä olevia sisältötyyppejä (content type) ja verkko-osia (web part) (25, s. 267). Tämän lisäksi sivustokokoelman ominaisuuksien (site collection features) kautta määritellään sivustokokoelman sisältämien sivustojen käytössä olevat ominaisuudet (26, s. 10).

Sivustot ja alisivustot sisältävät mm. kirjastoja ja listoja sekä sivuja. Sivu on aina osa jotakin sivustoa, ja sivusto on aina osa jotakin sivustokokoelmaa (kuva 5).



Kuva 5. SharePoint-sivustokokoelma, sivustot ja sivut (22, s. 48).

Sivut ovat näkymiä, joilla SharePoint-palvelusta tai jostain muusta verkkopalvelusta peräisin oleva data esitetään käyttäjälle. Sivunäkymä koostuu pääosin verkko-osa komponenteista ja niiden kautta esitettävästä tiedosta sekä mm. muotoillusta tekstistä ja kuvista. SharePoint toimii tältä osin kuin perinteinen julkaisujärjestelmä, joka kokoaa sivun sisällön käyttäjän selaimelle yhteen sivunäkymään. Sisältö esitetään aina yksittäiselle sivulla, jota vastaa yksi aspx-dokumentti. Julkaisujärjestelmän tapaan SharePoint sisältää myös useita valmiita verkko-osa komponentteja, joita voidaan käyttää SharePoint-sivulla.

Sivustokokoelmien luonti

Sivustokokoelman voi luoda vain Office 365 -palvelun tai SharePoint-palvelun järjestelmänvalvoja. Sivustokokoelma luodaan aina ns. mallipohjan (template) pohjalta (25, s. 268). Kirjoitushetkellä sivustokokoelman luontiin tarkoitettuja mallipohjia on olemassa vain klassisille SharePoint-sivuille. Moderneja sivustomalleja ovat ns. tiimisivustot (team sites) ja kommunikaatiosivustot (communication sites), mutta nämä sivumallit eivät ole kirjoitushetkellä valittavissa luotaessa sivustokokoelmaa.

4.2 SharePoint-kirjastot ja -listat

SharePoint-palvelussa tietoa tallennetaan pääosin SharePoint-kirjastoihin ja -listoihin. Listalla tarkoitetaan SharePoint-ympäristössä taulukkomuotoista tapaa tallentaa tietoa (23). Lista koostuu riveistä ja sarakkeista ja voi sisältää erityyppistä dataa, kuten tekstiä, päivämääriä jne. Kirjasto on SharePoint-sivustolla sijaitseva kohde, jonka kautta käyttäjä voi tallentaa, luoda, jakaa ja ladata erilaisia dokumentteja (24). Kirjastot voivat tallentaa ja esittää tallennetuista dokumenteista myös paljon erilaista metadatatietoa.

SharePoint sisältää kaksi erilaista käyttökokemusta, modernin ja klassisen. Käyttökokemus koskee myös SharePoint-listoja ja -kirjastoja. Klassiset ja modernit listat ja kirjastot eroavat toisistaan pääosin ulkoasun osalta. Moderneissa listojen ja kirjastojen toiminnallisuuksissa on myös pieniä eroavaisuuksia verrattaessa niitä klassisiin (25). SharePoint Online sisältää useita valmiita lista- ja kirjastotyyppisiä, joita voidaan käyttää SharePoint-sivuilla. Valmiita lista- ja kirjastotyyppisiä ovat mm. kalenteri, tehtävälista, dokumenttikirjasto ja kuvakirjasto. SharePoint-sivustolla voi olla useita erilaisia kirjastoja ja listoja. Niissä olevia tietoja voidaan tarkastella ja muokata SharePoint REST -rajapinnan kautta. Tämä mahdollistaa esimerkiksi Office 365 -ryhmien ja tiimien SharePoint-sivustojen tiedon hyödyntämisen SharePoint-intranetpalvelun personoinnissa.

4.3 Verkkosivuosakomponentti

Verkkosivuosakomponentti (web part) on SharePoint-sivustolla oleva asiakkaan selaimessa suoritettava, uudelleen käytettävä komponentti (30, s. 43). Sekä modernit että klassiset SharePoint-sivut sisältävät oletuksena valmiita verkkosivuosia. Klassiset SharePoint-sivut voivat sisältää myös ns. sovellusosakomponentteja (app part), joilla tarkoitetaan verkkosivuosakomponenttia, jonka sisällön luo SharePoint Add-in -sovellus (26). Jokainen lista ja kirjasto synnyttää automaattisesti kyseistä listaa tai kirjastoa vastaavan verkkosivuosakomponentin, joka voidaan vastaavasti lisätä SharePoint-sivulle. SharePoint-sivun sisältö koostuu pääosin verkkosivuosista ja niiden esittämästä tiedosta.

4.4 SharePoint Framework -ohjelmointirajapinta

SharePoint Framework (SPFx) on Microsoftin tarjoama uusi ohjelmointirajapinta ja kehitysympäristö asiakkaan selaimessa suoritettavien (client-side) sovellusten luomiseen. SharePoint Framework tarjoaa rajapinnan sekä Microsoftin omalle että kolmannen osapuolen SharePoint-sovelluskehitykselle (21). Kaikki moderneilla SharePoint-sivustoilla tarjolla olevat valmiit, Microsoftin tarjoamat verkkosivuosat on luotu SharePoint Frameworkin avulla. SharePoint Frameworkin avulla voidaan luoda omia verkkosivuosakomponentteja hyödynnettäväksi sekä klassisilla että moderneilla SharePoint-sivustoilla sekä SharePoint Online -palvelussa että pian myös paikallisessa SharePoint 2016 (Feature Pack 2) Server -palvelinohjelmistossa (21).

SharePoint Framework tuo mukanaan uudentyyppisen SharePoint-verkkosivuosan, jonka sisällä SharePoint Framework -sovellukset ajetaan. Kolmannen osapuolen SPFx-sovellukset voidaan ottaa organisaatiossa käyttöön siten, että SPFx-verkkosivuosien lisääminen

myös peruskäyttäjien luomille sivuille tai Office 365 -ryhmien ja -tiimien SharePoint-sivuille on mahdollista (27). Näin organisaation käyttöön ottamia SPFx-sovelluksia voidaan hyödyntää monipuolisesti Office 365 -palvelun sisällä.

5 SharePoint Online intranetratkaisuna

Microsoftin mukaan yli 200 000 organisaatiota käyttää SharePoint-palvelua intranetialustana (28). Roine ja Anttila kertovat puolestaan joka toisen suuren tai keskisuuren suomalaisen yrityksen käyttävän sitä oman intranetratkaisunsa pohjana (34, s. 32). Paikalliselle palvelimelle asennetun SharePoint-palvelun avulla luotu intranetratkaisu on ollut monimutkainen toteuttaa ja myös raskas ylläpidettävä. SharePoint Online -palvelu helpottaa intranetpalvelun ylläpitoa, eikä palvelun käyttöönotto pilvipalveluna vaadi asentamista.

5.1 Intranetsivuston luominen SharePoint Online -palveluun

Intranet luodaan SharePoint Online -palveluun yleensä luomalla sille oma sivustokokoelma niin, että intranetpalvelun aloitussivuksi määritellään organisaation SharePoint-palvelun oletussivu esimerkiksi "organisaatio.sharepoint.com" (25, s. 267). Intranetpalvelun sijoittaminen omaan sivustokokoelmaan helpottaa mm. palvelun käyttöoikeuksien hallintaa, yhtenäisen ulkoasun ja navigaation rakentamista.

SharePoint-intranetpalvelu luodaan yleensä käyttämällä sivustokokoelman kaikissa sivuissa samaa sivun mallipohjaa (template). Mallipohja määrittelee, mitä listoja, verkko-osia jne. sivu oletuksena sisältää. Lisäksi mallipohjan kautta voidaan määrittellä sivun ulkoasua (25, s. 268). SharePoint Online -palvelussa saatavilla olevat mallipohjat edustavat kaikki klassista SharePoint-käyttökokemusta. Kirjoitushetkellä SharePoint Framework -sovelluksia sisältävää modernia SharePoint Online -sivustokokoelmaa ei voi suoraan luoda. Tämä on haaste moderniin käyttökokemukseen pohjaavan intranetratkaisun rakentamiselle.

5.2 Intranetratkaisun laajentaminen

SharePoint Online -palvelu sisältää valmiita verkko-osakomponentteja, jotka luovat sivulle haluttuja toiminnallisuuden, kuten erilaisia listoja ja kirjastoja. Valmiiden ominaisuuksien lisäksi palvelua voidaan laajentaa erilaisilla sovelluksilla. SharePoint Online tukee tällä hetkellä käytännössä kahta erilaista tapaa lisätä palvelun toiminnallisuuksia omien sovellusten avulla. Vaihtoehtoina on käyttää joko SharePoint Add-in -sovelluksia

tai SharePoint Framework -sovelluksia. Näiden lisäksi SharePoint-sivustolla voidaan ajaa pienimuotoisia, JavaScript-koodia sisältäviä sovelluksia Script editor -verkko-osakomponentin sisällä.

Markkinoilla on jo muutamia palveluntarjoajia, kuten Valo Intranet, jotka tarjoavat SharePoint Online -palvelun päälle rakennettua intranetratkaisua, jota on laajennettu käyttämällä SPFx-komponentteja (29). SharePoint Framework -komponenteilla laajennettu SharePoint Online -intranetratkaisu mahdollistaisi intranetpalvelun personoinnin esittämällä yksittäisen käyttäjään liittyvää Office 365 -palvelussa syntyvää informaatiota. Yksittäinen SharePoint Framework -sovellus voi noutaa tiettyyn Office 365 -palveluun tai yrityksen käytössä olevaan kolmannen osapuolen palveluun liittyvää informaatiota ja esittää sen käyttäjälle SPO-intranetsivustolla. Yritys voisi valita, mitä SharePoint Framework -komponentteja se käyttää omassa intranetpalvelussaan. Tämän lisäksi yksittäinen käyttäjä voisi valita, mitä SharePoint Framework -komponentteja hän haluaa lisätä esimerkiksi SharePoint Online -tiimisivustolle.

Office 365 on jatkuvasti kehittyvä ja laajentuva palvelu. Saatavana olevien palveluiden määrä voi kasvaa ja palveluiden toimintaperiaatteet voivat muuttua. (12, s. 4). Modulaarinen SPFx-sovelluksiin pohjaava lähestymistapa intranetin toiminnallisuuden laajentamiseksi helpottaa uusien palveluiden pohjalle rakennettävien toiminnallisuuden lisäämistä intranetsivustolla ja mahdollistaa muutosten tekemisen niin, että yhteen komponenttiin tehtävät muutokset eivät vaikuta toisiin komponentteihin.

6 SharePoint Framework -sovelluskehitys

Sharepoint-sovellukset voidaan jakaa ns. palvelinpuolen ja asiakaspuolen sovelluksiin. Palvelinpuolen sovelluksissa ohjelmakoodi suoritetaan joko ulkoisella palvelimella (esim. Azure-pilvipalvelussa) tai SharePoint-palvelimella. Asiakaspuolen sovelluksissa ohjelmakoodia suoritetaan pääosin asiakkaan päätelaitteella eli käytännössä asiakkaan käyttämässä verkkoselaimessa. SharePoint Framework -sovellukset ovat asiakaspuolen sovelluksia, sillä niiden ohjelmakoodi suoritetaan käyttäjän selaimessa.

SPFx-sovellus on käyttäjän selaimessa toimiva verkko-osakomponentti. SPFx-sovellus voi sisältää HTML-, CSS- ja JavaScript-koodia. Käytännössä SPFx-sovellus voi hyödyntää melkein mitä tahansa olemassa olevaa JavaScript-kirjastoa. Selaimessa suoritettavan koodin lisäksi SPFx-sovellus voi hyödyntää myös palvelinpuolen koodia, joka on tallennettu esim. Azure-pilvipalveluun. SPFx-sovelluksia on kirjoitushetkellä tarjolla

kahta eri tyyppiä, SharePoint Framework Client side verkko-osakomponentteja ja ns. SharePoint Framework extension -komponentteja (30). Tässä työssä käsitellään ainoastaan SharePoint Framework client side verkko-osakomponentteja.

6.1 SharePoint Framework -sovelluskehityksen työkalut

SharePoint Framework -kehityksessä käytetään perinteisiä, avoimen lähdekoodin ns. web stack -työkaluja. Web stack -termillä tarkoitetaan yleisesti verkkosovelluksen kehitykseen tarkoitettua työkalukokoelmaa, joka voi pitää sisällään sekä asiakas- että palvelinpuolen työkaluja (31). Kyseessä on siis eri ohjelmistoista koostuva ”pino”, jonka avulla ohjelmiston kehitystyö voi edetä saumattomasti ja mahdolliset usein toistuvat, rutiininomaiset vaiheet, kuten moduulien lataaminen, riippuvuuksien hallinta, sovelluksen paketointi jakelua varten sekä siirto verkkopalvelimelle, voidaan automatisoida. Yksittäinen web stack voi pitää sisällään WWW-palvelinsovelluksen (esim. Node.js), pakettienhallintajärjestelmän, moduulien latausjärjestelmän, tietokannan ja erilaisia kirjastoja ja ohjelmistokehityksiä.

SPFx-sovelluskehityksessä käytettyjä työkaluja ovat mm. Node.js, npm, Yoeman ja Gulp. Näiden lisäksi on tarvittaessa mahdollista käyttää melkein mitä tahansa JavaScript-ohjelmistokehityksiä ja kirjastoja, kuten mm. React ja Angular. Itse ohjelmakoodi toteutetaan käyttämällä TypeScript-ohjelmointikieltä. Sovelluksen koodaaminen voidaan suorittaa esimerkiksi Microsoftin Visual Studio Code -ohjelmistolla. Samoja työkaluja, joita käytetään perinteiseen verkkosovellusten kehitykseen, voidaan nyt käyttää myös SharePoint-sovelluskehityksessä.

Node.js

Node.js on palvelinpuolen ns. ajonaikainen (runtime) ympäristö JavaScript-koodin suorittamiseen (32). Node.js-ympäristöä käytetään SharePoint Framework -sovelluskehityksessä pääasiassa ohjelmiston kehitystyön ajan palvelinympäristönä. Node.js-palvelinta voidaan käyttää apuna SPFx-sovelluksen testaamisessa paikallisella Node.js-palvelimelle SharePoint Workbench -ympäristön kautta.

npm

npm eli node packet manager on pakettienhallintaan tarkoitettu sovellus mm. Node.js-ympäristöön. npm-pakettienhallintasovellusta käytetään kolmannen osapuolen kirjastojen ja muiden JavaScript-moduulien lataamiseen ja hallintaan sovelluskehityksen aikana (33).

Yoeman

Yoeman on ohjelmistokehityksessä käytetty ns. tukirakennetyökalu (scaffolding tool). Yoeman luo ohjelmistokehitysprojektin tarvitsemat lähtötiedostot ja kansiorakenteen, lataa projektissa tarvittavat perusmoduulit sekä tarjoaa tehtävänkäsittelijän ja tehtävät mm. projektin testaamista ja paketointia varten (34). Yoeman-ohjelmisto koostuu useasta erillisestä työkalusta, joilla jokaisella on oma tehtävänsä. Yo-ohjelmisto luo projektin rungon. Yo-ohjelmiston lisäksi Yoeman-sovelluksen avulla luotu projekti sisältää aina tehtävänkäsittelijän. Yoeman mahdollistaa kirjoitushetkellä kahden eri tehtävänkäsittelijän, Gulp ja Grunt, käyttämisen. SharePoint Framework -generaattori käyttää tällä Gulp-tehtävänkäsittelijää (34).

Yoeman SharePoint Framework -generaattori

Yoeman käyttää projektin luomisessa ns. generaattoreita, jonka avulla luodaan projektin tarvitsemat lähtötiedostot ja kansiorakenteet ja ladataan mahdollisesti tarvittavat kirjastot npm-pakettienhallintasovelluksen avulla. SharePoint Framework -sovelluskehityksessä käytetyn Yoeman-generaattorin kehityksestä vastaa Microsoft. Sen tarjoamaa @microsoft/generator-sharepoint -generaattoria käytetään SharePoint Framework -projektissa projektin rungon luomiseen ja tarjoamaan Gulp-tehtäviä esimerkiksi valmiin sovelluksen pakointiin (35).

Gulp

Gulp on ns. tehtävänkäsittelijä (Task runner). Tehtävänkäsittelijällä tarkoitetaan ohjelmistoa, jonka avulla voidaan automatisoida ohjelmistonkehityksessä toistuvia, mekaanisia työvaiheita, kuten ohjelmakoodin minimoimista, sovelluksen jakeluversion rakentamista jne. Gulp-ohjelmistoa käytetään SharePoint Framework -kehityksessä mm ohjelmakoodin jakeluversion luomiseen ja minimointiin. Gulp vastaa perinteisessä SharePoint-sovelluskehityksessä Visual Studio -ohjelmiston kanssa käytettyä MS Build -työkalua. SharePoint Framework -sovelluskehityksessä käytetty Yoeman-generaattori pitää sisällään muutaman valmiin Gulp-tehtävän.

TypeScript

SharePoint Framework sovellusten koodaamiseen suositellaan TypeScript-ohjelmointikieltä. Se on JavaScript-kielen ns. "ylijoukko" (superset). TypeScript-koodi käännetään sovelluksen pakointivaiheessa perinteiseksi JavaScript-koodiksi. Tämän vuoksi myös JavaScript-koodia voidaan käyttää SharePoint Framework -kehityksessä. TypeScript

mahdollistaa JavaScriptiin nähden ohjelmiston kehitysvaiheessa mm. muuttujien vahvan tyyppityksen.

Office UI Fabric

Esitelyjen työkalujen lisäksi SharePoint Framework -kehityksessä hyödynnetään Office UI Fabric -kirjastoa. Office UI Fabric helpottaa yhdenmukaisen käyttöliittymän luomista SharePoint Framework -sovelluksiin tarjoamalla kokoelman valmiita kuvakkeita, tyylejä ja komponentteja. Office UI Fabric -kirjastoa on mahdollista käyttää joko liittämällä Office UI Fabric suoraan sivun lähdekoodiin tai React-, AngularJS- tai Fabric JS -ympäristöihin (36).

6.2 Kehitysympäristön rakentaminen

SharePoint Framework -sovelluskehitys vaatii kehitysympäristön rakentamista omalle tietokoneelle. Kehitysympäristö on mahdollista rakentaa Windows-, Mac Os- ja Linux-ympäristöihin (37). Kehitysympäristön rakentaminen alkaa Node.js-ympäristön ja npm-pakettienhallintasovellusten asentamisella. Tämän jälkeen voidaan asentaa Yoeman- ja Gulp-ohjelmistot sekä tarvittava SharePoint Framework Yoeman-generaattori (37). Näiden lisäksi ohjelmistokehitykseen tarvitaan käyttötarkoitukseen soveltuva koodieditori. Microsoft suosittaa ilmaista Visual Studio Code -ohjelmistoa. Valitussa koodieditorissa olisi hyvä olla tuki TypeScript-ohjelmointikielelle. Sovelluskehityksessä voidaan käyttää apuna myös muita ohjelmistoja, kuten mm. versionhallintasovelluksia tai -palveluita.

6.3 SharePoint Online -palvelun asetukset sovellustestausta varten

SharePoint Framework -sovellusten testaamista varten tarvitaan Office 365 -palvelun tilaus. Office 365 -palveluun on luotava järjestelmänvalvojan toimesta sovelluskokoelma ja sivustokokoelma (38). Sivustokokoelman asetukset täytyy määritellä niin, että SharePoint Framework -sovellusten käyttäminen sivustokokoelmassa on mahdollista. Tämän lisäksi SharePoint Online -palveluun on hyvä luoda myös ns. CDN-kansio SPFx-sovellusten tarvitsemien resurssitiedostojen jakelua varten.

Sovelluskokoelman luominen

SharePoint Framework -sovellusten testaaminen ja käyttö SharePoint Online -palvelussa vaatii ns. sovelluskokoelmasivuston (Application Catalog) luomisen (38). Sovelluskokoelma on SPO-palvelussa sijaitseva erillinen sivustokokoelma, jonka avulla palvelun järjestelmänvalvoja voi kontrolloida SPO-palvelussa käytettävissä olevia SharePoint Add-in- ja SharePoint Framework -sovelluksia (45, s. 103). Sovelluskokoelman kautta

SPO-palveluun lisätyt SPFx-sovellukset ovat tarvittaessa automaattisesti käytävissä kaikilla organisaation SPO-sivustoilla (27). Sovelluskokoelman luominen SPO-palveluun vaatii järjestelmänvalvojan oikeudet.

Sivustokokoelman luominen

SharePoint Framework -sovellusten testaamista varten tarvitaan sivustokokoelma. Luotavan sivustokokoelman mallipohjana voi käyttää esimerkiksi kehittäjille tarkoitettua sivustokokoelmaa (developer site collection), mutta tämä ei ole pakollista.

6.4 SharePoint Framework -sovelluksen luominen

SPFx-projektia kutsutaan nimellä solution. SharePoint Framework -projektin ja -sovelluksen luominen käynnistetään kutsumalla Yoeman SharePoint Framework -generaattoria komentokehotteesta komennolla `yo @microsoft/sharepoint`. Tämä käynnistää uuden projektin luonnin. Ensimmäisenä valitaan, minkätyyppinen SharePoint Framework -komponentti halutaan luoda (kuva 6). Kirjoitushetkellä vaihtoehtoja on kaksi: verkko-osa (web part) tai laajennus (extension). Yksittäinen SharePoint solution voi pitää sisällään yhden tai useampia SharePoint Framework -verkko-osia tai laajennuksia.'

```
C:\Users\tornjar\Desktop\SPFx\HelloEspoo>yo @microsoft/sharepoint

  --(o)--
  ( _U_ )
  |  A  |
  |  ~  |
  |  o  |
  |  Y  |

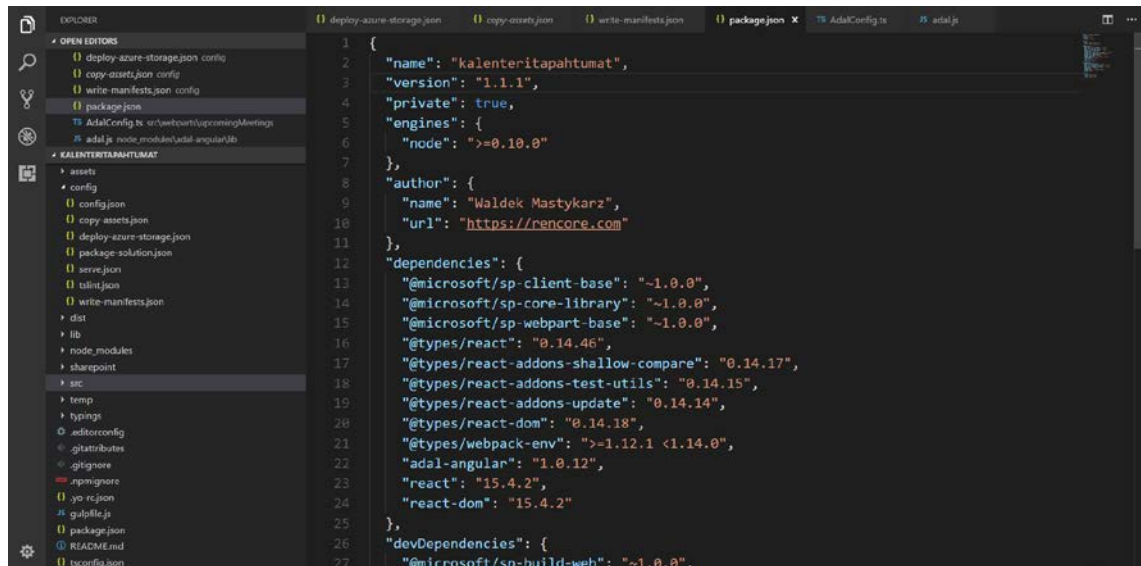
  Welcome to the
  SharePoint Client-side
  Solution Generator

? Which type of client-side component to create? (Use arrow keys)
> WebPart
  Extension (Preview)
```

Kuva 6. SharePoint Framework projektin luominen.

Seuraavaksi projektille määritellään perusasetukset, kuten nimi, kuvaus jne. Näiden lisäksi voidaan valita projektissa käytettävä JavaScript-kirjasto. Vaihtoehtoina ovat kirjoitushetkellä React tai Knockout (39). JavaScript-kirjastoja voidaan helposti lisätä myös ohjelmiston myöhemmässä kehitysvaiheessa npm-paketinhallintasovelluksen avulla. Kun SharePoint Framework -projektiin lisätään moduuli npm-paketinhallintasovelluksen avulla, npm asentaa samalla myös moduulin mahdolliset riippuvuudet eli ne toiset moduulit, joita kyseinen moduuli tarvitsee toimiakseen. Kun kaikki valinnat ja asetukset on

tehty, Yoeman-generaattori luo projektille valmiin rungon, joka sisältää tarvittavat tiedostot, kansiot ja JavaScript-moduulit (kuva 7).



Kuva 7. SPFx-projektikansiot ja -tiedostot.

6.5 Sovelluksen jakeluversion luominen

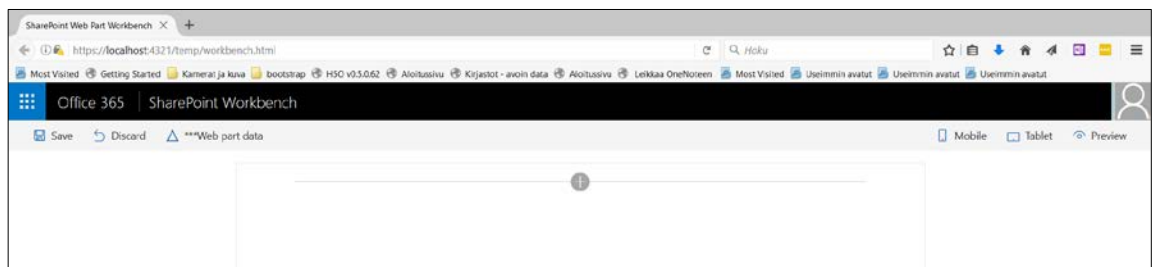
Sovelluksen jakeluversion luomiseksi tarkoitetaan tuotantoon tarkoitettua versiota sovelluksesta. Ennen sovelluksen jakeluversion luomista kannattaa tarkistaa, että gulp serve -toiminto on pysäytetty ja projektin asetukset, mm. CDN-kansion polku, Application ID -arvo jne., on varmasti asetettu oikein.

SharePoint Framework -sovelluksen jakeluversion luominen aloitetaan sovelluksen resurssitiedostojen paketoimalla. Tämä tehdään Gulp-tehtävänkäsittelijän bundle-tehtävän avulla. Bundle-tehtävä mm. kääntää TypeScript-koodin JavaScript-koodiksi ja luo lopuksi sovelluksen tarvitsemat resurssitiedostot. Bundle-komento luo oletuksena ns. debug build -pakettia, joka soveltuu ohjelmakoodin tarkasteluun. Jos sovelluksesta halutaan rakentaa loppujakelua varten tarkoitettu minimoitu ns. release build -versio, komennon tarkentimeksi lisätään --ship (47, s. 167–168). Tämä luo sovelluksen resurssitiedostoista mahdollisimman pienikokoiset versiot. Jos sovelluksen resurssitiedostoja on tarkoitus jaella Azure CDN -palvelun kautta, voidaan valmiit resurssitiedostot siirtää sinne automaattisesti komennolla *gulp deploy-azure-storage*.

Seuraavaksi luodaan itse jaeltava sovellus suorittamalla komento *gulp package-solution*. Tämä toiminto luo sharepoint/solution -kansioon spkg-päätteisen tiedoston, joka voidaan lisätä SharePoint Online -palvelun sovelluskokoelmaan (40). Sovelluspaketti ei sisällä edellä luotuja sovelluksen resurssitiedostoja.

6.6 Sovelluksen testaaminen

SharePoint Framework -sovellusta voi testata SharePoint Workbench -testiympäristössä joko lokaalisti tai SharePoint Online -palvelun avulla. SharePoint Workbench on asiakaspuolen testiympäristö, jonka avulla SPFx-verkko-osakomponentin testaaminen onnistuu ilman, että sitä tarvitsee ladata SPO-palvelimelle (39). SPFx-sovellusta on mahdollista testata SharePoint Workbench -ympäristön avulla joko paikallisesti Node.js-palvelimen avulla tai SPO-ympäristössä, ilman että sovellusta ladataan SPO-palvelimelle. Paikallinen testaaminen selaimessa vaatii ns. developer-sertifikaatin asentamista (39). SharePoint Workbench luo käynnistyessään asiakaspuolen sivun ja piirtoalueen (canvas), jonka avulla voi testata omaa SPFx-verkko-osaa (kuva 8).

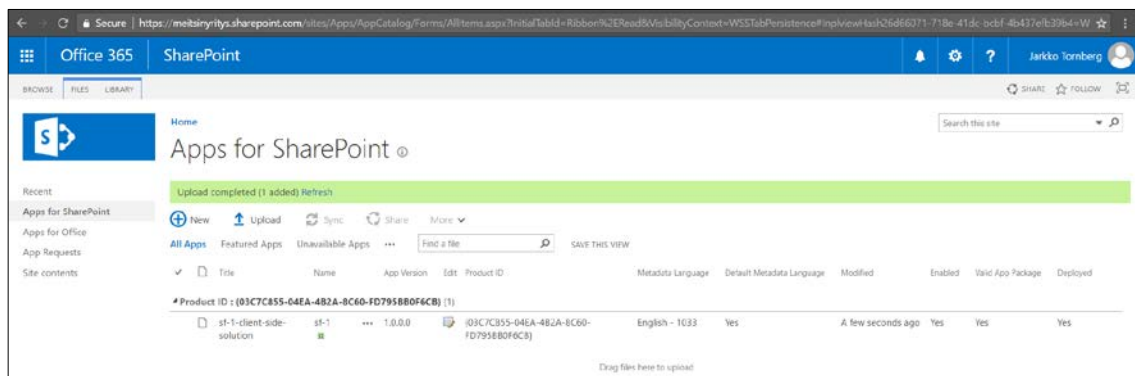


Kuva 8. SharePoint Workbench -testiympäristö

SPFx-sovellusta voi testata myös paketoimalla sovelluksen ja lataamalla valmiin sovelluksen SPO-palvelun Application catalog -kirjastoon ja ottamalla tämän jälkeen sovelluksen käyttöön SharePoint-sivustossa.

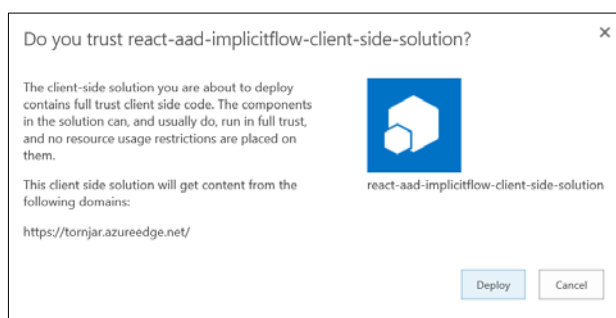
6.7 Sovelluksen asentaminen SharePoint Online -palveluun

Sovellus asennetaan lisäämällä sovelluspaketti SharePoint-sovelluskokoelma sivustolla (kuva 9). Jos sovellusta ei määritellä asentamaan globaalisti koko organisaatioon, täytyy sovellus lisätä yksitellen myös yksittäiselle SharePoint-sivustolle add app -toiminnolla, ennen kuin sovellusta voidaan asentaa yksittäiselle SharePoint-sivulle.



Kuva 9. Valmiin SPFX-sovelluksen lisääminen sovelluskokoelmaan.

Sovellusta lisättäessä palvelu pyytää vahvistuksen sovelluksen jakeluun (kuva 10).



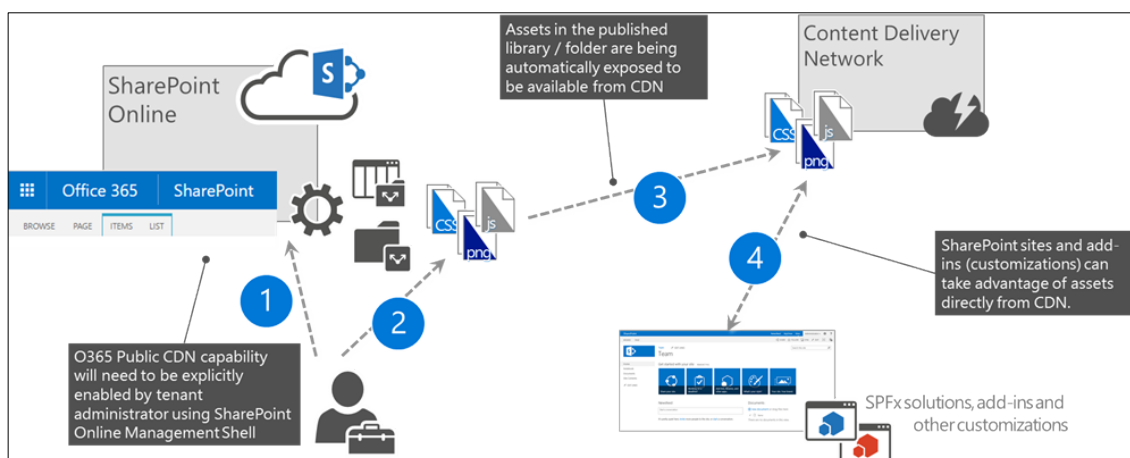
Kuva 10. Sovelluksen jakelun hyväksyminen.

6.8 Resurssitiedostojen jakelu

SPFx-sovellus tarvitsee toimiakseen ns. resurssitiedostoja (assets). Ne pitävät sisällään mm. sovelluksen tarvitsemat JavaScript-tiedostot, kuvat jne. Resurssitiedostot täytyy tallentaa sellaiseen sijaintiin, josta sovellus voi päästä niihin käsiksi. SPFx-sovelluksen käyttämien resurssitiedostojen tallennukseen ja jakeluun voidaan käyttää ns. sisällönjakeluverkkoa (Content Delivery Network, CDN). Pathan, Buyya ja Vakali määrittelevät sisällönjakeluverkon palveluksi, joka koostuu verkkopalvelimista ja välityspalvelimista, joiden kautta käyttäjän pyytämä sisältö pyritään jakelemaan mahdollisimman nopeasti ja tehokkaasti. (49, s. 16–17). Sisällönjakeluverkosto voidaan toteuttaa esim. Azure-pilvipalvelun tai SharePoint Online -palvelun avulla.

SharePoint CDN-kansion käyttäminen

Office 365 -palvelu tukee ns. julkisen CDN-kansion (public CDN) luomista Office 365-palveluun (41). CDN-kansio luodaan halutun SharePoint Online -sivuston Site Assets -dokumenttikirjastoon, joka määritellään PowerShell-komentojen avulla julkiseksi CDN-kansioksi (kuva 11) (22, s. 179–180).



Kuva 11. Office 365 -palvelun julkisen CDN -kansion käyttö (41).

Azure CDN-palvelun käyttäminen

CDN-palvelu voidaan määrittellä myös Microsoftin Azure-pilvipalveluun. Tämä vaatii olemassa olevan Azure-tilin, palvelun ylläpitäjän oikeudet ja seuraavat toimenpiteet:

- Azure Storage Account -tilin luomisen
- Blob Container -säiliön luomisen
- Storage Account access key avaimen luomisen
- CDN-profiilin ja ns. endpointin luomisen.

Tämän jälkeen SPFx-projektiin voidaan `deploy-azure-storage.json` -tiedostoon tehdä määrittelyt Azure-palvelun käyttämiseksi CDN-kirjastona. Määrittelyissä tarvitaan käytetyn Azure CDN -tilin ja Blob container -säiliön nimet sekä Storage account access key -arvo (40). Määrittelyn jälkeen sovellusta paketoitaessa Gulp-tehtävä kopioi automaattisesti sovelluksen tarvitsemat resurssitiedostot määriteltyyn Azure CDN -kansioon.

CDN-basepath -asetus

CDN-kansion määrittelyn jälkeen SPFx-projektin `write-manifest.json` -tiedoston `cdnBasePath`-attribuuttiin täytyy määrittellä `cdnBasePath` -arvo. Käytettäessä SharePoint CDN -kansiota määrittelyyn tarvitaan määritellyn CDN-kansion origin ID -attribuutin arvo, joka saadaan PowerShell-komennon `Get-SPOPublicCdnOrigins` avulla. Käytettäessä Azure CDN -palvelua `cdnBasePath`-attribuutin arvoksi annetaan määritelty Azure CDN endpoint -osoite (40).

6.9 Sovellusten hyödyntäminen organisaatiossa

SharePoint-sivustoja ja SPFx-sovelluksia voidaan hyödyntää organisaatiossa monella eri tavalla. Perinteisen intranetratkaisun lisäksi SharePoint Online -palvelun asetukset tarjoavat mahdollisuuden sallia SharePoint tiimi- ja yhteistyösivujen luomisen myös palvelun peruskäyttäjille (42). Myös jokainen palveluun luotu uusi Office 365 -ryhmä tai uusi tiimi saa automaattisesti oman SharePoint-tiimisivunsa.

Yksittäinen käyttäjä organisaatiossa voi kuulua useaan Office 365 -ryhmään. Jokaisella Office 365 -ryhmällä on automaattisesti oma SharePoint-tiimisivusto (43). Ryhmän omistajat voivat lisätä tiimisivustolle organisaation sovelluskokoelmaan (application catalog) lisättyjä SPFx-sovelluksia. Näin organisaation käytössä olevia SPFx-sovelluksia voidaan käyttää useassa eri kohdesivustossa SharePoint Online -palvelun sisällä.

7 Intranetpalvelun personointi SharePoint Framework -sovellusten avulla

Intranet-palvelu voidaan personoida joko käyttäjä- tai käyttäjäryhmäkohtaisesti (6). Amy Schade suosittelee sekä intranetin toiminnallisuuksien että sen sisällön personointia (44). Tässä insinööriyössä keskitytään ainoastaan intranetin käyttäjäkohtaiseen sisällön personointiin. Sisällön personointi voi tarkoittaa SharePoint intranet -palvelussa esimerkiksi käyttäjäkohtaisen kalanteritiedon noutamista Office 365 -palvelusta.

SharePoint Online -pohjainen intranetratkaisu mahdollistaa organisaation Office 365 -palvelussa syntyneen henkilökohtaisen datan hyödyntämisen Microsoftin tarjoamien rajapintojen kautta. SPFx-sovellus voi rajapinnan kautta hyödyntää sekä muissa Office 365 -sovelluksissa että SharePoint Online -palvelussa syntyvää käyttäjä- tai käyttäjäryhmäkohtaista dataa. Tämän lisäksi SPFx-sovellus voi periaatteessa noutaa dataa kolmannen osapuoleen palvelusta, joka tarjoaa ulkopuolisten sovellusten käyttöön esimerkiksi REST-tyyppisen rajapinnan.

7.1 Office 365 -palveluiden hyödyntäminen personoinnissa

Office 365 -palveluita ja niiden tuottamaa dataa voidaan hyödyntää Microsoftin tarjoamien rajapintojen kautta. O365-palvelun rajapinnat on perinteisesti jaettu Office 365 REST API -rajapintoihin ja SharePoint REST API -rajapintaan. Näiden lisäksi Microsoft tarjoaa uuden Graph API -rajapinnan Office 365 -palvelussa syntyvän datan hyödyntämiseen. Office 365 API on kokoelma useita eri rajapintoja. Lähes jokainen erillinen O365-palvelu, kuten esimerkiksi Exchange Online, tarjoaa oman rajapintansa (Outlook API) palvelun hyödyntämiseksi ulkopuolisessa järjestelmässä (45). SharePoint REST API -rajapinta on puolestaan tarkoitettu SharePoint-sivustojen sisältämän datan, kuten kirjas-tojen ja listojen, hyödyntämiseen. Uusi Graph API tarjoaa yhden ”oven”, jonka kautta on pääsy kaikkien Office 365 -palveluiden synnyttämään dataan.

Jotta sovellus voisi käyttää Office 365 -palvelussa olevaa dataa rajapintojen kautta, täytyy sovelluksen käyttäjän tai järjestelmänvalvojan antaa sovellukselle lupa käyttää valitua rajapintaa datan lukemiseen. Jotta lupa voitaisiin myöntää, sovellus täytyy tunnistaa eli autentikoida. Tämän lisäksi sovelluksen käyttäjällä on oltava lupa käyttää kyseistä sovellusta. Tämä puolestaan tarkoittaa, että sovelluksen käyttäjä täytyy myös tunnistaa eli autentikoida. Näiden vaatimusten toteuttamiseen tarvitaan Azure Active Directory -palvelua ja sen autentikointi- ja auktorisointipalveluita.

7.2 Hakemisto-, autentikointi- ja auktorisointipalvelut

Azure Active Directory (AAD) on hakemistopalvelu, jonka avulla voidaan hallinnoida mm. käyttäjiä, laitteita ja sovelluksia, eri palveluihin kirjautumista, käyttöoikeuksia ja resursseja. SharePoint Online- ja Office 365 -palvelut käyttävät AAD-hakemistopalvelua sekä käyttäjien että sovellusten autentikointiin (46). Tämän lisäksi AAD-hakemistopalvelua voidaan käyttää myös esimerkiksi SPFx-sovellusten rekisteröintiin ja sovelluksen käyttämien rajapintojen kautta tapahtuvaan resurssien auktorisointiin. Office 365- ja SharePoint-palveluissa käyttäjän identiteetti varmennetaan kyseisen organisaation Azure AD -palvelun avulla. Käytännössä tämä tarkoittaa, että SharePoint Framework -sovelluksen käyttäjä on yleensä jo autentikoitunut oman organisaation AAD-palvelua vastaan.

Jotta SPFx-sovellus voisi päästä Office 365 -rajapinnan kautta käyttämään suojattua resurssia, esimerkiksi lukemaan käyttäjän sähköpostia, sovelluksen täytyy saada tähän lupa resurssinhaltijalta, joka yleensä on myös sovelluksen sen hetkinen käyttäjä. SPFx-sovellus ei yleensä pyydä tätä lupaa suoraan resurssinhaltijalta eli käyttäjältä, vaan käyttää käyttöoikeuden pyytämiseen ns. auktorisointipalvelinta, joka myöntää käyttöoikeuden haluttuun resurssiin käyttäjän puolesta. AAD-palvelu käyttää palveluun rekisteröityjen ulkoisten sovellusten, kuten SPFx-sovellusten tarvitsemien resurssien auktorisointiin OAuth 2.0 -auktorisointikehystä (authorization framework) (12, s. 95). Sitä voidaan käyttää periaatteessa minkä tahansa HTTP-pohjaisen resurssin, esimerkiksi sovelluksen tai dokumentin, auktorisointiin mistä tahansa asiakassovelluksesta.

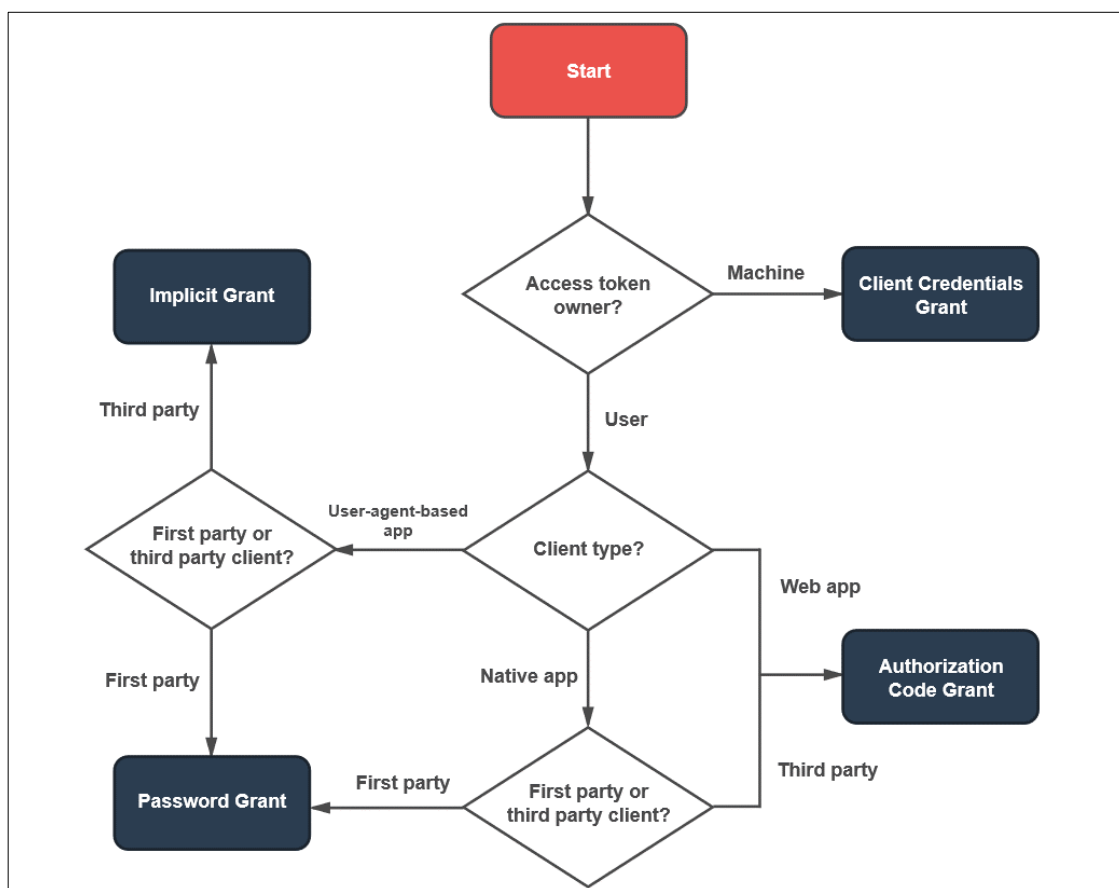
OAuth 2.0 -kehystä ei voida suoraan käyttää käyttäjän tai sovelluksen autentikointiin. Azure AD -palvelussa käyttäjän ja myös sovellusten autentikointiin käytetään OAuth 2.0 -kehysten kanssa OpenID Connect (OIDC) -protokollaa. OpenID Connect on avoimeen standardiin pohjaava autentikointiprotokolla, joka on rakennettu OAuth 2.0 -kehysten päälle huolehtimaan käyttäjän autentikoinnista (12, s. 110). Azure AD -palvelussa käytettäessä SharePoint palvelua ja SharePoint Framework sovelluksia OpenID Connect suorittaa mm. käyttäjän autentikoinnin ja OAuth 2.0 -protokolla huolehtii autentikoinnin jälkeen tarvittavien resurssien käyttöoikeuksien hallinnasta eli auktorisoinnista.

7.3 Auktorisointi OAuth 2.0 -kehysten avulla

Päästäkseen käyttämään haluamaansa resurssia OAuth 2.0 -kehysten avulla, SPFx-sovellus tarvitsee OAuth 2.0 -kehykseltä ns. access token -poletin. Sen avulla sovellus voi pyytää rajapinnalta käyttöoikeutta haluttuun resurssiin resurssinhaltijan eli yleensä

sovelluksen käyttäjän puolesta. Access token -polettia voidaan verrata esimerkiksi pääsylippuun, jonka avulla pääsylipun haltija on oikeutettu pääsemään sisään konserttiin. Samoin kuin pääsylippu, access token -poletti ei varsinaisesti sisällä tietoa pääsylipun haltijasta, ainoastaan tietoa siitä, mihin konserttiin ja mille istumapaikalle pääsylipun haltija on oikeutettu. Access token -poletti ei siis sisällä käyttäjän tunnistetietoja, kuten käyttäjätunnusta tai salasanaa. Se on tämän vuoksi tietoturvasyistä yleensä sidottu tiettyyn asiakassovellukseen.

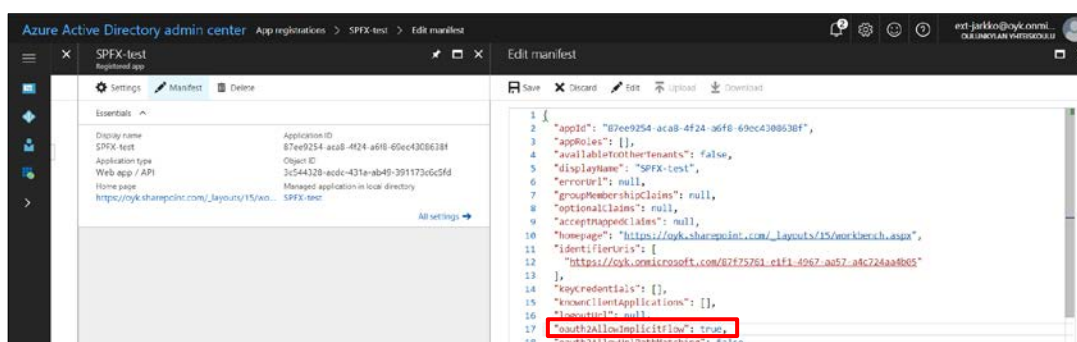
OAuth 2.0 -protokolla määrittelee neljä erilaista menetelmää, joilla asiakassovellus voi saada access token -poletin eli käytännössä käyttöoikeuden suojattuun resurssiin. Menetelmät ovat authorization code grant, implicit grant, resource owner password credentials grant ja client credentials grant. Valittava menetelmä määräytyy asiakassovelluksen ja sen ominaisuuksien perusteella kuvan 12 kaavion mukaan.



Kuva 62. OAuth 2.0- authoiointimenetelmät (76).

Implicit grant -menetelmä

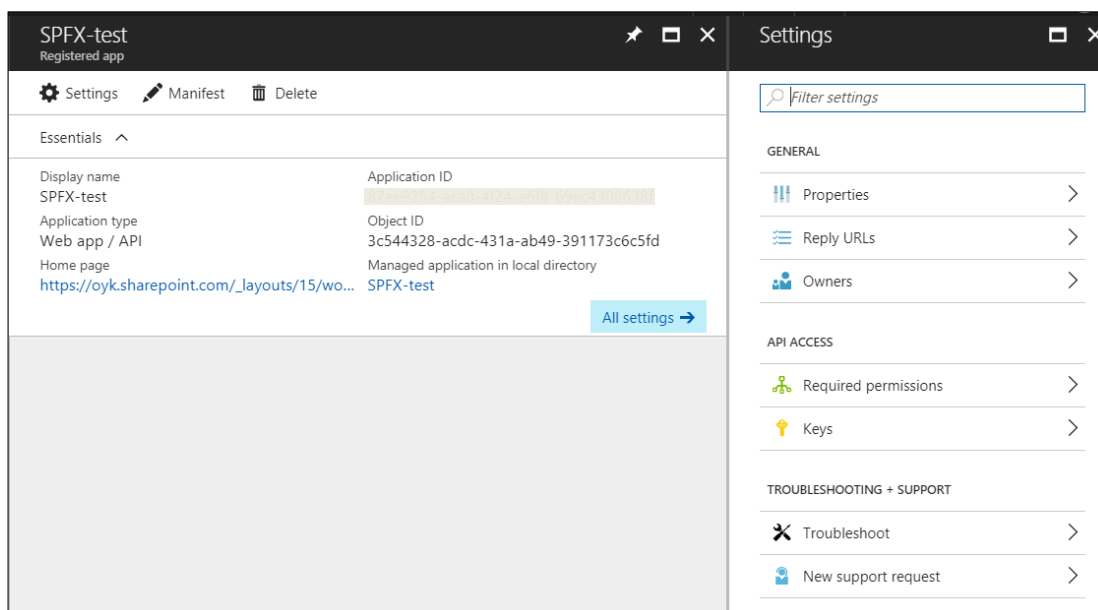
Implicit grant -menetelmää käytetään tapauksissa, joissa OAuth 2.0 -kehysten asiakkaana on julkinen, kolmannen osapuolen sovellus, joka suoritetaan yleensä selainympäristössä (esim. SPFX-sovellus). Tällaisissa tapauksissa asiakassovellus ei pysty tallentamaan tietoturvallisesti access token -polettia tai ns. client secret -koodia (56, s. 64). Implicit grant -menetelmässä access token -poletti voi päätyä käyttäjäagentin (selain) kautta myös muiden sovellusten tietoon. Koska poletti on ainoa, mitä tietyn suojatun resurssin käyttöön tarvitaan, tämä muodostaa tietoturvauhan! Tämän vuoksi rekisteröitäessä sovellusta Azure AD -palveluun sovellus täytyy erikseen määritellä käyttämään implicit grant -menetelmää (kuva 13).



Kuva 13. Implicit grant -menetelmän salliminen Azure AD -palvelussa.

7.4 Sovelluksen rekisteröiminen Azure AD -palveluun

Käyttäjätietojen lukeminen Microsoftin tarjoamien rajapintojen kautta on mahdollista toteuttaa kahdella tavalla. Ensimmäinen ja monipuolisempi vaihtoehto on rekisteröidä sovellus Azure AD -palveluun ja antaa palvelun kautta sovellukselle käyttöoikeudet käyttäjän dataan valitun rajapinnan kautta. OAuth 2.0 -protokolla määrittää vaatii asiakassovelluksen rekisteröimisen silloin, kun kyseessä on ns. julkinen asiakassovellus (public client) ja/tai kun asiakassovellus käyttää auktorisointiin implicit grant -menetelmää (47). Jos SPFX-sovellus haluaa käyttää Azure AD -palvelun OAuth 2.0 -kehystä, sovellus täytyy rekisteröidä AAD-palveluun. Rekisteröinnin yhteydessä sovellukselle määritellään sovellustyyppi ja nimi. AAD luo sovellukselle uniikin application ID -tunnuksen, jota sovellus käyttää asioidessaan OAuth 2.0 -kehysten kanssa (kuva 14). Sovellukselle rekisteröinnin yhteydessä annettu application ID -tunnus täytyy liittää SPFX-sovelluksen koodiin.



Kuva 14. Sovelluksen rekisteröinti Azure AD -palvelussa.

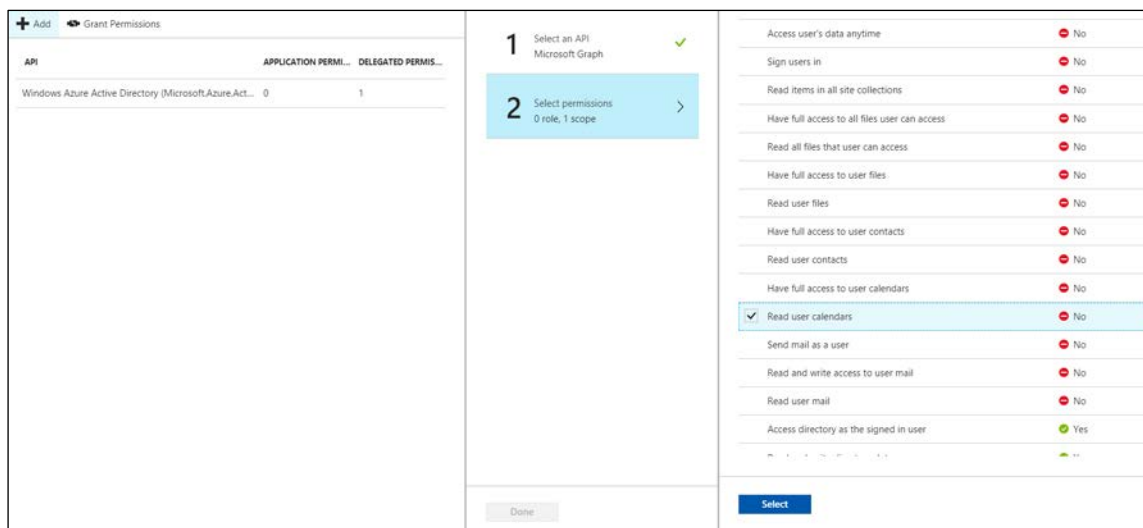
Application ID -tunnuksen lisäksi SPFX-sovellukselle ja muille julkisille verkkosovelluksille täytyy OAuth 2.0 -protokollamääritysten mukaan määrittellä myös ns. redirect URI (joskus Reply URI) -osoite, jota käytetään silloin, kun auktorisointipalvelin kommunikoi sovelluksen kanssa (47). Redirect URI -osoite on sovelluksen verkko-osoite, josta sovellus kutsuu rajapintaa. Sen määrittelyyn voidaan lisätä useampia URI-osoitteita ja osoitteissa voidaan käyttää ns. wildcard-parametreja. Nämä vaihtoehdot mahdollistavat yhteen Azure AD -palveluun rekisteröidyn SPFX-sovelluksen käyttämisen useilla saman organisaation SharePoint-sivustoilla ja saman sovelluksen käytön myös usean eri organisaation SharePoint-sivustoilla (ns. multitenant-sovellus).

GraphHttpClient-kirjasto

Yksi vaihtoehto sovelluksen rekisteröinnille on käyttää esituotantovaiheessa olevaa Microsoftin GraphHttpClient-kirjastoa, jolloin sovellusta ei tarvitse rekisteröidä Azure AD -palveluun. Käytettäessä GraphHttpClient-kirjastoa SPFX-sovellukselle voi olla vain hyvin rajoitetut oikeudet Graph API -rajapinnan käyttöön. Kirjoitushetkellä GraphHttpClient tukee vain kahta erityyppistä käyttöoikeusmäärittystä Graph API -rajapintaan, Read and write all groups ja Read all usage reports (48). Mainittujen rajoitusten ja esituotantoversioluonteen vuoksi GraphHttpClient-kirjastoa ei tällä hetkellä voi suositella tuotantokäyttöön.

7.5 Käyttöoikeuksien myöntäminen sovellukselle

Rekisteröinnin jälkeen sovellukselle voidaan määrittellä käyttöoikeudet haluttuun rajapintaan. Sovellukselle kannattaa tietoturvasyistä määrittellä vain ne oikeudet, joita sovellus todellisuudessa tarvitsee. Azure AD -palvelussa sovellukselle voidaan määrittellä käyttöoikeudet esimerkiksi Graph API -rajapintaa. Sovelluksen oikeudet voivat sisältää esimerkiksi oikeuden lukea käyttäjän sähköpostia (kuva 15).

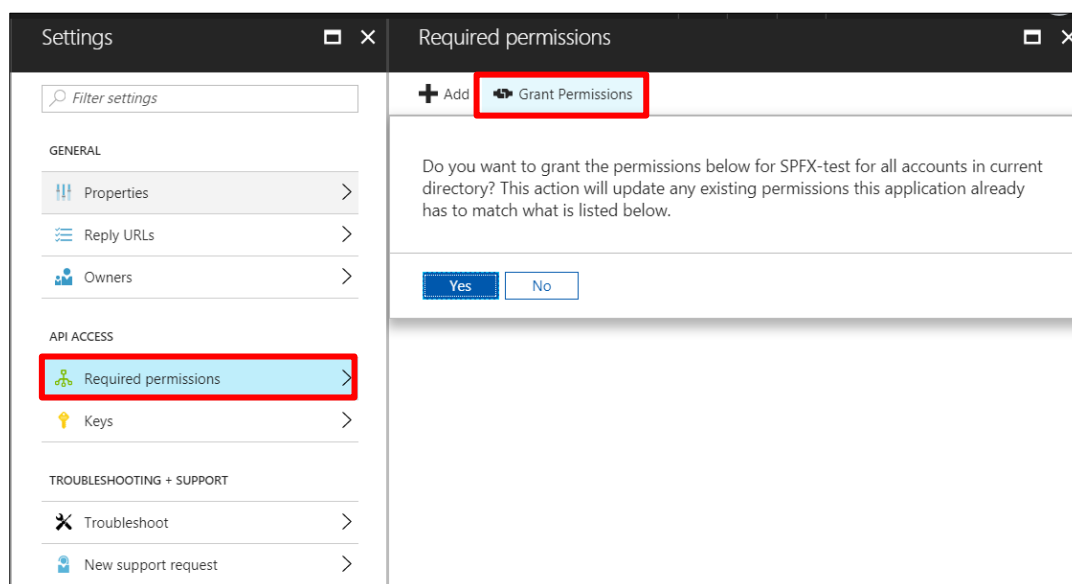


Kuva 15. Sovelluksen käyttöoikeuksien määrittäminen Azure AD -palvelussa.

Käyttöoikeudet toimitetaan Azure AD -palvelussa asiakassovellukselle access token -pöletin avulla OAuth 2.0 -auktorisointikehyksen välityksellä. Jotta SPFX-sovellus tai jokin muu palvelu tai sovellus pystyisi hyödyntämään Azure AD -palvelun rajapintoja, sovelluksen on pystyttävä:

- käyttämään Azure AD -palvelun OAuth 2.0- ja Open ID Connect -kehyksiä käyttäjän ja/tai sovelluksen autentikointiin ja auktorisointiin
- lähettämään perinteinen HTTP-pyyntö valitulle O365-palvelun rajapinnalle
- vastaanottamaan ja hyödyntämään O365-palvelun palauttamaa JSON- tai XML-muotoista tietoa.

Azure AD -palvelun käyttäjille täytyy myös määrittellä käyttöoikeus rekisteröityyn sovellukseen. Tämä tehdään Azure AD -palvelussa sovelluksen asetuksissa Grant permission -toiminnolla (kuva 16).



Kuva 16. Sovelluksen käyttöoikeuden myöntäminen käyttäjille.

7.6 Rajapintojen hyödyntäminen asiakassovelluksessa

Yksi suurimmista haasteista käyttäjätiedon hyödyntämisessä intranetin personoinnissa liittyy juuri rajapintojen käyttöön. Jotta rajapintojen autentikointi- ja auktorisointikehysten käyttö ei muuttuisi sovelluskehittäjälle ylivoimaiseksi haasteeksi, Microsoft tarjoaa sovelluskehittäjille valmiita ohjelmakirjastoja rajapintojen hyödyntämiseen. Kirjastot auttavat sovelluksen autentikoinnin ja auktorisoinnin teknisessä toteuttamisessa.

Myös SPFX-sovellukset voivat hyödyntää Microsoftin tarjoamia kirjastoja. SPFX-sovellukset voivat käyttää kahta eri kirjastoa asioidessaan AAD-palvelun OAuth 2.0 -rajapinnan kanssa. Microsoft tarjoaa Azure AD -palvelun kanssa käytettäväksi lähtökohtaisesti ADAL.js-JavaScript-kirjastoa. Tämän lisäksi tarjolla on uuden Azure AD v2.0 endpointin kanssa käytettäväksi tarkoitettu Microsoft Authentication Library (MSAL) -kirjasto.

ADAL.js

ADAL.js on Microsoftin tarjoama JavaScript-kirjasto, joka on tehty helpottamaan SPFX-sovellusten autentikointia ja resurssien auktorisointia Azure AD -hakemistopalvelua vastaan (49). ADAL.js-kirjasto on alun perin rakennettu toimimaan Angular.js-kirjaston kanssa, mutta ADAL.js-kirjastoa voidaan käyttää myös ilman Angular.js-kirjastoa.

MSAL

MSAL on Microsoftin uusi Azure AD V 2.0 -rajapinnan hyödyntämistä varten luotu uusi kirjasto. Azure AD V2.0 ei ole taaksepäin yhteensopiva vanhan V1.0 -rajapinnan kanssa.

Tämä tarkoittaa käytännössä sitä, että uuden rajapinnan kanssa käytettävät ”komennot” (syntaksi) ovat erilaisia kuin vanhan rajapinnan kanssa. Tämän vuoksi v2.0 tuo mukanaan myös uuden kirjaston rajapinnan käyttöön.

GraphHttpClient

Jos sovellusta ei rekisteröidä Azure AD -palveluun on Graph API -rajapintaa mahdollista käyttää ”suoraan”, kuten mitä tahansa REST-rajapintaa Microsoftin oman GraphHttpClient-kirjaston kautta. GraphHttpClient-kirjaston toiminnallisuudet ovat vielä rajoitettuja, eikä se tue kuin pientä osaa Graph API -rajapinnan ominaisuuksista. Taulukossa 1 on vertailtu autentikointiin ja auktorisointiin tarvittaviin JavaScript-kirjastojen ominaisuuksia.

Taulukko 1. Autentikointiin ja auktorisointiin tarkoitettujen JavaScript-kirjastojen vertailu

	ADAL	MSAL	GraphHttpClient
Sovelluksen rekisteröinti	Azure AD	Azure AD V2.0	ei vaadi rekisteröintiä
Mahdolliset käyttäjät	Azure AD	MS-tili & Azure AD	
Sovelluksen autentikointi	Azure AD	Azure AD v2.0	ei vaadi autentikointia
Sovelluksen auktorisointi	OAuth 2.0 implicit flow	OAuth 2.0 implicit flow	

7.7 Office 365 -palvelussa käytettävissä olevat rajapinnat

Sharepoint-rajapinnat

SharePoint tarjoaa sekä vanhan SOAP- että uudemman REST-tyyppisen rajapinnan SharePoint-palvelun datan hyödyntämiseen ulkopuolisissa palveluissa tai sovelluksissa. SharePoint online -palvelussa olevaa dataa voidaan hyödyntää SPFx-sovelluksessa periaatteessa neljällä eri tavalla, joko käyttämällä vanhaa SOAP-rajapintaa ja JSOM-kirjastoa, käyttämällä REST-rajapintaa ilman ulkopuolista kirjastoa tai käyttämällä REST-rajapintaa joko Microsoftin SPHttpClient-kirjaston tai PnP JS Core -kirjaston avulla.

SPHttpClient -kirjasto

Microsoft tarjoaa SharePoint REST -rajapinnan hyödyntämiseksi SPHttpClient-kirjastoa. SPHttpClient mm. lisää automaattisesti HTTP-kutsuihin tarvittavat header-kentät (50).

PnP JS Core -kirjasto

PnP JS Core -kirjasto on SharePoint REST -rajapinnan hyödyntämiseen luotu kirjasto, jonka avulla perusoperaatioiden suorittaminen on suoraviivaisempaa (51).

Kaikkia mainittuja tapoja voidaan käyttää SPFx-sovelluksissa noudettaessa SharePoint Online -palvelusta käyttäjä- tai käyttäjäryhmäkohtaista dataa. Vaihtoehdot eroavat toisistaan mm. ominaisuuksien ja käytön helppouden osalta (kuva 17). Luettava ja muokattava data voi pitää sisällään esimerkiksi käyttäjän SharePoint-tiimisivustolla olevalla SharePoint listalla olevia tietueita.

	J\$OM	Raw REST	SPHttpClient	PnP JS Core
Characteristics	<ul style="list-style-type: none"> • Introduced in SP2010 • Mimics CSOM • Works with delegates • Does not support promises • Not being invested in anymore 	<ul style="list-style-type: none"> • The same across technology stacks • Easy for GET requests • Complex for non-GET requests • Evergreen 	<ul style="list-style-type: none"> • SharePoint Framework-specific • Easy for all kinds of requests • For non Get operations, you'll still need to create JSON • Evergreen 	<ul style="list-style-type: none"> • Fluent API • Works natively with Promises • Open-source community-driven initiative
Future-proof	!	✓	✓	✓
Coverage	Fair	Excellent	Excellent	Good
Ease of use	Hard	Moderate	Moderate+	Easy

Kuva 17. SharePoint rajapintojen eri käyttötavat SPFx-sovelluksissa (77).

Office 365 -rajapinnat

Office 365 API -rajapinta sisältää useita yksittäisiä rajapintoja, joita voidaan hyödyntää myös SPFx-sovellusten kanssa. Office 365 API -rajapinnan kautta SharePoint-sivustolla voidaan näyttää tietoja esimerkiksi käyttäjän Outlook-sähköpostista tai -kalenterista. Office 365 REST API sisältää omat rajapinnat esim. OneDrive-tiedostokirjaston tai Outlook-sähköpostisovelluksen hyödyntämiseen.

Graph API -rajapinta

Office 365 ja SharePoint Online -palveluja yhdistää yksi yhteinen rajapinta, Graph API. Se on kokoelma Office 365 -palvelun tarjoamia rajapintoja, jotka on julkaistu yhden REST-endpoint -osoitteen kautta. Rajapinnasta on saatavilla kaksi eri versiota, v1.0 ja beta. Beta-versio tarjoaa testausvaiheessa olevia toiminnallisuuksia, eikä niitä Microsoftin mukaan tulisi käyttää tuotantoympäristössä (52). Graph API -rajapinta mahdollistaa kaikkien O365-palveluiden hyödyntämisen käyttäen yhtä endpointiä. Yhden ainoan rajapinnan käyttö helpottaa sovellusten kommunikointia Office 365 -palvelujen kanssa. Graph API ei vielä tue lähestulkoonkaan kaikkia Office 365 API- ja SharePoint API -rajapintojen sisältämiä toiminnallisuuksia.

7.8 Haasteet rajapintojen käytössä

SharePoint-sivustolla voi teoriassa olla useita SPFx-sovelluksia, jotka käyttävät samaa rajapintaa. OAuth 2.0 -protokolla antaa SPFx-sovellukselle oikeudet access token -poletin kautta. Access token -poletissa on määritelty, mitä oikeuksia, esim. oikeus lukea käyttäjän sähköpostia Graph API -rajapinnan kautta, poletin haltijalla on. Käytettäessä ADAL.js-kirjastoa selaimessa voi olla sivustokohtaisesti vain yksi voimassa oleva access token -poletti rajapintaa kohden (46). Käytännössä tämä tarkoittaa, että käytettäessä SPFx-sovelluksessa ADAL.js-kirjastoa ensimmäinen noudettu access token -poletti määrittelee myös muiden samalla sivustolla olevien, samaa rajapintaa käyttävien sovellusten oikeudet. Tämä luo merkittävän haasteen usean SPFx-sovelluksen käytölle SharePoint-verkkosivustolla.

8 Loppupäätelmät

Personoidun intranetpalvelun rakentaminen SharePoint Online -palveluun hyödyntämällä SharePoint Framework -sovelluskehitystä sisältää vielä haasteita teknologian nuoren iän vuoksi. Yhtenä suurena haasteena voi pitää modernien SharePoint-sivujen käytön vaikeutta sivujen mallipohjina luotaessa sivustokokoelmia. Sivustokokoelmia luotaessa käytössä olevat mallipohjat edustavat kaikki klassista käyttökokemusta. Toinen haaste liittyy Office 365 -palvelun rajapintojen auktorisointiin käytettävissä olevien JavaScript-kirjastojen avulla. Verkkoselain voi sivustokohtaisesti hallinnoida kerrallaan vain yhtä access token -polettia rajapintaa kohden. Käytännössä tämä tarkoittaa sitä, että samalla SharePoint-sivustolla ei voi tällä hetkellä olla useaa SPFx-sovellusta, jotka käyttävät samaa rajapintaa eri käyttöoikeuksilla. Seuraavassa on luokiteltuna SharePoint Framework -tekniikan tämän hetken haasteita ja mahdollisuuksia.

Tämän hetken haasteet:

- moderniin käyttökokemuksen perustavan Sharepoint-sivustokokoelman luominen haastavaa
- SPFx-sovellusten automaattinen liittäminen SharePoint-sivustolle vaikeaa
- käytettävän rajapinnan valinnan vaikeus: Graph API vai erilliset sovelluskohtaiset rajapinnat (Office 365 API ja SharePoint API)
- Graph API -rajapinnan osittain rajalliset ominaisuudet
- rajapintojen käytön rajoitus: vain yksi access token sivustoa ja rajapintaa kohden
- sovelluskehityksessä käytettyjen moduulien versionhallinnan haastavuus
- SPFx-sovellusten rekisteröiminen: Azure AD V1 vai V2

Mahdollisuudet:

- mahdollisuus hyödyntää suhteellisen helposti käyttäjän muissa Office 365 -palveluissa synnyttämää dataa
- SPFx-sovellusten rakentaminen mahdollista ilmaisilla, avoimen lähdekoodin työkaluilla
- mahdollisuus SPFx-sovellusten keskitettyyn jakeluun Azure-pilvipalvelun kautta useammalle toimijalle (multitenant-tyyppiset sovellukset)
- SPFx-sovellusten käyttäjien autentikointi kohtuullisen vaivatonta Azure AD -palvelun kautta käyttämällä Microsoftin tarjoamia valmiita JavaScript-kirjastoja
- sovelluskehittäjillä käytössä Office Fabric UI -sovelluksen ulkoasun muokkaamiseen

SharePoint Framework on uusi ja kehittyvä teknologia. Tämän vuoksi monet kirjoitushetkellä olevat haasteet todennäköisesti ratkeavat ajan myötä. Lokakuun lopussa Microsoftin Ignite 2017 -konferenssista julkaistujen tietojen perusteella näyttää siltä, että modernien SharePoint-sivujen käyttö sivujen mallipohjina luotaessa sivustokokoelmia on pian mahdollista. Microsoft on myös julkaissut ratkaisun, jonka avulla SharePoint Framework -sovelluksia voidaan automaattisesti lisätä SharePoint-palvelussa luoduille moderneille SharePoint-sivustoille. Microsoft on julkistanut myös tiedon GraphHttpClient-kirjaston päivityksestä, jonka avulla rajapintojen käyttöön ja sovellusten auktorisointiin liittyvä ongelma poistuu. GraphHttpClient-kirjaston päivityksen jälkeen yhdellä SharePoint-sivustolla voi olla useita, samaa rajapintaa käyttäviä, mutta eri käyttöoikeuksia tarvitsevia SPFx-sovelluksia. Nämä ominaisuudet yhdessä helpottavat personoidun, responsiivisen, SPFx-sovelluksia sisältävän SharePoint Online -pohjaisen intranetratkaisun toteuttamista.

Lähteet

- 1 Edera Easy. 2017. Verkkoaineisto. Edera Partners Oy. <<https://www.edera.fi/edera-easy-intranet>>. Luettu 5.3.2017.
- 2 Käsitteet - Intranet. 2017. Verkkoaineisto. Tilastokeskus <<http://www.stat.fi/meta/kas/intranet.html>>. Luettu 13.9.2017.
- 3 Pernice, Kara & Caya, Patty. 2014. Intranet Portals are the Hub of the Enterprise Universe. Verkkoaineisto. Nielsen Norman Group. <<https://www.nngroup.com/articles/intranet-portals/?lm=10-best-intranets-2016&pt=report>>. Luettu 13.9.2017.
- 4 Intranet Information Architecture (IA) Trends. 2014. Verkkoaineisto. Nielsen Norman Group <<https://www.nngroup.com/articles/intranet-information-architecture-ia/?lm=10-best-intranets-2016&pt=report>>. Luettu 13.9.2017.
- 5 2017 Intranet Design Annual Winners. 2017. Verkkoaineisto. Nielsen Norman Group. <<https://www.nngroup.com/news/item/2017-intranet-design-awards/>>. Luettu 1.10.2017.
- 6 Schade, Amy. 2016. Customization vs. Personalization in the User Experience. Verkkoaineisto. Nielsen Norman Group. <<https://www.nngroup.com/articles/customization-personalization/>>. Luettu 10.11.2017.
- 7 The personalized intranet. 2016. Verkkoaineisto. Digital Workplace Group. <<https://digitalworkplacegroup.com/wp-content/downloads/dwg-free/DWG-Personalization-and-Customization-Exec-Summary.pdf>>. Luettu 8.11.2017.
- 8 Digital Workplace Key Initiative Overview. 2014. Verkkoaineisto. Gartner. <<https://www.gartner.com/doc/2715219/digital-workplace-key-initiative-overview>>. Luettu 5.11.2017.
- 9 Schillerwein, Stephan. 2013. Digital Workplace Framework. Verkkoaineisto. <<https://de.slideshare.net/IntranetMatters/digital-workplace-framework>>. Luettu 8.11.2017.
- 10 Shehadi, Ramez & Karam, Danny. 2014. Five Essential Elements Of The Digital Workplace. Verkkoaineisto. Forbes. <<https://www.forbes.com/sites/boozandcompany/2014/03/31/five-essential-elements-of-the-digital-workplace/#19b9ba246258>>. Luettu 5.11.2017.
- 11 The digital workplace: Think, share, do. Verkkoaineisto. Deloitte. <www2.deloitte.com/content/dam/Deloitte/mx/Documents/human-capital/The_digital_workplace.pdf>. Luettu 4.11.2017.
- 12 PIALORSI, PAOLO. 2016. Programming Microsoft Office 365. Redmond, Washington: Microsoft Press.
- 13 Microsoft Office 365 now has 120 million Business users. 2017. Verkkoaineisto. manage-office.com. <<http://manage-office.com/microsoft-office-365-now-120-million-business-users/>>. Luettu 5.11.2017.
- 14 The Future of SharePoint. 2016. Verkkoaineisto. Microsoft. <<https://blogs.office.com/en-us/2016/05/04/the-future-of-sharepoint/>>. Luettu 18.8.2017.
- 15 What is SharePoint?. 2017. Verkkoaineisto. Microsoft. <<https://support.office.com/en-us/article/What-is-SharePoint-97b915e6-651b-43b2-827d-fb25777f446f>>. Luettu 18.8.2017.
- 16 Wright, Steve. 2013. Pro SharePoint 2013 App Development. New York: Springer Science+Business Media.

- 17 Compare SharePoint Online options. 2017. Verkkoaineisto. Microsoft. <<https://products.office.com/en-us/sharepoint/compare-sharepoint-plans>>. Luettu 14.10.2017.
- 18 SharePoint Service Description. 2017. Verkkoaineisto. Microsoft, <<https://technet.microsoft.com/en-us/library/sharepoint-service-description.aspx>>. Luettu 14.10.2017.
- 19 Kashman, Mark. 2017. Work better together with SharePoint team sites Office 365 app integrations. Verkkoaineisto. Microsoft. <<https://techcommunity.microsoft.com/t5/SharePoint-Blog/Work-better-together-with-SharePoint-team-sites-Office-365-app/ba-p/109550>>. Luettu 12.11.2017.
- 20 Customizing the "modern" experiences in SharePoint Online. 2017. Verkkoaineisto. Microsoft. <<https://docs.microsoft.com/en-us/sharepoint/dev/solution-guidance/modern-experience-customizations>>. Luettu 12.11.2017.
- 21 Your intranet in your pocket—the SharePoint mobile app for iOS is now available. 2017. Verkkoaineisto. Microsoft. <<https://blogs.office.com/en-us/2016/06/21/your-intranet-in-your-pocket-the-sharepoint-mobile-app-for-ios-is-now-available/?eu=true>>. Luettu 12.11.2017.
- 22 Roine, Jussi & Jääskeläinen, Olli. 2017. SharePoint Development with the SharePoint Framework. Birmingham: Packt Publishing.
- 23 SharePoint Framework (SPFx) enterprise guidance. 2017. Verkkoaineisto. Microsoft. <<https://dev.office.com/sharepoint/docs/spfx/enterprise-guidance>>. Luettu 17.8.2017.
- 24 Holme, Dan. 2016. SharePoint and OneDrive welcome Microsoft Teams. Verkkoaineisto. Microsoft. <<https://techcommunity.microsoft.com/t5/SharePoint-Blog/SharePoint-and-OneDrive-welcome-Microsoft-Teams/ba-p/26432>>. Luettu 10.11.2017.
- 25 Jamison, Scott; Hanley, Susan & Bortlik, Chris. 2014. Essential SharePoint 2013. New Jersey: Assison-Wesley.
- 26 Smith, Tony. 2016. SharePoint 2016 User Guide. Fifth Edition. New York: Springer Science+Business Media.
- 27 What is a list in SharePoint Online?. 2016. Verkkoaineisto. Microsoft. <<https://support.office.com/en-us/article/What-is-a-list-in-SharePoint-Online-93262a88-20ad-4edc-8410-b6909b2f59a5>>. Luettu 20.8.2017.
- 28 Introduction to libraries. 2017. Verkkoaineisto. Microsoft. <<https://support.office.com/en-us/article/Introduction-to-libraries-7d4221d9-8fb9-40d5-8441-2374c84b5e26>>. Luettu 20.8.2017.
- 29 Differences between classic and new experiences for lists and document libraries. 2017. Verkkoaineisto. Microsoft. <<https://support.office.com/en-us/article/Differences-between-classic-and-new-experiences-for-lists-and-document-libraries-30e1aab0-a5cc-4363-b7f2-09e2ae07d4dc?ui=en-US&rs=en-US&ad=US>>. Luettu 17.8.2017.
- 30 Londer, Olga & Coventry, Penelope. 2016. Microsoft SharePoint 2016 Step by Step. Redmond, Washington: Microsoft Press.
- 31 Add an app part to a classic page. 2017. Verkkoaineisto. Microsoft. <<https://support.office.com/en-us/article/Add-an-App-Part-to-a-classic-page-6f06c0b7-44b8-4c69-b4ad-85197eee8d78?ui=en-US&rs=en-US&ad=US>>. Luettu 22.8.2017.

- 32 Tenant-Scoped solution deployment for SharePoint Framework solution. Verkkoaineisto. Microsoft. <<https://dev.office.com/sharepoint/docs/spfx/tenant-scoped-deployment>>. Luettu 13.9.2017.
- 33 SharePoint Dev Center. 2017. Verkkoaineisto. Microsoft. <<https://dev.office.com/sharepoint>>. Luettu 12.11.2017.
- 34 Roine, Jussi ja Anttila, Juha. 2014. SharePoint-opas. Helsinki: Roine ja Anttila.
- 35 Valo Intranet. 2017. Verkkoaineisto. Blue Meteorite Ltd. <<https://www.valointranet.com/benefits/>>. Luettu 14.11.2017.
- 36 Overview of SharePoint Framework Extensions. 2017. Verkkoaineisto. Microsoft. <<https://docs.microsoft.com/en-us/sharepoint/dev/spfx/extensions/overview-extensions>>. Luettu 15.11.2017.
- 37 Wodehouse, Carey. 2015. Choosing the Right Software Stack. Verkkoaineisto. Upwork Global Inc. <<https://www.upwork.com/hiring/development/choosing-the-right-software-stack-for-your-website/>>. Luettu 11.11.2017.
- 38 Node.js. Verkkoaineisto. Node.js Foundation. <<https://nodejs.org/en/>>. Luettu 15.8.2017.
- 39 What is npm?. 2017. Verkkoaineisto. npmjs.com. <<https://docs.npmjs.com/getting-started/what-is-npm>>. Luettu 4.11.2017.
- 40 yeoman.io. Verkkoaineisto. yeoman.io. <<http://yeoman.io/>>. Luettu 17.8.2017.
- 41 Scaffold projects using yeoman SharePoint generator. 2017. Verkkoaineisto. Microsoft. <<https://dev.office.com/sharepoint/docs/spfx/toolchain/scaffolding-projects-using-yeoman-sharepoint-generator>>. Luettu 17.8.2017.
- 42 Use Office UI Fabric React components in your SharePoint client-side web part. 2017. Verkkoaineisto. Microsoft. <<https://docs.microsoft.com/en-us/sharepoint/dev/spfx/web-parts/get-started/use-fabric-react-components>>. Luettu 4.11.2017.
- 43 Set up your SharePoint client-side web part development environment. 2017. Verkkoaineisto. Microsoft. <<https://docs.microsoft.com/en-us/sharepoint/dev/spfx/set-up-your-development-environment>>. Luettu 11.11.2017.
- 44 Set up your Office 365 tenant. 2017. Verkkoaineisto. Microsoft. <<https://docs.microsoft.com/en-us/sharepoint/dev/spfx/set-up-your-developer-tenant>>. Luettu 11.11.2017.
- 45 Catrinescu, Vlad & Seward, Trevor. 2016. Deploying SharePoint 2016. New York: Apress.
- 46 Build your first SharePoint client-side web part. 2017. Verkkoaineisto. Microsoft. <<https://dev.office.com/sharepoint/docs/spfx/web-parts/get-started/build-a-hello-world-web-part#preview-the-web-part>>. Luettu 4.11.2017.
- 47 Roine, Jussi & Jääskeläinen, Olli. 2017. SharePoint Development with the SharePoint Framework. Birmingham: Packt Publishing.
- 48 Deploy your SharePoint client-side web part to a CDN. 2017. Verkkoaineisto. Microsoft. <<https://docs.microsoft.com/en-us/sharepoint/dev/spfx/web-parts/get-started/deploy-web-part-to-cdn>>. Luettu 14.11.2017.
- 49 Pathan, Mukaddim; Buyya, Rajkumar & Vakali, Athena. 2008. Content Delivery Networks. Berlin: Springer.

- 50 Juvonen, Vesa. 2017. General availability of Office 365 CDN. Verkkoaineisto. Microsoft. <<https://dev.office.com/blogs/general-availability-of-office-365-cdn>>. Luettu 12.11.2017.
- 51 Create a team site in SharePoint Online. 2017. Verkkoaineisto. Microsoft. <<https://support.office.com/en-us/article/Create-a-team-site-in-SharePoint-Online-ef10c1e7-15f3-42a3-98aa-b5972711777d>>. Luettu 13.9.2017.
- 52 Learn about Office 365 Groups. 2017. Verkkoaineisto. Microsoft. <<https://support.office.com/en-us/article/Learn-about-Office-365-groups-b565caa1-5c40-40ef-9915-60fdb2d97fa2>>. Luettu 13.9.2017.
- 53 Schade, Amy. 2016. 6 Tips for Successful Personalization. Verkkoaineisto. Nielsen Norman Group. <<https://www.nngroup.com/articles/personalization/>>. Luettu 10.11.2017.
- 54 Office 365 API reference. 2017. Verkkoaineisto. Microsoft. <<https://msdn.microsoft.com/en-us/office/office365/api/api-catalog>>. Luettu 11.11.2017.
- 55 Call the Microsoft Graph API using OAuth from your web part. Verkkoaineisto. Microsoft. <<https://dev.office.com/sharepoint/docs/spfx/web-parts/guidance/call-microsoft-graph-from-your-web-part>>. Luettu 22.8.2017
- 56 Spasovski, Martin. 2013. OAuth 2.0 Identity and Access Management Patterns. Birmingham: Packt Publishing.
- 57 The OAuth 2.0 Authorization Framework. 2012. Verkkoaineisto. Internet Engineering Task Force. <<https://tools.ietf.org/html/rfc6749#section-3.1.2>>. Luettu 10.9.2017.
- 58 Overview of the GraphHttpClient class. 2017. Verkkoaineisto. Microsoft. <<https://docs.microsoft.com/en-us/sharepoint/dev/spfx/overview-graphhttpclient>>. Luettu 15.11.2017.
- 59 Azure Active Directory Authentication Libraries. 2017. Verkkoaineisto. Microsoft. <<https://docs.microsoft.com/en-us/azure/active-directory/develop/active-directory-authentication-libraries>>. Luettu 15.11.2017.
- 60 SPHttpClient class. 2017. Verkkoaineisto. Microsoft. <<https://docs.microsoft.com/en-us/javascript/api/sp-http/sphttpclient>>. Luettu 13.11.2017.
- 61 JavaScript Core Library. 2017. Verkkoaineisto. SharePoint Patterns and Practices Community. <<https://github.com/SharePoint/PnP-JS-Core>>. Luettu 13.11.2017.
- 62 Use the Microsoft Graph API. 2017. Verkkoaineisto. Microsoft. <https://developer.microsoft.com/en-us/graph/docs/concepts/use_the_api>. Luettu 11.11.2017.
- 63 Intranet Information Architecture (IA) Trends. 2014. Verkkoaineisto. Nielsen Norman Group. <<https://www.nngroup.com/articles/intranet-information-architecture-ia/?lm=intranet-design-annual&pt=report>>. Luettu 13.9.2017.
- 64 GitHub/gulpjs/gulp. 2017. Verkkoaineisto. GitHub, Inc. <<https://github.com/gulpjs/gulp>>. Luettu 17.8.2017.
- 65 Overview of the SharePoint Framework. 2017. Verkkoaineisto. Microsoft. <<https://dev.office.com/sharepoint/docs/spfx/sharepoint-framework-overview>>. Luettu 24.8.2017.
- 66 SharePoint Framework (SPFx) enterprise guidance. Verkkoaineisto. Microsoft. <<https://dev.office.com/sharepoint/docs/spfx/enterprise-guidance>>. Luettu 17.8.2017.

- 67 White, Martin. 2017. IntranetFocus. Verkkoaineisto. <<http://intranetfocus.com/intranet-design-annual-2017/>>. Luettu 1.10.2017.
- 68 Hanley, Susan. 2017. End User Involvement, Governance, and Expert Help Drive Success in the 10 Best Intranets of 2017. Verkkoaineisto. Computerworld. <<https://www.computerworld.com/article/3157858/software/end-user-involvement-governance-and-expert-help-drive-success-in-the-10-best-intranets-of-2017.html>>. Luettu 1.10.2017.
- 69 How to sign in any Azure Active Directory (AD) user using the multi-tenant application pattern. 2017. Verkkoaineisto. Microsoft. <<https://docs.microsoft.com/en-us/azure/active-directory/develop/active-directory-devhowto-multi-tenant-overview>>. Luettu 23.8.2017.
- 70 Integrating applications with Azure Active Directory. 2017. Verkkoaineisto. Microsoft. <<https://docs.microsoft.com/en-us/azure/active-directory/develop/active-directory-integrating-applications>>. Luettu 23.8.2017.
- 71 Intranet Portals: UX Design Experience from Real-Life Projects. Verkkoaineisto. Nielsen Norman Group. <<https://www.nngroup.com/reports/intranet-portals-experiences-real-life-projects/>>. Luettu 11.11.2017.
- 72 Nielsen, Jakob. 2013. Intranet Social Features. Verkkoaineisto. Nielsen Norman Group. <<https://www.nngroup.com/articles/intranet-social-features/>>. Luettu 11.11.2017.
- 73 Introduction to libraries. 2017. Verkkoaineisto. Microsoft. <<https://support.office.com/en-us/article/Introduction-to-libraries-7d4221d9-8fb9-40d5-8441-2374c84b5e26>>. Luettu 20.8.2017.
- 74 Overview of the SharePoint Framework. 2017. Verkkoaineisto. Microsoft. <<https://dev.office.com/sharepoint/docs/spfx/sharepoint-framework-overview>>. Luettu 24.8.2017.
- 75 The Very Best Intranets & Digital Workplace from the 2016 Intranet Global Forum. 2016. Verkkoaineisto. Prescient Digital Media. <<https://www.slideshare.net/Prescient/the-very-best-intranets-digital-workplace-from-the-2016-intranet-global-forum>>. Luettu 8.11.2017.
- 76 Bilbie, Alex. 2017. A Guide To OAuth 2.0 Grants. Verkkoaineisto. <<https://alexbilbie.com/guide-to-oauth-2-grants/>>. Luettu 14.11.2017.
- 77 Mastykarz, Waldek & Juvonen, Vesa. 2017 Accessing data in SharePoint from SharePoint Framework solution. Verkkovideo. Microsoft. <https://dev.office.com/blogs/accessing-data-in-sharepoint-from-sharepoint-framework-solution>. Luettu 24.8.2017.