



Keskustelubotin toteuttaminen Microsoft Copilot Studio -työkalulla

Jukka-Pekka Sirviö

Haaga-Helia ammattikorkeakoulu

Tietojenkäsittelyn koulutusohjelma

Opinnäytetyö

2024

Tiivistelmä

Tekijä(t) Jukka-Pekka Sirviö
Tutkinto Tradenomi
Raportin/Opinnäytetyön nimi Keskustelubotin toteuttaminen Microsoft Copilot Studio -työkalulla
Sivu- ja liitesivumäärä 39 + 0
<p>Opinnäytteessä suunniteltiin, määriteltiin ja toteutettiin keskustelubotti käyttäen low-code-alueita. Toteutukseen käytetty työkalu oli Microsoft Power Platform -tuoteperheeseen sisältyvä Copilot Studio. Keskustelubotti toteutettiin toimeksiantajan sisäiseen käyttöön helpottamaan käyttäjäkunnan tietotarpeita hallinnollisissa asioissa.</p> <p>Botti toteutettiin valmiiseen kehitysympäristöön, joten raportissa ei käsitellä ympäristön pystyttämistä. Työkalua tai ympäristöä ei toteuttamisen hetkellä ollut otettu virallisesti toimeksiantajalla käyttöön, eikä bottia voinut tämän vuoksi julkaista. Tästä syystä työssä ei käsitellä botin julkaisuun ja käyttöönottoon liittyviä toimia. Työssä arvioitiin valitun työkalun ja low-code-kehittämismallin soveltuvuutta valittuun toteutustyöhön.</p> <p>Raportin kuvailemassa toteutusprojektissa suunniteltiin ja määriteltiin keskustelubotti soveltaen yleisiä vaatimusmäärittelymalleja. Projektille laadittiin ylätasoiset ei-toiminnalliset ja toiminnalliset vaatimukset kuvaamaan toteutuksen sisältöä. Nämä vaatimukset on kirjattu raporttiin. Konkreettisemmän tason määrittelyt kirjattiin työkalun toimintaa tukevana kulkuina erilliseen dokumentaatioon. Raportissa on kuvattuna dokumentaation malli.</p> <p>Keskustelubotin toteuttamisen osalta raportissa käydään läpi Copilot Studion ominaisuuksia, joita toteutustyössä käytettiin ja arvioidaan generatiivisen tekoälyn käyttöön liittyviä ongelmakohtia ja syitä jättää sitä käyttämättä projektissa. Toteutuksen osalta raportissa kuvataan käytännöt, joiden perusteella työkalun ominaisuuksia käytettiin ja minkälaiseen lopputulokseen niiden avulla päästiin. Tarkkoja käyttöohjeita ei esitellä, koska ne ovat luettavissa työkalun dokumentaatiosta.</p> <p>Toteutetun tuotteen ei nähdä hyödyntävän kovinkaan laajasti keskustelubottien moderneja ominaisuuksia, mutta sen nähdään täyttävän tavoitellun tarkoituksen. Jatkokehityksessä arvioidaan pystyttävän hyödyntämään työkalun ominaisuuksia, joita hyödyntämällä botin toimintaa voi parantaa ja modernisoida. Erityisesti generatiivisen tekoälyn käytössä voisi olla suuresti potentiaalia tehostamaan botin toimintaa. Generatiivista tekoälyä testattiin projektin aikana, mutta sen tuottamiin tuloksiin ei oltu tyytyväisiä ja tekoälyn käyttö päätettiin jättää minimaaliseksi.</p> <p>Projektin perusteella Copilot Studion arvioitiin olevan lupaava työkalu keskustelubottien toteuttamiseen. Projektissa hyödynnettiin vain yleisimpiä ominaisuuksia, mutta siitä tunnistettiin useita ominaisuuksia, joita olisi mahdollista hyödyntää jatkokehityksessä.</p>
Asiasanat low-code, keskustelubotti, ohjelmistokehitys, ohjelmistosuunnittelu

Sisällys

1	Johdanto	1
1.1	Rajaukset	2
1.2	Toteutusprojektin vaikuttavuus ja ajankohtaisuus	3
1.3	Keskeiset käsitteet	4
2	Keskustelubotin suunnittelu ja määrittely	5
2.1	Perusteet keskustelubotin rakentamiseen ja low-code-alustan käyttämiseen	5
2.2	Arvio työkalun low-code-luonteesta	5
2.3	Vaatusmäärittelymalli	6
2.4	Dokumentaatiomalli	7
2.5	Tekninen alusta, ei-toiminnalliset vaatimukset ja ympäristövaatimukset	7
2.6	Tietohallinnon toiminnalliset vaatimukset ja sisältö	9
2.7	Henkilöstöhallinnon toiminnalliset vaatimukset ja sisältö	11
3	Keskustelubotin toteutus	13
3.1	Työkalun ominaisuudet	13
3.1.1	Generatiivinen tekoäly	14
3.1.2	Muuttujat	16
3.2	Alustan valmistelu toteutusta varten	17
3.3	Teknisten vaatimusten toteutus	18
3.4	Toiminnallisten vaatimusten toteutus	20
3.4.1	Generatiivisen tekoälyn hyödyntäminen	21
3.4.2	Aiheiden syöttäminen	24
3.4.3	Toteutuskaari dokumentoidusta kulusta aiheeksi	26
3.5	Keskustelubotin testaus ja laadun arviointi	30
4	Pohdinta	33
4.1	Jatkokehityssuunnitelma	33
4.2	Tekijän oma oppiminen	36
	Lähteet	37

1 Johdanto

Opinnäytteessä suunniteltiin, määriteltiin ja toteutettiin virtuaalinen avustaja (jatkossa keskustelubotti tai botti) ohjaamaan käyttäjiä organisaation sisäisen hallinnon toimenpiteissä. Tavoitteena oli tuottaa ratkaisu, jonka ylläpito on tehtävissä niin helposti, että sitä voivat tehdä toimialojen asiantuntijat.

Opinnäytteen toimeksiantajalla on paljon hallinnollisen puolen ohjeita ja sääntöjä, mutta niille ei ole ollut yhtä selkeää tallennuspaikkaa. Kaikkia ohjeita ei tunneta käyttäjäkunnassa tai niiden etsimiseen joudutaan kuluttamaan ylimääräistä aikaa. Tämä aiheuttaa paljon kyselyitä hallinnollisten toimialueiden asiantuntijoille. Kyselyt ja ohjeiden etsiminen aiheuttaa toiminnan tehottomuutta läpi koko organisaation. Keskustelubotti, jolta käyttäjä voi kysyä neuvoa ja joka osaisi antaa useaan eri toimialaan liittyviä vastauksia, parantaisi tehokkuutta ja työtyytyväisyyttä sekä tehostaisi käyttäjien tasapuolista tiedonsaantia. Ohjeita päivitetään ja uusitaan usein, joten on tärkeää, että botin tietojen pitäminen ajan tasalla on tehokasta. Tämän vuoksi tavoitteena on saada ratkaisu, jossa ohjeiden laatijat itse voivat viedä tiedot ohjeista keskustelubottiin.

Ratkaisu kehitettiin hyödyntäen toimeksiantajalla käytössä olevaa Microsoftin Power Platform low-code-alustaa ja erityisesti Copilot Studio -työkalua, joka on tarkoitettu keskustelubottien kehittämiseen (Microsoft 2024a). Copilot Studio on tämän kyseisen työkalun uusi nimitys, joka on otettu käyttöön vuoden 2023 loppupuolella. Työkalu tunnettiin aiemmin nimellä Power Virtual Agents ja osa raportin viitteistä käsittelee työkalua tällä nimellä.

Low-code-ratkaisut tulevat mahdollistamaan entistä nopeamman ja helpomman sovelluskehittämisen ja tuovat kehittämistyökalut lähelle asiantuntijoita ja loppukäyttäjiä (Goasduff 2019). Toimeksiantajalla on halukkuutta omaan sovelluskehittämiseen ja low-code-ratkaisut voisivat olla riittävä työkalu tämän tavoitteen saavuttamiseen. Tästä näkökulmasta opinnäytteessä kehitettävä keskustelubotti toimii myös vahvistuksena valitun työkalun soveltuvuudelle toimeksiantajan käyttöön. Erittäin vahvistusta haettiin low-code-ratkaisujen soveltuvuudelle niin sanotulle kansalaiskehittämiselle eli sille, että sovelluskehittämistä tekevät asiantuntijat, joilla ei ole perinteistä IT-kehittämisen osaamista.

Ohjelmistovalmistaja SAP määrittelee low-coden tavaksi "suunnitella ja kehittää sovelluksia käyttämällä intuitiivisia graafisia työkaluja ja upotettuja toimintoja, jotka vähentävät perinteisiä -- kirjoitusvaatimuksia. -- low-code-kehitys tarjoaa lisätyn ja yksinkertaistetun käyttökokemuksen, jonka avulla käyttäjät voivat aloittaa luomisen nopeasti." (SAP s.a.) Low-code-ratkaisu on siis omiaan

helpottamaan ja nopeuttamaan organisaation omaa sovelluskehitystyötä, erityisesti kun omat tietohallintoresurssit ovat niukat. Low-code-kehittämiseen sisältyy myös ajatus kansalaiskehittäjistä eli sovelluskehityksen vastuun siirtäminen dedikoiduilta kehittäjiltä/ohjelmoijilta suoraan toiminnan asiantuntijoille, mikä vähentää tarvetta erillisille henkilöresursseille ja sisältää potentiaalin kehitystyön tehostumiselle. (SAP s.a.)

Raportissa käydään läpi keskustelubotin toteutusprojektin suunnittelu- ja määrittelyvaiheet luvussa 2 sekä toteutus- ja testausvaiheet luvussa 3. Luku 4 käsittelee pohdintaa lopputuloksesta, jatkokehitystarpeista ja arvioi raportin tekijän omaa oppimista projektin aikana.

1.1 Rajaukset

Toimeksiantajan organisaatorakenteessa on useita toimialueita, joissa keskustelubottia olisi mahdollista hyödyntää, mutta työmäärän hallinnan vuoksi projektissa käsiteltäviksi toimialueiksi rajataan tietohallinto ja henkilöstöhallinto. Tekijä toimii asiantuntijana tietohallinnossa ja osasi määrittellä tämän aihealueen tietoja. Toimeksiantajan henkilöstöhallinto on esittänyt toiveita tämän kaltaisen keskustelubotin hyödyntämisestä, joten projektissa tarjottiin heille tilaisuus osallistua sellaisen kehittämiseen.

Koska työkalut ovat vasta testikäytössä, ratkaisun julkaiseminen ja käyttöönotto rajataan työn ulkopuolelle. Jos Power Platform todetaan toimeksiantajalla low-code-kehittämiseen soveltuvaksi työkaluksi, julkaistaan kehitetty keskustelubotti myöhemmin käyttöön. Työssä kehitettävä botti on siis ns. proof-of-concept. Pitkällä aikavälillä botti on kuitenkin tarkoitus ottaa käyttöön, mutta tämä vaatii vielä tiettyjä uudistuksia toimeksiantajan infrastruktuuriin. Näillä uudistustarpeilla ei ole suoranaista vaikutusta opinnäytteen aiheeseen, joten tätä aihetta ei käsitellä raportissa.

Toimeksiantajan IT-infrastruktuuri hankitaan palveluna, eikä toimeksiantaja itse vastaa infrastruktuurin konfiguroinnista. Tämän vuoksi raportissa ei myöskään käsitellä kehitysympäristön pystyttämistä, vaan se on tullut annettuna palveluntuottajalta. Työssä ei myöskään käydä läpi Power Platformin tai Copilot Studion lisenssivaatimuksia tai työkalun käytön kustannuksia. Työ on tehty käyttäen Premium-lisenssiä, joka sisältää kaikki tarjolla olevat työkalut. Raportti keskittyy siihen, miten työkaluilla rakennetaan ratkaisu ja minkälaista osaamista työkalujen käyttö edellyttää.

Opinnäytteen alkuperäisenä suunnitelmana oli rajata generatiivisen tekoälyn käyttö kokonaan pois, koska opinnäytteen suunnittelun alkaessa, sitä ei ollut tarjolla suomenkielisenä Copilot Studiossa. Tähän kuitenkin tuli muutos, kun Microsoft tammikuussa 2024 lisäsi työkaluun tuen suomenkieliselle generatiiviselle tekoälylle. Raportissa käsitellään kokemuksia generatiivisen tekoälyn testaa-

misesta määrittelyn ja toteutuksen ohella sekä syitä siihen, miksi työssä päätettiin jättää sitä käyttämättä. Generatiivisen tekoälyn ominaisuudet eivät kuitenkaan ole opinnäytteen laajuuden pääosa.

1.2 Toteutusprojektin vaikuttavuus ja ajankohtaisuus

Toimeksiantajana toimii tekijän työnantaja Syyttäjälaitos (jatkossa toimeksiantaja).

Syyttäjä on valtion viranomainen, joka huolehtii siitä, että rikoksesta seuraa lain tarkoittama rangaistus. Syyttäjälaitos toimii syyttäjien hallinnollisena rakenteena, jossa työskentelee noin 650 henkilöä, joista valtaosa toimii rikosprosessin ydintehtävien parissa. Puhtaasti hallinnollisia tehtäviä suorittaa noin neljäkymmentä henkilöä. Hallinnolliset tehtävät jakautuvat mm. johtamiseen, henkilöstöhallintoon, taloushallintoon ja tietohallintoon. (Syyttäjälaitos s.a.)

Tekijä työskentelee Syyttäjälaitoksessa valtakunnansyyttäjän toimistoon asetetussa tietoyksikössä, jonka vastuulla on Syyttäjälaitoksen tietohallinnon kehittäminen. Opinnäytteen tavoite on tekijän työnantajan ICT-kehittämisen tavoitteiden mukainen.

Opinnäytteessä ei käsitellä syyttäjien ydintoimintaan liittyviä asioita, vaan vain hallinnollisia toimenpiteitä. Tavoitteena on parantaa toimeksiantajan käyttäjien tietämystä tietohallintoon ja henkilöstöhallintoon liittyvistä ohjeista ja käytännöistä. Nykytilassa ohjeita on paljon ja niitä on tallennettu eri paikkoihin. Käyttäjät eivät tiedä mistä etsiä yksittäisiä ohjeita, eivätkä he välttämättä edes tiedä tietyn ohjeen olemassaolosta. Botti osaa kysyttäessä ohjata käyttäjää löytämään ja soveltamaan oikeaa tilanteeseen liittyvää ohjetta. Lisäksi botti ohjaa toimialoja tunnistamaan tilanteita, joihin ei ole olemassa ohjeistusta.

Botti ohjaa käyttäjien kyselyitä oikeaan paikkaan. Esimerkiksi henkilöstöhallinto muodostuu kahdesta tasosta: Syyttäjälaitoksen sisäisestä henkilöstöhallinnon yksiköstä ja valtionhallinnon yhteisestä palvelukeskuksesta. Käyttäjät eivät aina tiedä onko heidän otettava yhteyttä henkilöstöyksiköön vai palvelukeskukseen, mikä johtaa ylimääräiseen työhön turhien yhteydenottojen vuoksi. Tietohallinnossa on samankaltainen tilanne ja siellä eri tasoisia toimijoita on vielä enemmän. Botti osaa ohjata yhteydenotot oikeaan osoitteeseen, mikä tehostaa kaikkien toimijoiden toimintaa.

Keskustelubotin soveltuvuudesta tällaiseen käyttöön toimeksiantajalla ei ole selvyyttä. Copilot Studion käytöstä julkishallinnossa on olemassa referenssejä (Microsoft 2024b), mutta botin käytöstä julkishallinnon organisaation sisäisessä käytössä ei ole saatavilla paljoa tietoa. Opinnäytteessä

tämä osuus jää lopullisesti selvittämättä, koska tuotosta ei ole tarkoitus ottaa käyttöön projektin aikana. Toteutuksen ohessa on kuitenkin mahdollista tehdä arviointia botin soveltuvuudesta ylläpidettävyyden ja tietosisällön näkökulmasta.

1.3 Keskeiset käsitteet

Avustaja: Microsoftin Copilot Studion dokumentaatioissa käyttämä nimitys tekoälyohjatusta keskustelubotista.

Chatbot: kts. keskustelubotti.

Copilot Studio: Microsoftin työkalu avustajien eli keskustelubottien low-code-kehittämiseen, joka on osa Power Platform -tuoteperhettä ja tunnettu aiemmin nimellä Power Virtual Agents (Microsoft 2024a). Tärkeä huomio on se, että Copilot Studio on eri asia, kuin Microsoft Copilot. Copilot on tekoälypohjainen avustajaominaisuus, jota Microsoftin tuotteet, kuten Copilot Studio, voivat hyödyntää (Muchmore 2024).

Entiteetti: Copilot Studion ominaisuus, jonka voi katsoa vastaavan perinteisen ohjelmoinnin globaalia muuttujaa.

Generatiivinen tekoäly: Tekoäly, joka osaa luoda uutta sisältöä hyödyntämällä koneoppimista ja suuria tietomääriä (Adobe s.a.).

Keskustelubotti: Sovellus, joka keskustelee käyttäjän kanssa ottamalla vastaan luonnollisella kielellä esitettyjä kehoitteita ("prompteja") ja jonka vastaukset ovat myös luonnollisella kielellä esitettyjä.

Low-code: sovelluskehittämisen malli, jossa perinteisen ohjelmoinnin sijaan sovelluksia kehitetään käyttäen graafisia työkaluja ja niiden valmiita toimintoja. Low-code-kehittäminen vaatii jonkin verran ohjelmointiosaamista. (SAP s.a.)

Power Platform / Power Apps: Microsoftin low-code-kehittämiseen tarkoitettu kehittämisalusta ja sarja graafisia työkaluja. (Microsoft s.a.)

2 Keskustelubotin suunnittelu ja määrittely

Tämä luku käsittelee keskustelubotin suunnittelua ja määrittelyä. Ensimmäisissä aliluvuissa käsitellään yleisellä tasolla keskustelubottien ja low-code-työkalun luonnetta. Seuraavaksi käydään läpi projektissa käytetyt vaatimusmäärittelyn ja dokumentoinnin mallit. Viimeiset aliluvut käsittelevät projektissa tehtyä teknistä ja toiminnallista vaatimusmäärittelyä.

2.1 Perusteet keskustelubotin rakentamiseen ja low-code-alustan käyttämiseen

Perinteisesti tiedonkeruuseen on käytetty lomakkeita, johon käyttäjät ovat täyttäneet tietoja. Keskustelubotit kuitenkin mahdollistavat tietojen keräämisen keskustelumaisen käyttöliittymän välityksellä luoden käyttäjälle lähestyttävämmän ilmapiirin. Botin kanssa toimiminen on käyttäjälle mielekkäämpää, kun hän ei joudu etsimään omaan asiaansa sopivaa lomaketta, eikä myös täyttämään sellaisia tietoja, joilla ei ole merkitystä käyttäjän kysymyksessä. (Rivera 2023, luku 2; Goasduff 2019)

Low-code-kehittämisellä pyritään mahdollistamaan organisaatioiden pyrkimys nopeuttaa teknologiavetoista ongelmien ratkaisua sekä tehdä toteutustyöstä nopeampaa ja kustannustehokkaampaa. Low-code-alustoilla, kuten Power Platform tuoteperheellä, voi toteuttaa sovellusratkaisuja perinteisiä ohjelmistokehitysmalleja nopeammin. Lisäksi työkalujen helppokäyttöisyys tarkoittaa sitä, että ratkaisun toteutus ei edellytä yhtä suurta projektiorganisaatiota kuin mitä perinteisessä kehitysprojektissa on tarvittu. (Rivera 2023, luku 1.)

Botti olisi teoriassa mahdollista toteuttaa pelkästään generatiivista tekoälyä hyödyntäen, mutta tämä voisi aiheuttaa ongelmia. Tekoälyjen ongelmana on usein huono kyky ymmärtää keskusteltavaa tekstisisältöä, ja ne edellyttävät suurta määrää opetusaineistoa (Nirala & Singh 2022, luku 1.1). Näitä ongelmia arvioitiin projektin aikana.

2.2 Arvio työkalun low-code-luonteesta

Microsoft määrittelee Copilot Studion low-code-työkaluksi (Microsoft 15.11.2023) eli määritelmän mukaisesti kehittämistyö sitä käyttäen voi vaatia jonkin verran ohjelmointiosaamista. No-code-työkalu tarkoittaisi, että kehittäminen olisi tehtävissä täysin ilman ohjelmointia tai vaatimusta ohjelmointiosaamisesta, niin että käytettävä työkalu tarjoaa täysin visuaalisen käyttöliittymän kaikkien toimintojen tekemiseen (Nakanishi s.a.). Copilot Studio on työkaluna hyvin lähellä näiden määritelmien rajaa, koska lähes kaikki sen ominaisuudet ovat käytettävissä käyttöliittymän välityksellä. Vain jotkin yksittäiset edistyneemmät ominaisuudet vaativat ohjelmointiosaamista. Työkaluun syötettyjä toimintoja voi tarkastella json-muotoisena koodina, jota on mahdollista muokata ja kopioida.

Käytännössä opinnäytetyössä kaikki toteutus tehtiin käyttöliittymän välityksellä. Tämän seurauksena projektista voisi puhua no-code-toteutuksena, mutta koska työkalun virallinen määritelmä on low-code, on raportissa käytetty tätä termiä.

2.3 Vaatimusmäärittelymalli

Low-code-kehittämisen parhaat käytännöt noudattavat perinteisen sovelluskehittämisen askeleita: määrittele, toteuta, testaa. Pääasiallinen ero on siinä, että toteutuksessa ei niinkään kirjoiteta ohjelmakoodia, vaan sovellus rakennetaan valmiista komponenteista. (SAP s.a.)

Määrittelyaskeleessa yleensä tuotetaan vaatimuksia. Yksinkertaistetusti vaatimukset ovat lista asioista, joita järjestelmän kehittämisessä on huomioitava. Vaatimukset kertovat mitä järjestelmän pitää tehdä ja minkälaisia ominaisuuksia sen kuuluu sisältää. Vaatimukset voivat myös asettaa järjestelmälle tai toteutusprosessille rajoitteita. Vaatimukset muodostetaan vaatimusmäärittelyprosessissa. (Wiegers & Beatty 2013, luku 1.)

Vaatimusmäärittelyssä laajimmillaan dokumentoidaan toiminnan nykyisen ja tavoitellun tilan yleiskuvaus, ulkoiset ja sisäiset sidosryhmät, käyttötapaukset sekä käsiteltävät tiedot. Näiden tietojen perusteella muodostetaan toiminnalliset ja ei-toiminnalliset vaatimukset sekä rajoitteet. Projektiin valittu määrittelymalli määrittää miten nämä tiedot dokumentoidaan ja missä laajuudessa. Esimerkiksi osaa näistä tiedoista on mahdollista kuvata ns. käyttäjätarinoina, joissa käydään läpi tilanteita, joissa tietyn roolin käyttäjät käyttävät järjestelmää suorittaakseen jonkin prosessin. Toinen yleinen tapa on dokumentoida testitapaukset, joissa käydään yksityiskohtaisesti läpi yksittäiset toimenpiteet, joita järjestelmässä suoritetaan, toimenpiteiden odotetut lopputulokset sekä tilanteet, jotka aiheuttavat toimenpiteessä poikkeaman. (Wiegers & Beatty 2013, luku 1.)

Projektissa määrittely tehtiin hyvin kevyellä mallilla. Määrittelyssä katsottiin riittävien listaukset toiminnallisista ja ei-toiminnallisista vaatimuksista sekä dokumentaatiomallin mukaiset kulut. Perusteina tällä mallilla toimimiseen on se, että kehitysprojekti on organisaation sisäisesti toteuttama ja käytännössä vain yhden kehittäjän tuotos. Projekti myös toteutettiin valmiiseen kehittämisalustaan, joka noudattaa tiettyä rakennetta. Laajempaa ja tarkempaa määrittelyä ei näiden syiden vuoksi nähty tarpeelliseksi. Kehitys- ja määrittelyresurssien oikean suuntaamisen vuoksi oli enemmän perusteltua pitää määrittely kevyenä.

Keskustelubotin määrittely tehtiin JHS-173 suositusta soveltaen. Suositus on laadittu perinteisiä ohjelmistoprojekteja varten, joten se ei täysin sovellu low-code-kehittämisessä pyrittyyn nopeaan ja kustannustehokkaaseen malliin, mutta suosituksen peruseriaatteen pätevät myös low-code-ratkaisun suunnittelussa ja dokumentoinnissa. Tavoitteena määrittelyssä on varmistaa toteutusprojektin

onnistuminen. Puutteellista määrittelyä pidetään merkittävänä syynä ohjelmistoprojektien epäonnistumiseen, minkä vuoksi määrittelyyn on pienissäkin projekteissa käytettävä aikaa. (JUHTA 2018, 9.)

JHS-173 luokittelee vaatimukset toiminnallisiin, ulkoisiin eli liityntävaatimuksiin, ei-toiminnallisiin, rajoituksiin ja ympäristövaatimuksiin (JUHTA 2018, 5-8). Projektissa toiminnalliset vaatimukset luokitellaan tietohallinto- ja henkilöstöhallintoluokkiin. Kaikki muut vaatimustyytit käsitellään ns. teknisinä vaatimuksina samannimisessä luokassa.

2.4 Dokumentaatiomalli

Tässä raportissa vaatimukset on koodattu merkinnällä:

V<vaatimusluokka>.<vaatimuksen numero>

Vaatimusluokat ovat 1 tekniset vaatimukset, 2 tietohallinnon vaatimukset, 3 henkilöstöhallinnon vaatimukset. Vaatimuksen numero on luokan sisäinen juokseva numerointi. Esimerkiksi tietohallinnon kolmas vaatimus saa koodiarvon V2.03. Suosituksista poiketen projektissa määriteltiin rajoitukset osaksi teknisiä vaatimuksia eli vaatimusluokkaan 1.

Vaatimuksien lisäksi tietohallinnon ja henkilöstöhallinnon sisältöä dokumentoitiin aiheiden mukaisiin kulkuihin. Yksi aihe käsittää botille esitettävän kysymyksen sekä viestit ja kysymykset, joita botti käyttäjälle kysymyksen jälkeen esittää. Botti tunnistaa kysymyksen ja siihen liitettävän aiheen käynnistinsanojen perusteella. Kulkudokumentaatioon kuvattiin botin palauttamamat viestit, ehtokysymykset ja ehtojen käsittelyt sekä kutsut toisiin aiheisiin.

Kulkukuvaukset toimivat toteutuksen pääasiallisena määrittelylähteenä. Vaatimuksien merkitys projektissa on jäsentää työkaluun vietävää sisältöä ja helpottaa kulkumäärittelyjen hallintaa.

Kulun toteutuskaari dokumentaatiosta järjestelmään syötetyksi tiedoksi on kuvattu esimerkkitapauksella luvussa 3.4.3.

2.5 Tekninen alusta, ei-toiminnalliset vaatimukset ja ympäristövaatimukset

Keskustelubotin teknisenä alustana toimii Microsoft Power Platform, koska se on toimeksiantajan käytössä oleva low-code-alusta. Toimeksiantajan näkökulmasta alustaan liittyy teknisiä ja ei-toiminnallisia vaatimuksia, jotka tämä valittu alusta täyttää. Näiden vaatimusten merkitys ei ole toteutuksen kannalta merkittävä, koska työkalun valintaan ei ole liittynyt erillistä harkintaa. Vaatimuksilla on merkitystä kuitenkin siinä tilanteessa, kun arvioidaan, onko low-code-alusta oikea lähtökohta

tällaiseen toteutukseen. Lisäksi niillä on mahdollista esittää perusteluita koko tuotteen merkitykselle toimeksiantajan käytössä.

Ei-toiminnallisia vaatimuksia ja rajoitteita ovat:

- V1.01 Valtionhallinnon yleiset tietoturva-vaatimukset. Toimeksiantaja on valtionhallinnon virasto, joten sen käyttämiä tietojärjestelmiä ja niiden kehittämistä ohjaa Valtioneuvoston periaatepäätös julkisen hallinnon digitaalisesta turvallisuudesta.
- V1.02 Ratkaisun käyttöoikeuksia voi rajata, jotta vain organisaation henkilöstöllä on pääsy siihen ja vain ylläpidolla ja kehittäjillä on pääsy ratkaisun tekniseen alustaan.
- V1.03 Ratkaisu tallentaa toiminnasta lokia ja erityisesti kerää tietoa sille esitetyistä kysymyksistä ja vastauksista, jotta toimintaa voi seurata sekä vastausten puutteisiin ja virheisiin voidaan puuttua.
- V1.04 Ratkaisulla on olemassa selkeä dokumentaatio ja käyttöohjeet, jotta kehittäminen on helppoa ja mahdollista myös kansalaiskehittäjille.
- V1.05 Ratkaisua voi muuttaa graafista käyttöliittymää käyttämällä, jotta kehittäminen on helppoa ja mahdollista myös kansalaiskehittäjille.
- V1.06 Graafisen käyttöliittymän tulisi olla helppokäyttöinen, jotta kehittäminen on helppoa ja mahdollista myös kansalaiskehittäjille.
- V1.07 Ratkaisu sisältää työvälineet testaamista varten, jotta julkaistava tuotteen laadun voi varmistaa ja kehitystyö on tehokasta.

Ympäristövaatimukset kuvaavat ohjelmiston toimintaympäristöön kohdistuvia vaatimuksia. Keskustelubotin ympäristövaatimuksia ovat:

- V1.08 Ratkaisu hyödyntää olemassa olevia kirjautumisjärjestelmiä, jotta käyttäjien on helppo ottaa tuote käyttöön.
- V1.09 Ratkaisu voidaan julkaista laajemman tuoteperheen julkaisujärjestelmässä, jotta ratkaisu voidaan julkaista osaksi suurempaa kokonaisuutta ja tuote pystyy hyödyntämään muita toimintoja.

Vaatimuksien laatimisessa on hyödynnetty Microsoft Copilot -tekoälyä (entinen Bing Chat).

Alustaan ei toteutuksen tässä vaiheessa kohdistu liityntävaatimuksia. Pidemmällä aikavälillä liityntävaatimus voisi olla kyky integroitua Microsoftin ohjelmistotuoteperheisiin, mikä tässä alustassa täyttyy. Mahdolliset muut liityntävaatimukset liittyisivät integraatioihin muihin toiminnallisiin järjestelmiin, joita tässä työssä ei käsitellä tietoturvasyistä ja työn laajuuden hallitsemiseksi.

Tietosuoja asettaa keskustelubotin käytölle ja sisällölle rajoitteita. Opinnäytteen laajuudessa bottissa ei käsitellä tietoa, jota sille ei voisi antaa tai tällaisen tiedon syöttäminen on mahdollista välttää ohjaamalla käyttäjät botin ulkopuolella oleviin ohjeisiin, jossa asiaa käsitellään. Toimeksiantajan toisissa toimialoissa käsitellään tietoa, joka on salassa pidettävää, turvaluokiteltua tai erityisiä henkilötietoja sisältävää. Tällaista tietoa ei voi tietoturvan ja tietosuojan varmistamiseksi antaa keskustelubotille, koska alustan tietoturvallisuudesta ja luottamuksellisuudesta ei ole riittävää dokumentaatiota. Myöskään saatavilla olevat tiedot alustan toimittajan pääsystä rekisteröityjen henkilötietoihin ja miten niitä käsitellään, ei ole tarpeeksi läpinäkyvää. Tämä rajoittaa botin käyttömahdollisuuksia erityisesti toimeksiantajan ydintehtävän osalta. Kirjoitushetken mukaisen ohjeistuksen perusteella pilvipalvelussa ajettavassa järjestelmässä voi käsitellä vain julkista tietoa. Turvaluokiteltu tai rikosasioihin liittyvä henkilötieto on selkeästi rajattava pois sisällöstä ja kaikki salassa pidettävä tai erityisiä henkilötietoja sisältävä aineisto on epäselvällä alueella. Tämän vuoksi keskustelubotissa käsiteltävä tieto on oltava pelkästään julkista. (Tietosuojavaltuutetun toimisto, s.a.)

Tietoturvallisuus on huomioitu Copilot Studiossa niin, että se noudattaa Microsoftin turvallisuuskehityksen elinkaarta. Elinkaari koostuu joukosta tarkkoja käytäntöjä, jotka tukevat suojausta ja yhdenmukaisuutta koskevia vaatimuksia. (Microsoft 2024c.)

2.6 Tietohallinnon toiminnalliset vaatimukset ja sisältö

Tietohallinnon sisältö liittyy toimeksiantajan IT-palveluiden organisointiin ja ohjeisiin. Keskustelubotin pitää vaatimuksien perusteella osata vastata kysymyksiin siitä, miten toimeksiantajan tietohallinto on organisoitu, miten tietohallinnon sidosryhmät toimivat, mitä eri IT-palvelut tekevät, mistä yksittäinen IT-palvelu on saatavilla ja mistä kysyjä voi saada tukea ongelmiansa hoitamiseen.

Botin tietohallintosisältö muodostettiin asiantuntijoiden kanssa pidetyissä työpajoissa. Työpajoissa suunniteltiin käyttötapauksia sekä kirjoitettiin määrittelydokumentaatioon kulkuja bottiin syötettäväksi. Tietojen syöttäminen kuvataan tarkemmin luvussa 3, jossa käsitellään botin toteuttamista. Jotta aineisto olisi dokumentoitu, syötetty tieto on muutettu vaatimuksiksi, jotka on tiivistetysti kuvattu tähän raporttiin.

Tietohallinnon toiminnalliset vaatimukset ovat:

- V2.01 Käyttäjä ohjataan Valtorin asiointipalveluun tai ICT-tukeen yleisiä ICT-häiriötilanteita koskevissa kysymyksissä.
- V2.02 Käyttäjä ohjataan ORK:n asiointipalveluun toimintariippuvaisten sovellusten häiriötilanteissa.

- V2.03 Yleisissä häiriötilanteissa käyttäjä ohjataan tarkistamaan yhteiset häiriöilmoitukset (intra, Teams-kanava, palveluportaali).
- V2.04 Käyttäjän kysyessä ihmistä auttamaan, hänet ohjataan ICT-yhdys henkilön luokse.
- V2.05 Käyttäjän pyytäessä lisää tilaa sähköpostiin tai Työkansioihin, ohjataan ottamaan yhteyttä esimieheen ja ICT-yhdys henkilöön.
- V2.06 Käyttäjän kysyessä tunnuksista, ohjataan tunnuksia käsittelevään aineistoon.
- V2.07 Käyttäjän kysyessä yleisien sovelluksien toiminnasta, ohjataan yhteiseen koulutusaineistoon ja IT-palveluiden aineistopankin aihetta käsittelevään ohjesivuun.
- V2.08 Käyttäjien kysyessä ydintoiminnan sovelluksien toiminnasta, ohjataan IT-palveluiden aineistopankin aihetta käsittelevään ohjesivuun.
- V2.09 Lisäpyynnöistä ohjataan ottamaan yhteyttä ICT-yhdys henkilöön tai Teamsin lähituki-kanavalle.
- V2.10 Laitteiden katoamistilanteessa ohjataan ottamaan yhteyttä ICT-yhdys henkilöön, esimieheen ja Valtorin ICT-tukeen.
- V2.11 Jos Valtorin ohjeistuksessa on valmista, validia aineistoa, annetaan vastauksessa linkki ohjeisiin, jotta vastaukset pysyvät paremmin ajan tasalla.

Työpajassa todettiin, että sisältöaineisto on hyvin laaja. IT-palveluihin liittyviä aiheita on hyvin paljon ja jokaiseen liittyy yleensä useita erilaisia kysymyksiä. Erilaisia ylätasoa aiheita tunnistettiin noin 30 ja jokaiseen näistä liittyy useita kysymyksiä. Laajimpiin aiheisiin liittyy kymmeniä kysymyksiä. Tämän vuoksi hyväksyttiin se, että projektin aikana kaikkia kysymyksiä ei saada vietyä bottiin ja tavoitteeksi päätettiin ottaa se, että sinne viedään vain kaikkein yleisimmät kysymykset ja niiden liitännäisaiheet.

Botti ja sisältö pitää suunnitella sellaiseksi, että tietohallinto-organisaatio pystyy päivittämään aiheita ja lisäämään niitä lisää nopeasti ja vaivattomasti. Määrittelyssä havaittiin myös se, että useat kulut sisältävät viittauksia samanlaisiin tietoihin, esimerkiksi useassa kulussa on tarvetta kertoa ICT-tuen yhteystiedot. Tämän vuoksi nähtiin tarpeelliseksi komponentoida aiheita niin, että esimerkiksi edellä mainitussa tilanteessa ICT-tuen yhteystieto on oma aiheensa, jota kutsutaan jostain toista aihetta kutsuttaessa. Tällainen komponentointi tehostaa kehittämistä ja sen voi nähdä hyödylliseksi ylläpidossa. Copilot Studio tukee tällaista toimintatapaa, koska työkaluun on rakennettu valmiiksi mahdollisuus kutsua toista aihetta yhdestä kulusta (Microsoft 2024d).

Projektitokouksissa asiantuntijoiden kanssa sovittiin, että kansalaiskehittäjien määrää kasvatetaan ja heidät koulutetaan käyttämään Copilot Studiota. Tarkoituksena on laajentaa sisällön päivittäjien

määrää ja tehostaa botin ylläpitoa. Tarkempi ylläpidon suunnittelu päätettiin jättää myöhäisempään hetkeen odottamaan botin julkaisua ja käyttöönottoa.

2.7 Henkilöstöhallinnon toiminnalliset vaatimukset ja sisältö

Henkilöstöhallinnon toiminnalliset vaatimukset ovat käytännössä samat kuin tietohallinnon. Pääasiallinen ero on toimialueiden sisällössä. Myös henkilöstöhallinnon aiheiden laajuus on suuri, joten niissä päädyttiin tekemään samanlaista komponentointia kuin tietohallinnossa. Yhteiskäyttöisiä aiheita, kuten yhteystietoja, tehtiin omiksi aiheiksi, joiden käyttötarkoitus on pääasiassa se, että niitä kutsutaan muista aiheista.

Henkilöstöhallinnon työpajassa asiantuntijoille esiteltiin Copilot Studion ominaisuuksia. Generatiivista tekoälyä arvioitiin aiemmin toimitetun aineiston perusteella ja asiantuntijoiden kanssa todettiin, että tekoäly antoi liian epävarmoja vastauksia. Tämän vuoksi päätettiin, että kysymykset ja vastaukset syötetään botille käyttäen työkalun aiheominaisuutta. Generatiivista tekoälyä testataan aiheiden rinnalla, mutta tarkoitus ei ole tässä vaiheessa hyödyntää sitä laajasti. Työpajassa sovittiin, että asiantuntijat valmistelevat muutamia peruskysymyksiä ja niiden kulkuja. Saatuaan kulkujen dokumentaation valmiiksi, ne syötettiin botille.

Myöhemmin pidetyissä tarkistuskokouksissa tarkasteltiin määrittelyn ja toteutuksen tuloksia ja sovittiin jatkoaskeleista. Aineiston syöttämistä botille edistettiin myös sillä, että yhdelle asiantuntijalle hankittiin tunnukset ja käyttölisenssit Power Platform –alustalle, jolloin hän pystyi lyhyen opastuksen jälkeen itse viemään sisältöä alustalle. Tämä myös mahdollisti asiantuntijan tekemän botin itsenäisen testauksen.

Jatkokokouksissa sovittiin myös, että kaikkia aiheita ei ole tarpeellista saada projektin yhteydessä vietyä bottiin, vaan sen voi jättää osin puutteelliseksi. Syynä tähän oli se, että botin käyttöönotto ei ollut projektin päättymisen kannalta välittömästi ajankohtaista, vaan se odotti vasta tulevaisuudessa. Oleellista oli saada bottiin jonkin verran sisältöä, jotta sen toimivuutta pystyi testaamaan ja esittelemään toisille. Julkaisu ja käyttöönotto on projektin rajauksien mukaisesti myöhemmässä vaiheessa suoritettava tehtävä.

Työpajoissa määritellystä sisällöstä laadittiin vaatimukset, jotka ovat tiivistetysti seuraavat:

- V3.01 Käyttäjää on ohjattava työaikaleimauksista, jotta leimaukset tehdään oikein ja tieto on luotettavaa
- V3.02 Käyttäjälle on tarjottava malleja työaikaleimauksien tekemiseen, koska leimauskäytännöt ovat monimutkaisia

- V3.03 Käyttäjiä on ohjattava sairauspoissaolojen käsittelystä, jotta he osaavat merkitä niitä oikein ja ymmärtävät merkintöjen tarkoituksen.
- V3.04 Käyttäjiä on ohjattava henkilöstöhallinnon toiminnasta, jotta he ymmärtävät käytäntöjen merkityksen
- V3.05 Käyttäjiä on ohjattava osa-aikaisuuskäytännöistä, jotta he osaavat tehdä oikeita päätöksiä harkitessaan osa-aikaisuutta ja ymmärtävät omat oikeutensa
- V3.06 Käyttäjiä on ohjattava vuosilomien käytännöistä, jotta he osaavat hakea lomia oikein ja ymmärtävät omat oikeutensa
- V3.07 Käyttäjiä on ohjattava työsuhde-eduista ja siihen liittyvistä järjestelmistä, jotta he osaavat käyttää etujaan oikein
- V3.08 Ohjeiden päivittämisen pitää olla helppoa, koska ohjeet voivat muuttua usein ja käyttäjien on saatava välittömästi ajan tasaista tietoa

3 Keskustelubotin toteutus

Luku käsittelee projektinaikaista keskustelubotin toteuttamista. Ensimmäiset aliluvut kuvaavat Copilot Studion ominaisuuksia, joita toteutustyössä käytettiin. Erityistä huomiota kiinnitetään työkalun muuttajaominaisuuksiin ja generatiiviseen tekoälyyn. Seuraavissa aliluvuissa käsitellään konkreettista toteutustyötä ja sitä, miten ominaisuuksia hyödynnettiin projektissa. Viimeisessä aliluvussa käsitellään sitä, miten keskustelubottia testattiin Copilot Studion testaustyökaluilla.

3.1 Työkalun ominaisuudet

Keskustelubottia voi lähteä rakentamaan Copilot Studiolla muutamasta eri näkökulmasta. Yleisin tapa on hyödyntää generatiivista tekoälyä ja toinen tapa on viedä kysymykset ja vastaukset bottiin aiheina. Aiheina toteutetun ratkaisun etuna on se, että vastaukset ovat varmasti juuri sellaisia, kuin käyttäjälle halutaan annettavan. Huono puoli aiheiden laajassa käytössä on ylläpidon työläisyys.

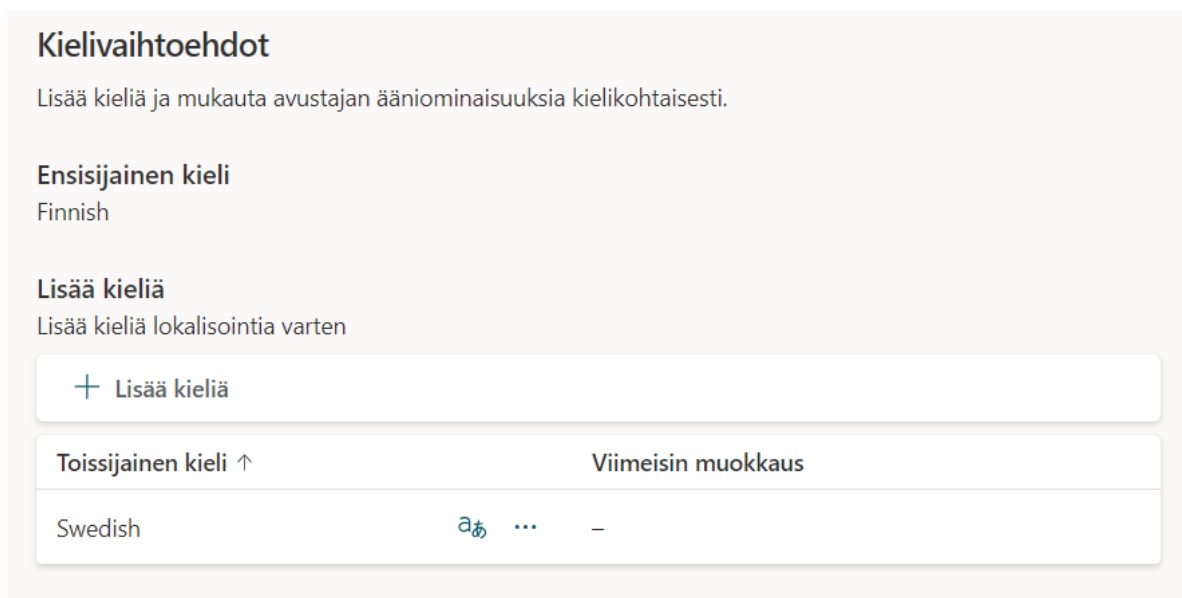
Molempia tapoja voi laajentaa Copilot Studion keskustelu- ja tekoälylaajennuksilla. Keskustelulaajennuksilla voi luoda bottiin yleiskäyttöisiä aiheita, jotka voivat olla käytössä useassa eri botissa. Tekoälylaajennuksilla voi hyödyntää generatiivisen tekoälyn lisäksi muita tekoälypohjaisia toimintoja. Näitä toimintoja on käytettävissä Power Platformin tuotteissa ja niitä voi myös kehittää itse esimerkiksi Power Automatea hyödyntämällä. Esimerkiksi, jos botin käsittelemä aineisto olisi saatavilla vain englanniksi, mutta syötteet tulisivat suomeksi, voisi botin käskellä käyttämään tekoälyä syötteen kääntämiseen englanniksi, generoimaan englanninkielisen vastauksen ja sitten kääntämään vastaus jälleen suomeksi. Tiedon käsittely tapahtuisi siis käytännössä englanniksi, mutta käyttäjälle jäisi kokemus suomenkielisestä botista. (Microsoft 2024e.)

Keskustelubotti olisi mahdollista perustaa valmista mallia käyttäen. Microsoft on julkaissut UKK-tyylisen botin mallin Githubissa (Microsoft 2024f). Tämä oli mahdollinen ratkaisu tämän työn käyttötapauksessa, mutta koska osa työn tavoitetta on oppia käyttämään työkalua edistyneesti, päätettiin tekemään botti tyhjältä pohjalta.

Copilot Studio ja Power Platform tukevat luotujen ratkaisujen hallintaa tiedostoina. Palveluun luotu botti on mahdollista viedä sisältöineen tiedostoksi yhdestä ympäristöstä ja tuoda tiedostona toiseen ympäristöön. Näin ratkaisuja voi siirtää esimerkiksi kehitysympäristöstä testaukseen tai tuotantoon. Samalla tavalla kehitetyistä ratkaisuista voi ottaa varmuuskopioita. (Microsoft 2024g.)

Copilot Studio tukee suomen kielistä kehittämistä nykyään myös generatiivisen tekoälyn työkalujen avulla. Aiemmin, Copilot Studion käyttöliittymässä generatiivisen tekoälyn kohdalla oli selkeä maininta siitä, että suomea ei tueta sen käytössä. Tammikuun 2024 jälkeen tämä merkintä on poistettu. Lisäksi tuki suomella on mainittu dokumentaatiossa. (Microsoft 2024h).

Copilot Studiossa on tuki monikielisille toiminnalle (Microsoft 2024i). Jatkokehityksessä tätä toimintoa hyödyntäen pitäisi keskustelubotissa tarjota tuki ruotsinkielisille käyttäjille. Tämä on melko suuritöinen tavoite, joka on käyttäjien yksipuolisuuden varmistamiseksi kuitenkin toteutettava. Kuvassa 1 on nähtävissä Copilot Studion kieliasetukset. Asetuksiin on valittavissa ensisijainen kieli ja lisäkieliä toissijaisiksi. Kuvan mukaisesti toteutuksessa olevaan bottiin on valittu ensisijaiseksi kieleksi suomi ja toissijaiseksi ruotsi, vaikka ruotsia ei raportin kirjoitushetkellä botissa käytetäkään.



Kuva 1. Copilot Studion kieliasetukset.

3.1.1 Generatiivinen tekoäly

Generatiivinen tekoäly hyödyntää sisäisiä ja ulkoisia lähteitä selvittääkseen mitä käyttäjä on siltä kysynyt ja koostaakseen selkokielisen vastauksen. Sisäisiä lähteitä voivat olla asiakirjatiedostot, joihin botille on annettu pääsy. Tiedostojen käyttöä voi tarvittaessa kontrolloida aihekohtaisesti. Ulkoisia lähteitä ovat Azure OpenAI -tekoäly tiedot ja Bing-hakukoneen hakuosumat. (Microsoft 2024j.)

Botti pyrkii ensisijaisesti hakemaan vastauksen sille syötettyjen aiheiden perusteella, mutta jos se ei löydä niistä soveliaista vastausta, se pyrkii muodostamaan vastauksen sille annettujen lähteiden perusteella. Sisäisiä lähteitä ovat botille syötetyt tiedostot, internetsivut, Dataverse-tietokannat ja

Power Platform -toiminnot sekä edellä mainitut ulkoiset lähteet. Botin vastaus ilmoittaa, kun vastaus on generatiivisen tekoälyn luoma ja käyttäjä voi antaa nopeaa palautetta näille vastauksille peukalo ylös tai alas –kuvakkeilla. (Microsoft 2024k.)

Kun projektin keskustelubottia alun perin suunniteltiin, alustalla ei ollut käytössä suomenkielistä generatiivista tekoälyä ja alkuperäinen suunnitelma oli toteuttaa botti pelkästään käyttäen Copilot Studion aiheet-toiminnallisuutta. Suomen kielen tuki Copilot-tekoälylle kuitenkin julkaistiin tammi-kuussa 2024. Aineiston käsittely ja syöttäminen botille olisi tehokkaampaa generatiivista tekoälyä hyödyntämällä. Generatiivisen tekoälyn saa toimimaan samalla tavalla kuin aiheet syöttämällä tekoälylle tiedoston, jossa on sisältöä (otsikkoja, kysymyksiä, vastauksia) samankaltaisessa muodossa. Tällainen tiedostopohjainen sisällön syöttäminen on nopeampaa ja helpommin joukkoistetavissa kuin aiheiden syöttäminen. Pääasiallinen ongelma tässä on sama kuin muissakin generatiivisen tekoälyn toiminnoissa eli vastauksien sisältö ei ole täysin kontrolloitavissa ja ne voivat olla huonolaatuisia. Generatiivisen tekoälyn käyttäminen aiheuttaa myös ajoittain botissa hallusinoitua ja vääriin asioihin vastaamista. (Microsoft 2024k.)

Käyttäjän on mahdollista antaa palautetta generatiivisen tekoälyn luomiin vastauksiin. Jokaisen viestin ohessa näkyy oletuksena peukalo ylös ja alas –ikonit, joista voi käyttäjä voi ilmoittaa oliko vastaus hyödyllinen vai ei. Tämä kouluttaa tekoälyä ja sen pitäisi johtaa jatkossa parempiin generatiivisiin vastauksiin. (Microsoft 2024k.)

Tekoälyn käytön suunnittelussa on huomioitava vastuullisuus siitä näkökulmasta, että ihmisen olisi aina valvottava ja vahvistettava tekoälyn toimenpiteitä. Tekoälyn vastaukset eivät aina ole oikein, joten jos tekoälylle antaa vallan suorittaa toimenpiteitä ilman ihmisen vahvistamista, voivat myös toimenpiteet ja niiden seuraukset olla väärinä. (Urrutia 2024.)

Projektiin toteutettavan botin käyttäjät eivät ole erityisen IT-orientoituneita, joten on varauduttava siihen, että botille esitettävät kysymykset voisivat olla vaikeita tekoälyn hallittavaksi. Lisäksi opetusaineistoa ei ole välttämättä saatavilla riittävää määrää sellaisessa muodossa, että Copilot Studion generatiivinen tekoäly osaisi sitä tarpeeksi hyvin käsitellä. Käytännön testien perusteella Copilot Studion käyttämän Azure OpenAI:n vastaukset eivät ole kovinkaan hyviä. Vastaukset ovat pääasiassa kopioita opetusaineistossa olevasta tekstistä ja yleensä liian laseja vastatakseen selkeästi tehtyyn kysymykseen. Myös tekoälyn vastauksiin liittämät viitteet ovat tarpeettomia ja jossain määrin häiritseviä.

3.1.2 Muuttujat

Copilot Studio -työkalu mahdollistaa muuttujien käyttämisen keskustelujen ohjaamisessa. Keskusteluaiheissa on mahdollista käyttää globaaleja ja lokaaleja muuttujia sekä entiteettejä. Jokaista näistä voi käyttää eri tavoin muuttujina keskustelujen hallintaan.

Entiteetit ovat muuttujamaisia bottikeskustelun komponentteja, joita ei perinteiseen malliin aseteta kulkuun, vaan työkalun tekoälyominaisuudet tunnistavat ne sille syötetyistä kehoitteesta. Microsoftin kuvauksen mukaisesti entiteetti kuvaa "tietoyksikköä, joka edustaa tietyn tyyppistä todellisen maailman kohdetta". Esimerkkeinä entiteeteistä ovat puhelinnumerot, postinumeroit sekä kaupunkien ja henkilöiden nimet. Entiteetit pitää ensin syöttää työkalulle, jotta työkalu osaisi käyttää niitä. (Microsoft 2024I.)

Globaalina muuttujana toimimisen lisäksi entiteeteillä on myös tehostavaa merkitystä botin toiminnassa. Entiteetin osaavat hyödyntää luonnollisen kielen käsittelyä ja siten tunnistavat paremmin syötteissä olevaa dataa sekä muuntavat sen työkalun edellyttämään muotoon. Tämä tekee botin käytön helpommaksi, kun käyttäjälle annetaan vapautta esittää asiansa omin sanoin sen sijaan, että hän joutuisi syöttämään tiedot tarkasti järjestelmän edellyttämässä muodossa. (Rivera 2023, luku 10.)

Työkalussa on tallennettuna yleisiä entiteettejä, joiden hyöty tässä projektissa oli kuitenkin hyvin rajallinen. Toteutuksen aikana entiteeteinä työkaluun syötettiin toimeksiantajan organisaation rakenne ja tietohallintoon liittyviä entiteettejä.

Entiteettiä lisättäessä, sille annetaan kuvaava nimi sekä syötetään suljettuina luetteloina tai säännöllisenä lausekkeena nimeen liittyvät tiedot. Projektissa käytettiin vain luetteloja, mutta esimerkiksi IP-osoitteita tai muita rakenteeltaan säännöllisiä tietoja, voisi viedä entiteettiin säännöllisinä lausekkeina hyödyntäen tavanomaisia regex-sääntöjä. Tietoja vietäessä luettelona, syötetään entiteettiin tietoja saman kaltaisella ajatusmallilla kuin mitä tehtäisiin relaatiotietokannan taulua täyttäessä. Luetteloon syötetään aiheen mukaiset termit, joita tekoäly tunnistaa kyseisen entiteetin nimikkeiksi. Nimikkeille voi myös syöttää synonyymejä. Merkittävä ero relaatiokantaan verrattuna on tietenkin se, että entiteettien välillä ei ole relaatioita, eikä siksi myöskään avainmuuttujia.

Copilot Studio tukee globaaleja muuttujia, joita hyödyntäen bottiin voi tehdä ehdollisia kulkuja. Muuttuja syötetään osana keskustelun työnkulkua ja pitää ensin alustaa. Jos muuttuja on asetettu globaaliksi, voivat toiset kulut käyttää sitä. Parhaiden käytäntöjen mukaan alustus kannattaa tehdä mahdollisimman aikaisessa vaiheessa keskustelua, esimerkiksi suositeltua on tehdä alustus kes-

kustelun avausviestin yhteydessä. Alustus ei näy botin käyttäjälle. Muuttujia voi myös käsitellä kukohtaisesti eli globaalin sijasta ne ovat alustettu lokaaleiksi. Tällainen muuttuja asetetaan ja käsitellään yhden kulun aikana. Tällöin on kuitenkin huomioitava, että alustus suoritetaan joka kerta, kun kulku käynnistyy. (Microsoft 2024m.)

Copilot Studio hyödyntää järjestelmämuuttujia, jotka käsittelevät keskustelun kulkujen transaktioita, joista Microsoft käyttää nimitystä tapahtuma. Työkalu käyttää järjestelmäkulkuja keskustelukulkujen teknisessä toteutuksessa ja niillä voi noutaa lähinnä teknistä tietoa keskustelusta (Microsoft 2024m). Näitä muuttujia ei voi muuttaa, mutta niitä voi hyödyntää kehitystyössä. Järjestelmämuuttujien hyöty keskusteluissa ja viesteissä on kovin rajattua, mutta toteuttamisessa niistä voi olla hyötyä, esimerkiksi virheenkorjausta tehtäessä. Järjestelmämuuttujilla voi olla ainakin teoriassa käyttöä ehdollisten kulkujen ohjaamisessa.

3.2 Alustan valmistelu toteutusta varten

Projektin keskustelubotti luodaan oikeusministeriön hallinnonalan Power Platform -ympäristöön, jonka on pystyttänyt hallinnonalan tietoteknisten palveluiden toimittaja Oikeusrekisterikeskus. Ympäristön pystytys ja tarkka kuvaus eivät ole opinnäytetyön päätavoitteiden kannalta merkittävää ja voivat sisältää tietoa, joka paljastuessaan voisi vaarantaa ympäristön turvallisuuden, joten ympäristön tekninen kuvaaminen rajataan raportin ulkopuolelle. Riittävä kuvaus on se, että Copilot Studiolla tuotettava keskustelubotti on luotu käytössä olevaan kehitysympäristöön.

Copilot Studio mahdollistaa bottien kopioimisen ympäristöjen välillä (Microsoft 2024g), joten botin julkaiseminen tapahtuu kopioimalla kehitysympäristössä oleva iteraatio tuotantoympäristöön sekä suorittamalla julkaisuaskeleet tässä ympäristössä.

Kuvassa 2 näytetään botin luomisvaiheen asetukset. Copilot Studio käyttää tässä näkymässä botista nimitystä Avustaja. Botin tietämysosoitteeksi asetettiin kehittämissympäristön Sharepoint-kansio, jonne tallennetaan tekoälyn käsiteltäväksi tarkoitettua aineistoa (osoite on turvallisuusyistä piilotettu kuvasta). Valmisteluvaiheessa botin kieleksi asetettiin suomi ja nimeksi "SY bot (dev)". Lisäasetuksissa botille syötettiin Bing Chat -tekoälyllä generoitu kuva (tämä ei näy kuvassa). Muutoin luomisvaiheessa käytettiin oletusasetuksia.

Avustajan nimi * ⓘ

SY bot (dev)

Mitä kieltä haluat avustajan puhuvan? * ⓘ

suomi

🔗 Anna avustajalle tietämystä määrittämällä generatiivinen tekoäly ⓘ

Lisää tietämystä avustajaasi osoittamalla sille ulkoinen verkkosivusto, jotta se voi välittömästi vastata kysymyksiin sivuston tietojen perusteella. [Lisätietoja](#)

https://[REDACTED]

Tekoälyn luomassa sisällössä voi olla virheitä, joten älä unohda varmistaa, että se on tarkka ja asianmukainen. Lisätietoja: [lisäehdot](#).

Muokkaa lisäasetuksia > **Luonti** Peruuta

Kuva 2. Keskustelubotin luomisnäkökulma Copilot Studioissa (url-osoite piilotettu tietoturvasyistä)

Alustuksen yhteydessä bottiin ei syötetty muita tietoja, vaikka esimerkiksi työkalun entiteetteihin olisi voinut viedä toimeksiantajan osasto- ja yksikköjaon. Tässä vaiheessa toteutusta ei ollut vielä selvää, että niille on tarvetta. Tarve havaittiin kuitenkin myöhemmin ja mm. osastojako vietiin entiteetteihin. Määrittelystä, jota tehtiin yhtäaikaaisesti alustuksen ja toteutuksen kanssa, nousi myös muita vastaavia tarpeita, joita toteutettiin sekä entiteeteillä että globaaleilla muuttujilla.

3.3 Teknisten vaatimusten toteutus

Copilot Studio on työkaluna tarkoitettu nimenomaan keskustelubottien rakentamiseen, joten itse työkalu ei edellytä merkittäviä konfigurointeja kehittämisen aloittamisessa. Aiemmin raportissa kuvatut tekniset vaatimukset täytyvät lähtökohtaisesti. Konfigurointia tarvittiin vain muutamien sisällön valmisteluun liittyvissä asioissa.

Copilot Studion entiteetteihin luotiin muutama selkeä uusi tapaus, kuten organisaation rakennetta kuvaavat entiteetit. Entiteettejä on raportin kirjoitushetkellä käytetty vain muutamissa aiheissa ja niiden hyödyntäminen eri aiheiden välillä on vielä rajoittunutta.

Copilot Studiolla luotuun bottiin on sisäänrakennettu ns. järjestelmäaiheet. Järjestelmäaiheet ohjaavat botin teknisiä ja yleisiä toimintoja, kuten keskustelun aloittamista ja lopettamista. Niitä ei voi poistaa, mutta yksittäisen järjestelmäaiheen voi sammuttaa, jos sille ei ole käyttöä. (Microsoft 2024n.)

Järjestelmäaiheet eivät oletussisältöisenä sovellu julkaistun botin käyttöön, joten niiden viestisisältöä täytyi muuttaa. Botin on syytä noudattaa organisaatiolle ja käyttötilanteihin sopivaa luonnetta

(Goasduff 2019), joten koska botti on tehty sisäiseen käyttäjiä ohjeistavaan käyttöön, viesteistä tehtiin vähemmän muodollisia ja organisaatioon itseensä viittaavia.

Generatiivisen tekoälyn asetuksissa sisällön moderointi asetettiin Korkeaksi, jotta botin vastaukset perustuisivat pääasiassa vaatimusten mukaiseen syötettyyn sisältöön. Sisältö ja botilta halutut vastaukset ovat hyvin organisaatiospesifisiä, joten tekoälyn ei haluta juurikaan käyttävän ulkoista (hallitsematonta) tietoa vastauksissaan. Alkuperäisissä testeissä tällaisen tiedon käyttö aiheutti lähes pelkästään vääriä vastauksia, koska botti haki tietoa avoimesta verkosta, eikä käyttänyt sille tarjottua aineistoa. Käytännössä asetuksen Korkeaksi asettaminen ei johtanut myöskään riittävään lopputulokseen. Botin palauttamat generatiiviset vastaukset muodostettiin sille annetusta aineistosta karkeasti arvioiden 50 % tapauksista ja loppuissa käytettiin ulkoista aineistoa, joka sisälsi väärää tietoa. Botin tekemissä generatiivisissa vastauksissa oli usein viitteitä sille syötettyyn tiedostoaineistoon, mutta viitteet olivat joissain tapauksissa vääriä ja turhia. Kaiken kaikkiaan generatiivisesti muodostetut vastaukset koettiin huonoiksi.

Koska generatiivisen tekoälyn haluttiin pystyvän hakemaan tietoa Sharepoint-jaosta, asetettiin Suojaus-asetuksista Todentaminen aluksi toimimaan Vain Teamsille ja Power Appsille kuvan 3 mukaisesti. Tämä mahdollistaa lisäksi botin julkaisemisen aikanaan Teamsiin. Haittana tästä asetuksesta on se, että bottia ei voi testausmielessä julkaista erillisenä nettisivuna ainakaan ilman erityisiä konfiguraatiomuutoksia.

Todentaminen ×

Tarkista käyttäjän käyttäjätiedot keskustelun aikana. Botti saa suojatun käyttöoikeuden käyttäjän tietoihin ja voi tehdä toimintoja käyttäjän puolesta, mikä tuottaa yksilöllisemmän käyttökokemuksen. [Lisätietoja](#)

Valitse vaihtoehto

- Ei todentamista**
Botin perusmäärittäminen, jossa ei ole todennustoimintoa tai todennusmuuttujia.
- Vain Teamsille ja Power Appsille**
Käytettävissä olevat UserID- ja UserDisplayName-todennusmuuttujat. Määrittää automaattisesti Azure Active Directory (AAD) -todennuksen Teamsille ja Power Appsille. Kaikki muut kanavat poistetaan käytöstä. [Lisätietoja](#)

- Manuaalinen (mukautetulle verkkosivustolle)**
Tue AAD-todennusta tai mitä tahansa OAuth2-tunnistetietojen toimittajaa. Käytettävissä ovat todennusmuuttujat, kuten todennustunnus.

Syötä tunnistetietojen toimittajan (IdP) tarjoamat tiedot ja testaa yhteys. Jos haluat käyttää kertakirjautumista AAD:n avulla, sisällytä tunnuksen vaihdon URL-osoite. [Lisätietoja](#)

Vaadi käyttäjiä kirjautumaan

Kuva 3. Todennusasetukset

Copilot Studio sisältää myös laajennusominaisuuksia, joilla keskusteluiden ja tekoälyn toiminta voisi olla kehittyneempää. Tässä tilanteessa näitä ei kuitenkaan päädytty käyttämään, mutta niissä on nähtävissä hyviä mahdollisuuksia jatkokehittämistä varten.

3.4 Toiminnallisten vaatimusten toteutus

Henkilöstöhallinto toimitti aineistoa, joka vietiin generatiivisen tekoälyn saataville. Osaan vaatimuksista ei ollut saatavilla valmiina aineistoa, jota tekoäly osaisi hyödyntää tai josta saisi botille soveltuvia vastauksia. Näiden aineistojen osalta botille syötettiin tietyistä asioista aineistoa aiheina. Tietohallinnon ja IT-palveluiden vaatimuksien toteuttaminen ja sisällön syöttö tapahtui samalla tavalla.

Vaatimukset ja sisältö oli ennen syöttämistä pilkottu aihealueisiin ja aihealueen yksittäisiksi kysymyksiksi, joita käyttäjien odotettiin esittävän botille. Kysymykset syötettiin botille yksittäisinä aiheina. Aiheille asetettiin aloituslauseita, joiden perusteella botti ymmärtää vastata kyseisen aiheen mukaisesti. Vastaukset syötettiin suurimmassa osassa aiheita kahtena tai kolmena erillisenä viestinä luettavuuden varmistamiseksi. Joissain aiheissa toteutettiin pidempiä kulkua ja ehtolauseita, jotta yhdellä aiheella saatiin hoidettua laaja kysymysskala. Esimerkiksi puhelinlaitteiden käyttöä koskevat aiheet sisältävät käytännössä kaksi kulkua. Kulun alussa kysytään, onko kysyjällä käytössä Android- vai iOS-laite, ja vastauskulku jatkuu tämän valinnan mukaisesti. Periaatteessa molemmille laitteille olisi voinut tehdä omat aiheet kaikille kysymyksille, mutta kysymyslimiitin ja ylläpidon näkökulmasta on parempaa, että ne käyttävät yhteistä kulkua.

Aiheiden suuri määrä johtaa aloituslauseiden suureen määrään ja myös niiden päällekkäisyyteen. Päällekkäiset lauseet aiheuttivat ajoittain botille hämmennystä ja tarvetta kysellä tarkentavia kysymyksiä. Käyttäjän kannalta kysymykset ovat turhauttavia, joten suunnittelun näkökulmasta on parasta, että niitä ei pääsisi syntymään. Tämä aiheutti tarvetta jatkuvalla testaamisella ja lauseiden korjaamiselle.

Aiheiden tueksi alustaan syötettiin myös entiteettejä kuvaamaan organisaatorakennetta, toimialueita ja muita selkeitä valintatilanteita. Kuvassa 4 on esimerkki toimialuejaosta, johon on syötetty kaikki hallinnollisten asioiden toimialueet, joita toimeksiantajalla on organisaatiossaan. Entiteeteillä pyritään tekemään botin toimintaa ketterämmäksi niin, että sillä ei tarvitse kertoa samoja tietoja useaan kertaan. Käytännössä kaikkia entiteettejä ei tämän hetken toteutuksessa hyödynnetä, mutta niillä nähdään olevan käyttöä jatkossa.

Toimiala

Nimi *

Kuvaus

Menetelmä
 Luettelo
 avustaja yrittää löytää vastaavuuden luettelokohteeseen asiakkaan sanojen perusteella.

Muokattu
 muutama sekunti sitten

Älykäs kohdistaminen
 käytössä
 Älykkään kohdistamisen avulla avustaja voi vmmärtää luonnollista kieltä. Tämä voi auttaa

Luettelokohteet

Syötä kohde Lisää

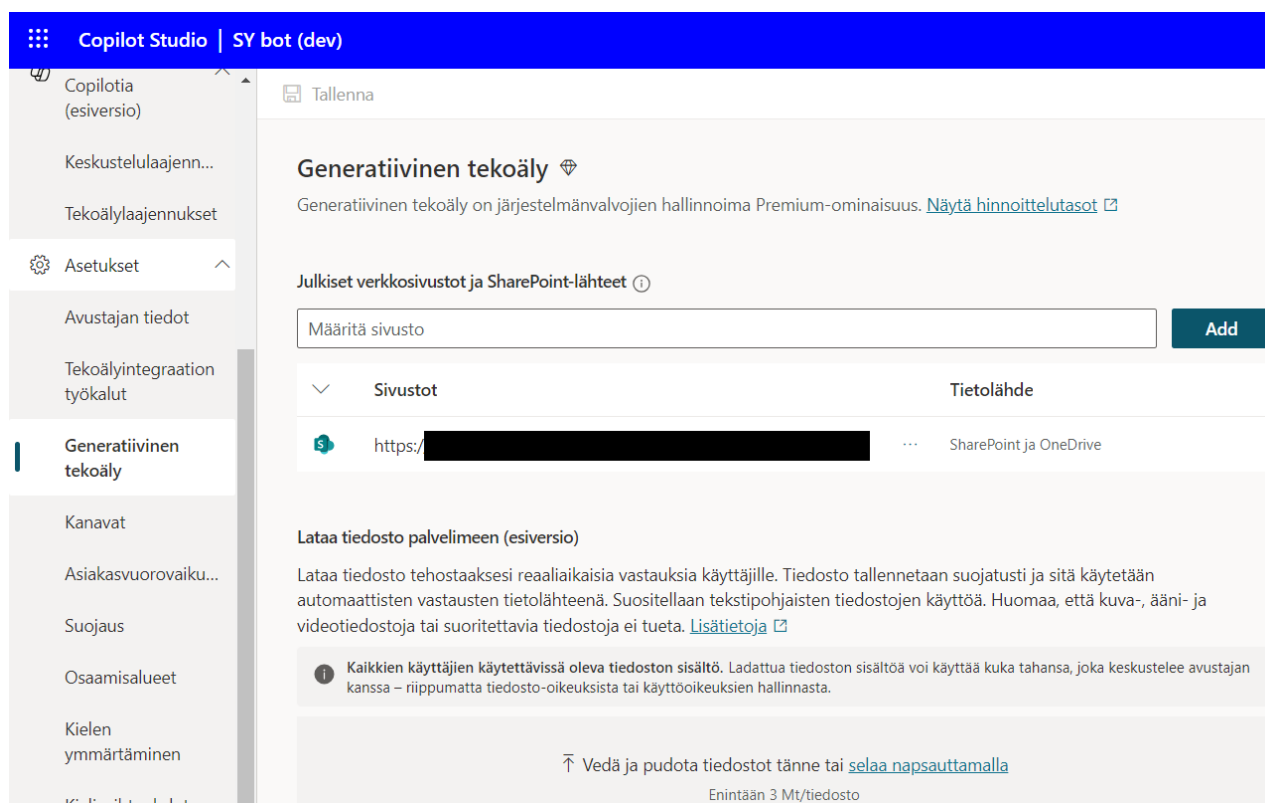
Kohde	Synonyymit
Tietohallinto	ICT, IT, IT-palvelut, ICT-palvelut
Henkilöstöhallinto	heha, henkilöstöasiat
Taloushallinto	taha, talousasiat
Matkahallinto	matkat
Viestintä	tiedotus
Johdon tuki	+ Synonyymit

Tallenna Sulje

Kuva 4. Esimerkki toimialaentiteetistä.

3.4.1 Generatiivisen tekoälyn hyödyntäminen

Copilot Studio pystyy hyödyntämään generatiivista tekoälyä lähdeaineistojen tulkitsemiseen. Työkaluun voi syöttää osoitteita ulkoisiin internetsivuihin tai sisäisiin sivustoihin, joihin sille on annettu pääsy, joista tekoäly pyrkii hyödyntämään sisältöä vastauksia antaessaan. Sivujen lisäksi työkalulle voi antaa tiedostoja, joita se lukee ja hyödyntää samalla tavalla. Dokumentaation perusteella tiedostoina voi käyttää tavanomaisia Office-tiedostomuotoja, mutta käytännön havainto on, että docx-muotoiset tekstitiedostot toimivat kaikkein parhaiten. Tekoälylle syötettävät tiedostot voi julkaista sivustolle, kuten esimerkiksi Sharepoint-ympäristöön, ja tekoälyn voi ohjata lukemaan ne sivuston osoitteesta yksittäisten tiedostojen sijaan. Tämä helpottaa ylläpitoa, kun uutta aineistoa voi lisätä viemällä tiedostoja sivustolle. Kuva 5 on näytönkaappaus toteutuksen generatiivisen tekoälyn asetuksista, jossa botille on syötetty Sharepoint-jako osoitteeksi. Osoite on sama kuin mikä syötettiin kuvassa 1. Tässäkin kuvassa osoite on piilotettu tietoturvasyistä.



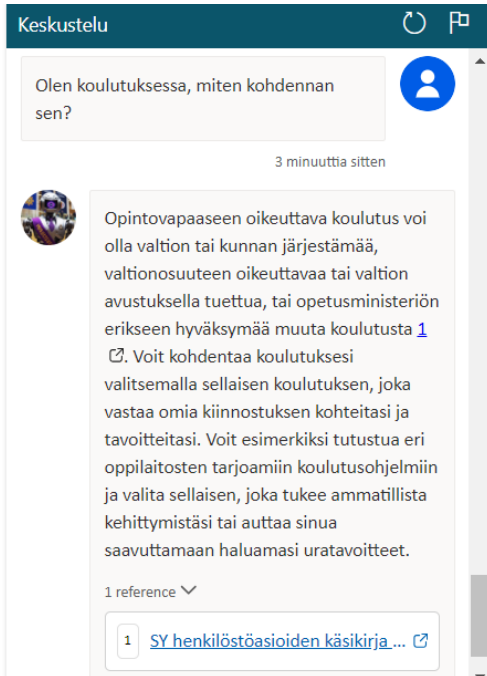
Kuva 5. Generatiivisen tekoälyn asetukset.

Käytännön toteutuksessa kuitenkin huomattiin, että botti ei osannut kovinkaan hyvin käsitellä sille annettua aineistoa. Generatiivinen tekoäly hallusinoi melko paljon ja antoi erittäin huonoja vastauksia. Jos tekoälyn aineistoa yritti rajata, se ei enää osannut vastata kysymyksiin. Tekoälyllä vaikutti olevan melko suuria vaikeuksia ymmärtää kysymyksiä ja sille annettua aineistoa. Syitä tekoälyn tähän toimintaan ei ole varmasti tiedossa, mutta osa vastauksista viittasi siihen, että se käsiteli kysymyksiä ja aineistoa kääntämällä ne ensin englanniksi. Tämä mahdollisesti aiheutti epäselvyyksiä kysymyksiin ja vastauksiin. Tämä erityisesti näkyi siinä, miten tekoäly sekoitti kahden täysin erillään olevan aineiston vastauksia keskenään.

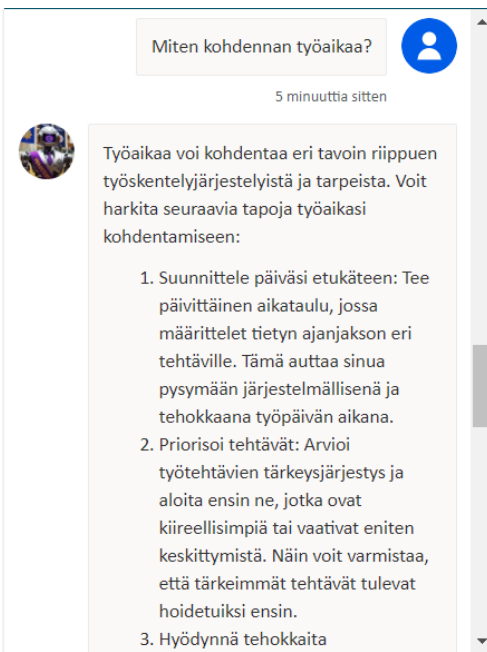
Tekoäly myös priorisoi vastauksissaan avoimet verkkosivut tiedostoaineiston sijaan, vaikka juuri päinvastainen priorisointi olisi ollut parempi. Tähän olisi mahdollista vaikuttaa konfiguraatioita muuttamalla, mutta käytännössä tekoälyn kyky lukea tiedostoja vaikutti liian heikolta tähän tilanteeseen. On mahdollista, että tiedostoaineiston sisältö ei ollut sopivaa tekoälyn käyttöön. Testaukset toisella aineistolla, joka rakennettiin nimenomaan tekoälyn käytettäväksi, ei kuitenkaan johtanut parempiin tuloksiin.

Kuvissa 6 ja 7 on kaksi yleistä henkilöstöhallintoon kohdistuvaa kysymystä, joihin liittyen botille oli syötetty aineistoa tekoälyn generoimia vastauksia varten. Kuvassa 6 botti on vastannut aineiston

perusteella, mutta aivan väärään kysymykseen. Kuvan 7 vastaus ei hyödynnä aineistoa, vaan se on generoitu avoimen verkon tietojen perusteella, eikä vastaus noudata lainkaan toimeksiantajan linjaa. Vastaukset tällaisenaan eivät olisi hyödyllisiä käyttäjälle.



Kuva 6. Generatiivisen tekoälyn muodostama vastaus, joka vastaa väärään kysymykseen.



Kuva 7. Epäonnistunut avoimen verkon tiedoista muodostettu vastaus.

3.4.2 Aiheiden syöttäminen

Aiheita syöttämällä botille voi antaa avainsanoja, joita botti käyttää käynnistimenä kysyjälle annettavaan vastaukseen. Käytännössä aiheiden käyttöä voi ajatella jos-lauseina niin, että jos kysymyksessä on sana x, niin vastataan vastaus y. Jos useassa aiheessa on sama avainsana, botti osaa automaattisesti kysyä mitä asiaa käyttäjä tarkoittaa. Tämän vuoksi aiheet on syytä nimetä järkevästi, koska botti nimenomaisesti tarjoaa vaihtoehtoja aiheiden nimien perusteella. (Rivera 2023, luku 7.)

Yksi aihe sisältää keskustelusolmuja, joista käynnistin on ensimmäinen. Käynnistintä seuraa yleensä viestejä käyttäjälle, jotka pyytävät tarkennusta tai antavat vastauksen, mutta solmut voivat tehdä muutakin. Solmu voisi esimerkiksi muuttaa keskustelukohtaisen muuttujan asetusta tai käynnistää jonkin toisen Power Platform tuotteen toiminnan. (Microsoft 2024c.)

Keskustelusolmu voi myös ohjata keskustelun toisen aiheeseen, joka suoritetaan siinä olevien solmujen mukaisesti ja jonka päätyttyä alkuperäisen aiheen solmut jatkuvat. Tällä tavalla voi esimerkiksi viitata johonkin useissa aiheissa toistuvaan tietoon, kuten jonkin organisaation yhteystietoihin. (Microsoft 2024d.)

Toteutus aihepohjaisesti on melko työlästä, koska kaikki aiheet, niiden käynnistinsanat ja vastauksien kulut ehtoineen on syötettävä työkaluun käsin. Aiheiden syöttämisessä voi hyödyntää tekoälypohjaista generaattoria, mutta käytännössä sen tuotokset ovat laadultaan niin huonoja, että ne täytyy kirjoittaa kokonaan uusiksi. Generaattorin hyöty on siinä, että generaattori luo valmiin rakenteen, jota yleensä voi ainakin osin hyödyntää. Generaattorin kyky käsitellä suomea on melko rajoitunutta kirjoitushetkellä. (Microsoft 2024d.)

Työläydestä huolimatta projektissa päädyttiin botin luomiseen ja sisällönsyöttöön ensisijaisesti aihepohjaisesti, koska generatiivisen tekoälyn toiminta koettiin testauksessa liian epäluotettavaksi. Jatkokehityksessä generatiivisen tekoälyn mahdollisuuksia on syytä selvittää lisää. Mahdollisesti tulevaisuudessa, teknologian kehittyessä, on tehtävä sisällön migraatio tekoälypohjaiseen muotoon.

Aiheita arviottiin syötettävän melko suuri lukumäärä, joten ylläpidon helpottamiseksi aiheiden nimiin laitettiin luokittelu toimialan mukaan kuvan 8 mukaisesti. Tietohallinto-aiheisiin asetetaan merkintä ICT ja henkilöstöhallintoaiheisiin merkintä HEHA. Toteutuksen aikana tehtyjä testailuja varten käytetään merkintää TEST. Jos aihealueita mahdollisesti laajennetaan tulevaisuudessa, pitää uudet alueet merkitä samalla logiikalla, esimerkiksi taloushallinto merkinnällä TAHA.

Aiheet ①

Määritä vuoropuhelun logiikka, jotta voit hallita paremmin ennakoitavissa olevia skenaarioita.

Kaikki **Aiheet (50)** 🗂️ Järjestelmä (9)

Nimi	Käynnistin	Kuvaus	Virheet	Muokkaus	Muokannut
HEHA Vanhenpainvapaa	Lauseet				JS Jukka-Pekka Sirviö kaksi kuukautta sitten
HEHA verokortti (päivitys kesken)	Lauseet				
HEHA Veroprosentti	Lauseet				
Hei	Lauseet	Tämä aihe käynnistyy, kun käyttäjä sanoo näkemii...			JS Jukka-Pekka Sirviö kaksi kuukautta sitten
ICT Laitteiden käyttö ulkomailla	Lauseet				JS Jukka-Pekka Sirviö 19 päivää sitten
ICT Miten asennan sovelluksen puhelimeen	Lauseet				JS Jukka-Pekka Sirviö kuukausi sitten
ICT Miten asennan sovelluksen tietokoneeseen	Lauseet				JS Jukka-Pekka Sirviö kuukausi sitten
ICT Miten järjestän eTuve-kokouksen	Lauseet				JS Jukka-Pekka Sirviö kuukausi sitten
ICT Miten osallistun eTuve-kokoukseen?	Lauseet				JS Jukka-Pekka Sirviö kuukausi sitten
ICT Miten päivitän sovelluksen puhelimeen	Lauseet				JS Jukka-Pekka Sirviö kuukausi sitten
ICT Miten saan uuden puhelimen	Lauseet				JS Jukka-Pekka Sirviö kuukausi sitten
ICT Miten teen, kun puhelin on varastettu	Lauseet				JS Jukka-Pekka Sirviö kuukausi sitten

Kuva 8. Aiheet ja aihealueet Copilot Studiassa

Aiheet syötetään kulkuina, joka sisältää käynnistyssanoja (triggereitä), erimuotoisia viestejä ja kysymyksiä. Viestit voi sisältää tekstiä ja kuvia muutamalla eri tavalla esitettynä. Kysymykset tallentavat ehdotietoja botin käyttöön ja suuntaa kulun vastauksen mukaiseen polkuun. Esimerkiksi, jos käyttäjä kysyy “miten puhelimen sovellus päivitetään”, on ensimmäinen viesti kysymys, jolla selvitetään, onko käyttäjällä käytössä Android- vai iOS-laite. Vastauksen perusteella viestintä jatkuu omaa polkuaan pitkin niin, että käyttäjä saa ohjeita juuri oman laitteensa mukaisesti.

Aiheiden kulut voivat tallentaa kysymyksistä ja käyttäjän syötteistä tietoa, mutta tallennukset tapahtuvat aihekohtaisiin muuttujiin. Aiheissa voisi hyödyntää entiteettejä, jotka tallentavat käyttäjistä tietoja botin hyödynnettäväksi globaalilla tasolla. Tätä ominaisuutta käyttäen käyttäjä voisi käydä laajemman keskustelun esimerkiksi omasta puhelimestaan. Häneltä kysytään ensimmäisen aiheen läpikäynnissä mikä hänen laitteensa on, mutta seuraavissa aiheissa, botti muistaisi hänen vastauksensa. Entiteettien käyttö on hieman tavallisia kulkuja vaativampaa, mutta helpottaa kehitystyötä ja käyttöä pitkällä tähtäimellä. (Rivera 2023, luku 7.)

Microsoft on laatinut parhaita käytäntöjä keskustelubottien luomista varten, mutta ne käsittelevät pääasiassa suunnitteluvaihetta (Microsoft 2024p). Toteutuksen parhaita käytäntöjä ja suosituksia käsitellään eri osissa Copilot Studio –dokumentaatiota, mutta kokemattoman toteuttajan tueksi niitä voisi olla tarjolla enemmän. Aiheiden syöttämiselle ei esitetä dokumentaatioissa lainkaan parhaita käytäntöjä, mikä on ongelma aihepohjaiselle bottikehitykselle. Projektin aikana mietintää aiheutti

erityisesti se, miten toistuvat aiheet kannattaa toteuttaa. Esiin nousseita kysymyksiä olivat muun muassa se, että mihin kannattaa vetää raja yhteisten aiheiden tekemiseen. Aiheiden lukumäärä on rajoitettu, mutta kuitenkin melko korkea. Kuitenkin on varauduttava siihen, ettei aiheita synny liikaa, mikä voi aiheuttaa ongelmia, jos yhteiskäyttöisiä komponentteja on paljon.

3.4.3 Toteutuskaari dokumentoidusta kulusta aiheeksi

Kuvissa 9, 10 ja 11 kuvataan esimerkkinä, miten dokumentoidusta määrittelystä muodostuu botin antama vastaus. Kuvista on nähtävissä, miten yksittäinen kulku on dokumentoitu, miten sisältö on dokumentaation perusteella syötetty Copilot Studioon ja miten botti vastaa sisällön perusteella kysymykseen. Kuvassa 12 näytetään työkalun generoima koodimuoto, johon sisältö tallentuu.

Kuvassa 9 on nähtävissä Word-dokumenttiin kirjattu suunnitelma kululle. Otsikkona on tuleva aiheen nimi. Otsikon alle on trigger-merkinnällä merkitty sanat ja lauseet, jotka toimivat aiheen herätteinä. Seuraavan viivan kohdalla on viesti, jonka aihe lähettää käyttäjälle. Jos viestejä olisi useita, olisi dokumentaatioissa useita viivoja. Dokumentaatioon merkitään myös ehtolauseet, joita esimerkki kulku ei kuitenkaan sisällä.

Miten asennan päivityksen Software Centeristä?

- Trigger: päivitys, Software Center, miten asennan päivityksen, miten päivitän sovelluksen, sovelluksen päivitys
- 1. Avaa Software Center.
2. Mene vasemman reunan osioon Asennuksen tila.
3. Riveillä näkyy asennetut sovellukset. Jos sovellus on päivitettävissä, kohdassa Tila lukee "Päivitys saatavilla".
4. Klikkaa riviä avataksesi sovelluksen näkymän.
5. Klikkaa Asenna.

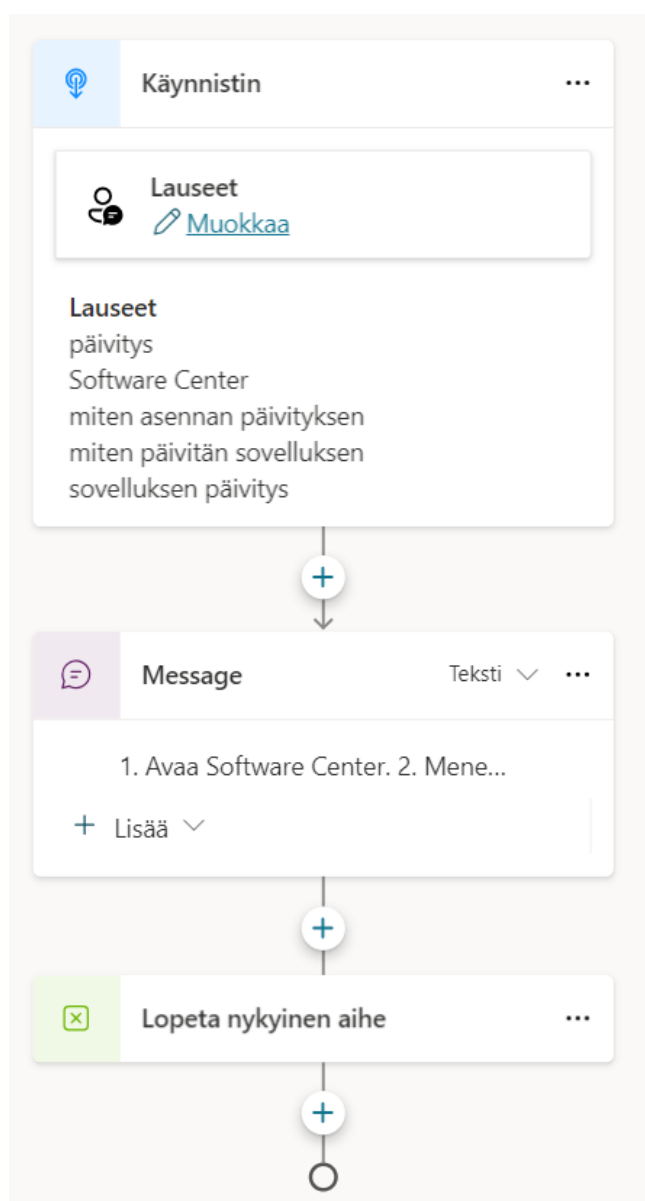
Kuva 9. Dokumentoitu yksittäisen kulun määrittely

Kuvassa 10 on edellä kuvatun dokumentaation tiedot vietyinä Copilot Studio canvas-työkaluun, jolla aiheet luodaan. Käynnistin-solmuun on syötetty avainsanat ja Message-solmuun viestin sisältö. Aihe päätetään Lopeta nykyinen aihe -solmulla.

Jos bottia haluaa tehdä elävämmän kaltaiseksi, voisi eri viesteihin syöttää useita erilaisia vastauksia. Botti voisi silloin hieman varioida vastauksiaan, vaikka vastausten generointi ei olisikaan käytössä. Tämä kasvattaisi botin rakentamisen työläyttä, joten projektissa vaihtoehtoisia viestejä ei juurikaan käytetty.

Kuvassa näkyviin +-kuvakkeisiin olisi mahdollista lisätä toisia solmuja, jotka voivat olla yksittäisiä viestejä tai laajempia kulkuja, kuten ehtolauseita. Ehtolauseet näytettäisiin canvas-työkalussa rinnakkain rivissä.

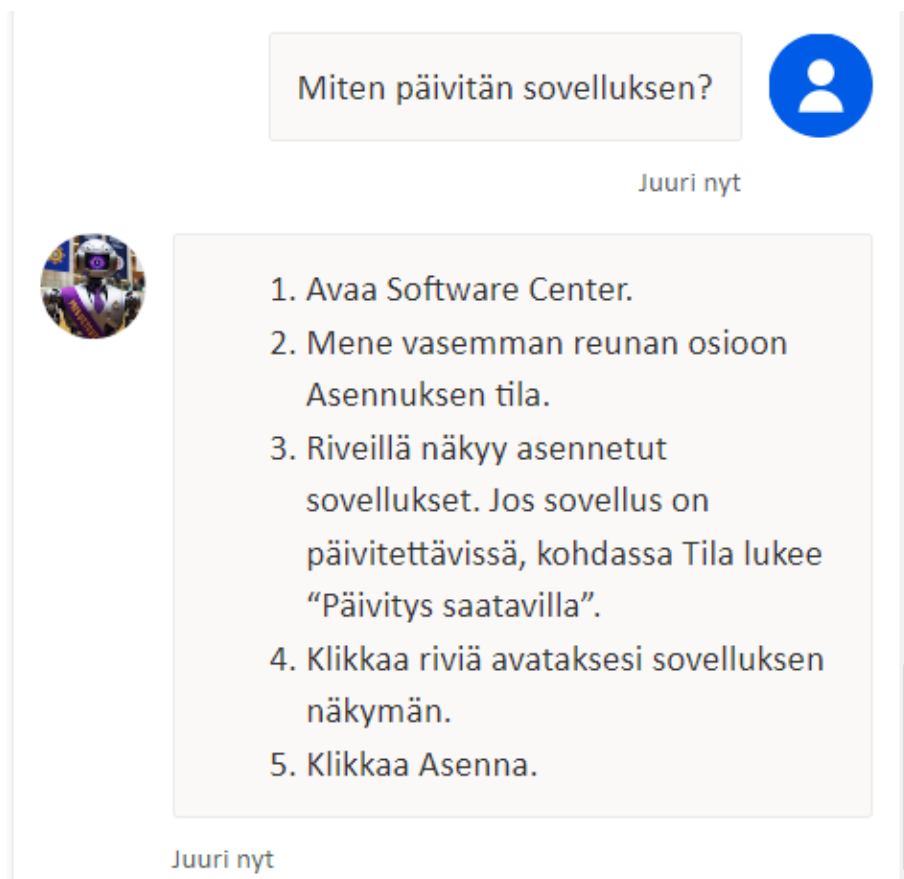
Aiheen lopettaminen ei olisi välttämätöntä, mutta se voi vähentää botin virheitä sillä, että se ei ala sekoittamaan eri asioihin liittyviä kyselyitä. Ongelma lopettamisessa on se, että näin toimiminen rajoittaa botin kykyä jatkaa saman aiheen käsittelyä. Projektissa on kuitenkin oletettu, että botille esitettävät kysymykset ovat yksinkertaisia, joten laajemmalle keskustelulle ei olisi tarvetta. Keskustelun jatkamisen mahdollistaminen olisi myös tärkeämpää, jos generatiivinen tekoäly olisi käytössä, mutta projektin botissa sen käyttö on hyvin rajoittunutta.



Kuva 10. Copilot Studioon viety sisältö

Kuvassa 11 on nähtävillä edellä kuvattu aihe niin kuin se esiintyy käyttäjälle eli miltä aihe näyttää Copilot Studion keskustelunäkymässä ja miltä se näyttäisi julkaistuna jollekin sivulle oletusulkosivustaan.

Kuvan tilanteessa botti on saanut käyttäjältä kysymyksen, joka sisältää esimerkkiaiheen aloitussanat ja botti on osannut vastata kysymykseen dokumentaation mukaisesti aiheeseen syötetyllä viestillä. Vastaus ei ole generoitu, vaan aiheeseen syötetyn tekstin mukainen.



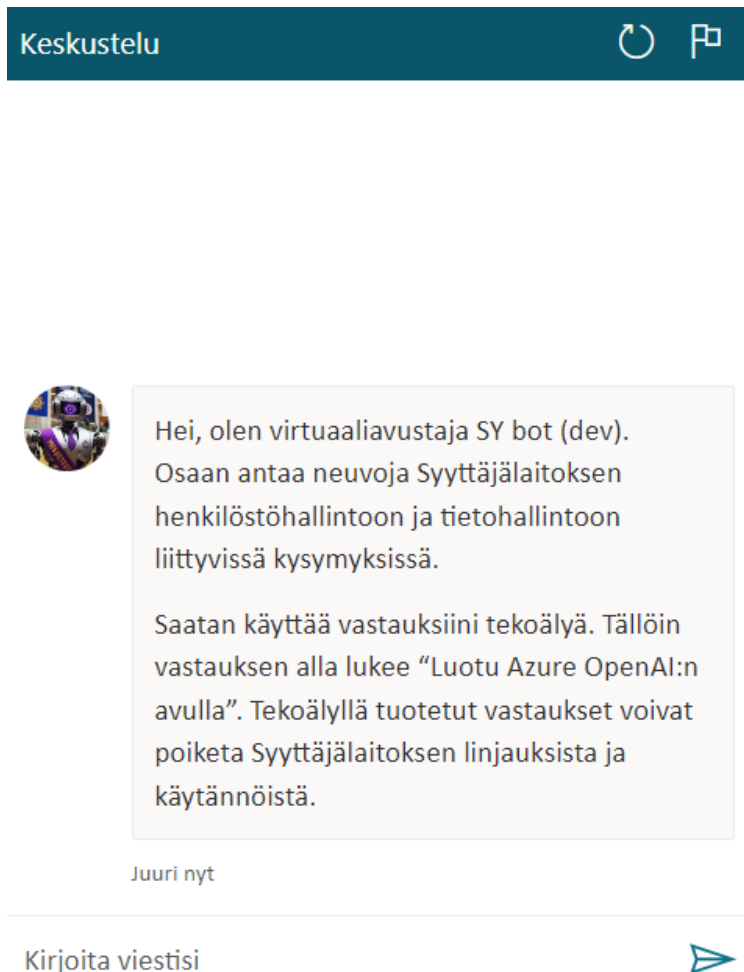
Kuva 11. Keskustelubotti käsittelee syötettyä kysymystä vastaavan kysymyksen

Kuvassa 12 on nähtävillä esimerkkiaiheen koodieditori, josta näkee myös aiheen kulun sisällön. Koodi on muokattavissa ja kopioitavissa, mikä helpottaa aiheiden ylläpitoa ja luomista. Kuvasta on piilotettu id-tiedot tietoturvasyistä.

```
Aiheet > ICT Sovelluksen päivittäminen Software Centeristä  av
1 kind: AdaptiveDialog
2 beginDialog:
3   kind: OnRecognizedIntent
4   id: main
5   intent:
6     triggerQueries:
7       - päivitys
8       - Software Center
9       - miten asennan päivityksen
10      - miten päivitän sovelluksen
11      - sovelluksen päivitys
12
13  actions:
14    - kind: SendActivity
15      id: sendActivity_██████
16      activity: |-
17        1. Avaa Software Center.
18        2. Mene vasemman reunan osioon Asennuksen tila.
19        3. Riveillä näkyvät asennetut sovellukset. Jos sovellus on päivitettävissä, kohdassa Tila lukee "Päivitys saatavilla".
20        4. Klikkaa riviä avataksesi sovelluksen näkymän.
21        5. Klikkaa Asenna.
22
23    - kind: EndDialog
24      id: ██████
```

Kuva 12. Copilot Studion generoima koodi kulusta.

3.5 Keskustelubotin testaus ja laadun arviointi



Kuva 13. Botin aloitusnäky

Kuva 13 esittelee botin aloitusviestin, jolla käyttäjää tervehditään. Botin viestintä on melko formaalia, mikä on toimeksiantajan sisäisen viestinnän ja dokumentaation mukaisen persoonan mukaista.

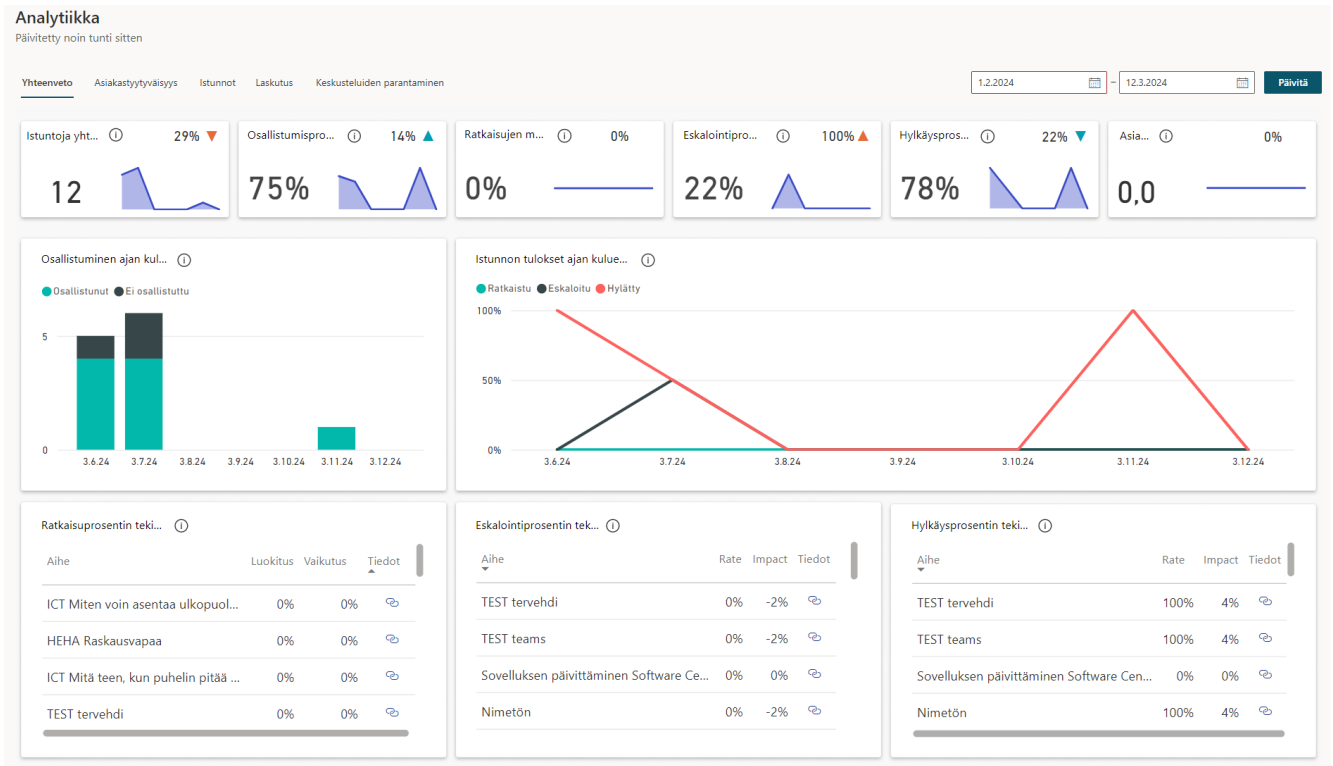
Keskustelubotin laatua mitataan yleensä käyttäjäkäytön avulla. Työkalu tarjoaa valmiit keinot tämän analytiikkadatan seurantaan, mutta koska työn rajauksen mukaisesti toteutusta ei julkaista, ei käyttäjien käytöstä pystytä keräämään. Laadun tarkastelu rajoittuu siis määrittelyyn ja testaukseen osallistuvien asiantuntijoiden palautteeseen, joka on suppeampaa kuin todellinen käyttäjäpalaute. Kehitysprosessiin osallistuvien asiantuntijoiden palaute voi olla myös positiivisempaa todelliseen tilanteeseen nähden. Toimeksiantajan henkilöstön suhtautuminen tietoteknisiin uudistuksiin on usein ollut kriittistä, joten saadun palautteen uskottavuus on kyseenalaistettavissa.

Copilot Studioissa bottia voi testata jatkuvasti erillisessä testauspaneelinäkymässä, jota kehittäjä voi pitää avattuna tai suljettuna työkalua käyttäessään. Paneelista botille voi antaa syötteitä samalla tavalla kuin jos botti olisi julkaistu sivulle. Paneelista pääsee myös suoraan muokkaamaan keskustelussa käynnissä olevaa aihetta. Keskustelun voi myös nopeasti nollata ja aloittaa alusta. Projektin aikana bottia testattiin pääasiallisesti tämän paneelin kautta. (Rivera 2023, luku 7)

Tekoälypohjaisten keskustelubottien yleisimmät mittarit ovat skaalautuvuus, ihmismäisyys, yhteistoimivuus (interoperability) eli kyky uudelleen käyttää aiemmin syötettyjä tietoja ja nopeus (Nirala, Singh 2022, luku 8). Projektissa tuotettu botti vastaa melko huonosti kaikkiin näihin mittareihin. Ihmismäisyyttä mitataan esimerkiksi Turingin testillä, jossa kone laitetaan vastaamaan kysymyksiin rinnakkain ihmisen kanssa. Projektissa botille ei tehty tätä testiä, mutta on selvää, että botin ihmismäisyys on melko heikkoa. Botti vastaa kysymyksiin aina samalla tavalla, koska vastauksia ei generoida tekoälyn avulla. Samoin skaalautuvuus ja yhteistoimivuus ovat heikkoja, koska botti on luotu hyvin selkeää omaa toimintaympäristöään varten. Toisaalta projektin tavoitteena ei ollutkaan tuottaa erityisen ihmismäistä keskustelumahdollisuutta vaan tavoite oli tuottaa selkeitä vastauksia antava työväline.

Botin laajempi testaus jäi projektissa asetettujen rajoitteiden vuoksi tekemättä. Koska bottia ei julkaistu, ei sitä saatu myöskään laajalle käyttäjäryhmälle testattavaksi. Testausta tehtiin pääasiassa kehittäjän toimesta ja aiheiden toimivuuden varmistamiseksi.

Botin toimivuus on siksi varmistettu vain kehittäjän oman mielikuvituksen rajoissa. Todelliset käyttäjät voivat keksiä kysyä botilta asioita, jotka kehittäjälle ei ole tullut mieleen. Botille esitettäviä kysymyksiä ja sen kanssa käytyjä keskusteluita on kuitenkin mahdollista tarkastella Copilot Studioon integroidussa analytiikkatyökalussa. Työkalu tarjoaa muutaman erilaisen raportin, joista voi tarkastella esiasetettuja mittareita. Kuvassa 14 on nähtävillä Yhteenveto-raportti, josta näkee muun muassa toteutuneiden istuntojen määriä ja niiden lopputuloksia. Työkalusta on lisäksi mahdollista viedä keskusteluita tarkasteltavaksi muihin työkaluihin csv-muotoisena. (Microsoft 2024q).



Kuva 14. Analyysityökalun Yhteenveto-raportti

Projektin aikana analyysityökaluun kerääntyi aineistoa vain botin toteutuksen aikaisesta testauksesta, missä istuntomäärät olivat loppujen lopuksi aika vähäiset. Tämän vuoksi analyysityökalun hyödyt ja käytännöt jäivät epäselviksi. Botin julkaisun ja käyttöönoton myötä istuntomäärät kasvavat ja raporttien antamien mittaustuloksien merkitys aukeaa paremmin.

4 Pohdinta

Lopputuloksen laatu on ristiriitainen. Botin rakentamisessa ei käytetty kaikkein edistyneimpiä ominaisuuksia, joita työkaluissa oli tarjolla. Laajan aineiston syöttäminen aiheina aiheuttaa melko raskaan ylläpitotarpeen ja botin sisältö voi jäädä jopa sen kehittäjille epäselväksi. Toisaalta taas botti tekee luotettavasti sen, mitä sen odotetaan tekevän. Testattaessa erilaisilla kysymyksillä botti yleensä vastasi kuten sen haluttiin vastaavan. Laatuksen arviointi on tietenkin subjektiivista, koska bottia ei ole julkaistu, eikä otettu vielä käyttöön. Toisaalta alusta tarjoaa melko hyvät työkalut kysymysten seurantaan, joten tehostetulla ylläpidolla ja seurannalla käyttöönottovaiheessa, on mahdollista mitigoida luotettavuusongelmia.

Kehittyneemmistä ominaisuuksista erityisesti generatiivisen tekoälyn arviointiin käytettiin toteutuksen yhteydessä melko paljon aikaa. Lopputuloksena oli kuitenkin todettava, että tekoälypohjainen ratkaisu ei toiminut sillä tasolla, että siihen voisi tässä tapauksessa luottaa. Mahdollisia syitä tähän olivat tekoälylle syötetyn aineiston huono laatu sekä se, että näitä työkalun ominaisuuksia ei osattu käyttää riittävässä laajuudessa. Luotettavuusarvioinnin perusteella tekoälyn käyttö päätettiin jättää toteuttamatta. Generatiivisen tekoälyn testaamista kuitenkin jatketaan ja jatkokehityksen tavoitteena on siirtyminen pienin askelin käyttämään tätä ominaisuutta.

Raportin kirjoittamishetkellä bottiin on viety aiheiksi tärkeimmät ja yleisimmät kysymykset, joita tietohallinto ja henkilöstöhallinto saavat käyttäjiltä, mutta botti ei osaa vielä vastata aivan kaikkeen. Botti olisi periaatteessa julkaistavissa, mutta käyttäjille pitäisi viestiä sen keskeneräisyydestä. Iteratiivinen julkaisu- ja testaus sykli voisi johtaa kehitystyön tehostumiseen, koska on hankalaa ennalta tunnistaa kysymyksiä ja sanamuotoja, joita käyttäjät tulevat käyttämään botille kommunikoidessa. Kehittämiseen osallistuvat asiantuntijat ovat itse niin syvällä omien aiheiden osaamisessaan, että kysymysten käsittely tavallisen käyttäjän näkökulmasta on vaikeaa.

Copilot Studio työkaluna ja sen rinnalla kulkeva Power Platform jättivät projektin aikana melko positiivisen vaikutelman. Työkalujen kyky integroitua keskenään ja hyödyntää muita Microsoftin tuotteita kuvastaa korkeaa potentiaalia siitä, miten niillä työskentelyä voisi kehittää ja tehostaa.

4.1 Jatkokehityssuunnitelma

Toteutettu keskustelubotti täyttäisi paikkansa ohjaavana välineenä riittävän hyvin raportin kirjoitushetkellä. Tarkempi analyysi on tehtävä konkreettisen käyttöönoton myötä, mutta tällä hetkellä näyttää siltä, että ratkaisu tulee tukemaan hyvin käyttäjien tiedonhakutarpeita. Botti antaa selkeitä vastauksia sille esitettäviin kysymyksiin.

Tiedonhakutarpeiden näkökulmasta selkeä puute on se, että botti käsittelee vain henkilöstö- ja tietohallinnon toimialoja. Muiden toimialojen tietojen tuominen bottiin on selkeä jatkokehitysaskel, joka käyttöönoton jälkeen on tarpeellista tehdä. Taloushallinnon, matkahallinnon ja viestinnän ohjeistuksien tuominen bottiin helpottaisi käyttäjiä löytämään vastauksia kysymyksiinsä. Bottia esiteltiin toteutuksen aikana muillekin toimialoilla ja se herätti hyvin positiivista vastakaikua. Myös kiinnostusta ja innostustakin kansalaiskehittämiseen oli havaittavissa, mikä kertoo työkalun helppokäyttöisyydestä.

Aiheiden lukumäärä Copilot Studioissa on rajattu 1000 kappaleeseen eli yksittäinen aiheperustaisesti toteutettu botti ei voi käsitellä loputonta määrää eri kysymyksiä. Käytännössä tämän aiheiden määrän pitäisi kuitenkin riittää useamman toimialueen tietojen käsittelyyn botissa. Yhdellä aiheella voi olla 200 eri käynnistyslausetta, minkä senkin pitäisi olla riittävä määrä botin järkevään käyttöön. Käytännössä käynnistimiä voisi olla tarvittaessa enemmän, syöttämällä limiitin ylittävät käynnistyslauseet uuteen erilliseen aiheeseen, joka ei tekisi mitään muuta kuin kutsuisi alkuperäistä aihetta. Tämä tietenkin vähentää 1000 aiheen limiittiä. (Microsoft 2024r.)

Korjaus tähän jossain määrin teoreettiseen ongelmaan olisi generatiivisen tekoälyn laajempi käyttö. Tähänkin kuitenkin liittyy ongelmia, erityisesti tekoälyn kykyyn käsitellä suomen kieltä. Toinen ongelma on se, että tekoälyn opetusaineistoa ei välttämättä ole tarpeeksi saatavilla sellaisessa muodossa, että Copilot-tekoäly osaisi sitä käsitellä. Tähän mahdollisesti voi olla ratkaisuna siirtymisen tiedostopohjaisesta asiakirjojen ja tiedon hallinnasta websivupohjaiseen.

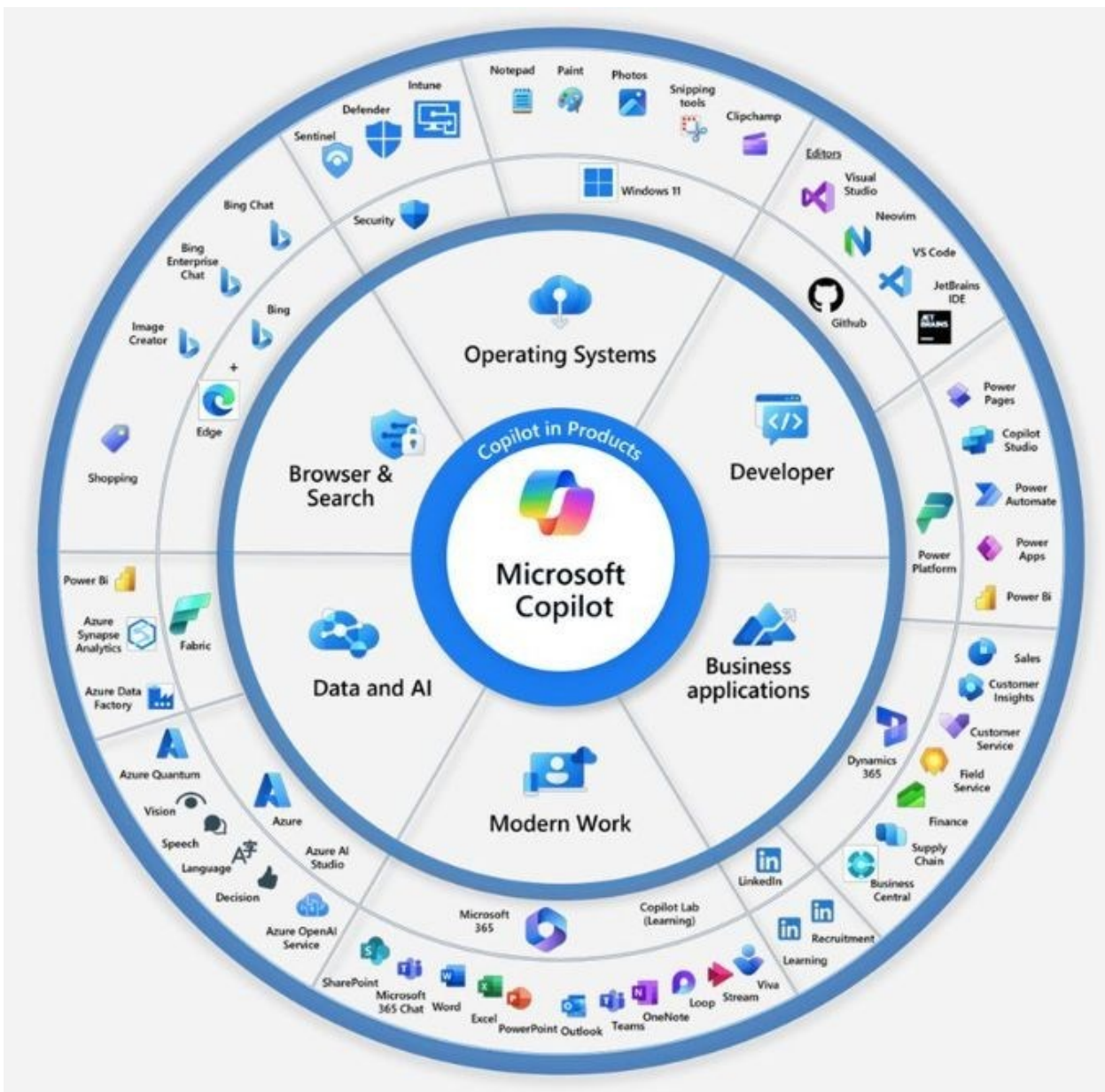
Projektissa ei tehty botin julkaisua ja käyttöönottoa, joten seuraavien askeleiden tulisi kohdistua tähän tavoitteeseen. Ensimmäinen asia on tunnistaa julkaisukohde. Toimeksiantajan tämänhetkessä infrastruktuurissa selkein julkaisukohde olisi Teams, mutta tulevaisuudessa on tulossa tarjolle myös muita vaihtoehtoja.

Ennen käyttöönottoa uudelle järjestelmälle on tehtävä tietosuojaan vaikutusten arviointi. Toimeksiantajan tietosuoja-aineistoon pitää myös lisätä viitteet uuteen järjestelmään, jotta rekisteröidyillä on saatavilla tarvittavaa tietoa oikeuksiensa toteutumisesta. Järjestelmän käyttöönotto on myös käsiteltävä henkilöstökokouksessa.

Ennen käyttöönottoa on suunniteltava botin ylläpidosta eli sovittava, miten botin käyttöä seurataan ja kenen vastuulla on minkäkin toimialueen kysymysten syöttäminen tai korjaaminen järjestelmässä. Botin toimintaan liittyvät omistajuudet ja vastuut on myös sovittava.

Copilot Studio sisältää useita kehittyneitä ominaisuuksia, joita tässä työssä ei käsitelty. Näiden hyödyntäminen paremman käyttökokemuksen saavuttamiseksi on myös potentiaalinen jatkoaskel

keskustelubottiprojektissa. Microsoftin tuotekatalogissa on myös monia potentiaalisia ominaisuuksia, joilla botin toimintaa voisi kehittää paremmaksi. Esimerkiksi Power Automatella voisi botille esitetystä kysymyksestä koostaa valmiita palvelupyynnöksiä lähetettäväksi toimialoille tai lähettää viestejä Teams-kanaviin. Tässä raportissa on pääasiassa käsitelty Power Platform –tuotetta, mutta Microsoftilla on tavoitteena eri tuoteperheiden yhä vahvempi integroituminen, jolloin Copilot Studio voisi hyödyntää yhä laajemmassa määrässä Microsoftin tuotteita. Kuvassa 15 esitellään tuotekatalogia, johon Microsoft on integroimassa Copilot-tekoälyään ja joihin sen myötä Copilot Studiolla voisi olla pääsy tuotteissa olevien valmiiden integraatioiden välityksellä. (Urrutia 2024.)



Kuva 15. Microsoftin tuotteet, jotka hyödyntävät Copilot-tekoälyä

4.2 Tekijän oma oppiminen

Power Platform ja Copilot Studio olivat tekijälle vain kevyesti tuttuja ennen opinnäytetyön aloittamista. Työn aikana pääsi tutustumaan melko laajasti Copilot Studion ominaisuuksiin ja dokumentaatioon. Työkalun helppokäyttöisyyden myötä sen perusominaisuuksien käyttö ei edellyttänyt merkittävää oppimisen tarvetta. Pääosa opinnäytteen tuotteesta oli toteutettavissa näillä perusominaisuuksilla.

Työkalu tarjoaa myös edistyneempiä toimintoja. Merkittävin näistä on generatiivinen tekoäly, mutta sen lisäksi muun muassa mahdollisuus integroitua muuhun Power Platform –tuoteperheen ratkaisuihin olisivat ominaisuuksia, joista voisi olla hyötyä. Näiden käyttö ei onnistu täysin intuitiivisesti ja vaativat dokumentaation tarkkaa sisäistämistä. Dokumentaation taso vaihtelee, joten osa ominaisuuksista edellyttää myös muun ohjeistuksen etsimistä. Vaikka toteutettu tuote ei näitä ominaisuuksia käytä, sai tekijä tutustuttua niihin syvemmin työn ohessa. Tämä mahdollistaa tuotteen kehittämisen tulevaisuudessa tehokkaammaksi ja käytettävämmäksi.

Helppokäyttöisyys tekee Copilot Studiosta soveltuvan kansalaiskehittämiseen. Yksinkertaisia ja joitain monimutkaisempia kulkuja on mahdollista syöttää työkaluun helposti. Toteutuksen yhteydessä joitain toimialojen asiantuntijoita opastettiin työkalun käyttämiseen, mikä mahdollistaa työkalun käytön laajentamisen ja ylläpidon tehostumisen sitten, kun toteutettu botti aikanaan otetaan käyttöön. Monipuolisemmat toiminnot tulevat kuitenkin edellyttämään kansalaiskehittäjien tukemista, mikä tulee edellyttämään oman osaamisen jatkuvaa kehittämistä. Opinnäytteen aikana tekijä tunnistikin työkalusta useita osa-alueita, joihin on tärkeää jatkaa tutustumista, jotta niitä voisi käyttää jatkossa tehokkaammin.

Copilot Studion opiskeluun on tarjolla laajasti aineistoa. Työkalun dokumentaatio on hyvin laaja ja internetin keskusteluyhteisöistä, kuten Reddit ja StackOverFlow, löytyy aiheeseen liittyvää sisältöä. Copilot Studiolla on perustettuna Microsoftin keskusteluyhteisö, josta voi etsiä ohjeita ja kysyä ratkaisuja ongelmiin (Microsoft 2024s). Yhteisössä on tällä hetkellä tuhansia viestejä, joten voi olettaa, että sieltä löytyy vastauksia omiin ongelmatilanteisiin. Lisäksi Microsoft tarjoaa opiskelupaketteja, joissa syvennyttään työkalujen käyttämiseen ja joiden pohjalta voi hakeutua sertifioiduksi toteuttajaksi.

Lähteet

Adobe s.a. Mitä generatiivinen tekoäly on ja miten se toimii? Luettavissa:

<https://www.adobe.com/fi/products/firefly/discover/how-generative-ai-work.html>. Luettu: 9.4.2024.

Goasduff L. 31.7.2019. Chatbots Will Appeal To Modern Workers. Gartner. Luettavissa:

<https://www.gartner.com/smarterwithgartner/chatbots-will-appeal-to-modern-workers>. Luettu 30.4.2024.

JUHTA - Julkisen hallinnon tietohallinnon neuvottelukunta. 15.5.2018. JHS 173 ICT-palvelujen kehittäminen: Vaatimusmäärittely. Luettavissa: <https://www.avoindata.fi/data/fi/dataset/jhs-173-ict-palvelujen-kehittaminen-vaatimusmaarittely>. Luettu 17.2.2024.

Microsoft s.a. Microsoft Power Platform. Luettavissa: <https://powerplatform.microsoft.com/fi-fi/what-is-power-platform>. Luettu: 9.4.2024.

Microsoft 15.11.2023. Introducing Microsoft Copilot Studio and new features in Copilot for Microsoft 365. Luettavissa: <https://www.microsoft.com/en-us/microsoft-365/blog/2023/11/15/introducing-microsoft-copilot-studio-and-new-features-in-copilot-for-microsoft-365/>. Luettu: 30.4.2024.

Microsoft 2024a. Microsoft Copilot Studioin yleiskatsaus. Luettavissa: <https://learn.microsoft.com/fi-fi/microsoft-copilot-studio/fundamentals-what-is-copilot-studio>. Luettu: 6.4.2024.

Microsoft 2024b. Microsoft Copilot Studioin yhteensopivuustarjoomat. Luettavissa: <https://learn.microsoft.com/fi-fi/microsoft-copilot-studio/admin-certification>. Luettu: 6.4.2024.

Microsoft 2024c. Copilot Studion suojaus ja hallinto. Luettavissa: <https://learn.microsoft.com/fi-fi/microsoft-copilot-studio/security-and-governance>. Luettu: 6.4.2024.

Microsoft 2024d. Aiheiden hallinta Microsoft Copilot Studioissa. Luettavissa: <https://learn.microsoft.com/fi-fi/microsoft-copilot-studio/authoring-topic-management>. Luettu: 14.4.2024.

Microsoft 2024e. Avustajan laajennusten luominen ja määrittäminen (esiversio). Luettavissa: <https://learn.microsoft.com/fi-fi/microsoft-copilot-studio/copilot-plugins-overview>. Luettu: 14.4.2024.

Microsoft 2024f. Käyttövalmiiden avustajamallien käyttäminen Microsoft Teamsissa. Luettavissa: <https://learn.microsoft.com/fi-fi/microsoft-copilot-studio/fundamentals-templates>. Luettu: 6.4.2024.

Microsoft 2024g. Avustajien vieminen ja tuominen ratkaisujen avulla. Luettavissa: <https://learn.microsoft.com/fi-fi/microsoft-copilot-studio/authoring-export-import-bots>. Luettu 14.4.2024.

Microsoft 2024h. Tuetut kielet. Luettavissa: <https://learn.microsoft.com/fi-fi/microsoft-copilot-studio/authoring-language-support>. Luettu: 6.4.2024.

Microsoft 2024i. Monikielisten avustajien määrittäminen ja luominen. Luettavissa: <https://learn.microsoft.com/fi-fi/microsoft-copilot-studio/multilingual>. Luettu: 6.4.2024.

Microsoft 2024j. Ladattujen asiakirjojen käyttö generatiivisia vastauksia varten (esiversio). Luettavissa: <https://learn.microsoft.com/fi-fi/microsoft-copilot-studio/nlu-documents>. Luettu: 6.4.2024.

Microsoft 2024k. Generatiiviset vastaukset. Luettu: <https://learn.microsoft.com/fi-fi/microsoft-copilot-studio/nlu-boost-conversations>. Luettu: 6.4.2024.

Microsoft 2024l. Entiteettien ja paikkojen täyttämisen käyttäminen Microsoft Copilot Studion avustajissa. Luettu: <https://learn.microsoft.com/fi-fi/microsoft-copilot-studio/advanced-entities-slot-filling>. Luettu: 14.4.2024.

Microsoft 2024m. Käytä muuttujia. Luettavissa: <https://learn.microsoft.com/fi-fi/microsoft-copilot-studio/authoring-variables>. Luettu 14.4.2024.

Microsoft 2024n. Järjestelmäaiheiden käyttäminen Microsoft Copilot Studioissa. Luettavissa: <https://learn.microsoft.com/fi-fi/microsoft-copilot-studio/authoring-system-topics>. Luettu: 6.4.2024.

Microsoft 2024o. Aiheiden luominen ja mukauttaminen. Luettavissa: <https://learn.microsoft.com/fi-fi/microsoft-copilot-studio/authoring-create-edit-topics>. Luettu: 14.4.2024.

Microsoft 2024p. Kehotteen muokkaamisen avulla voit antaa mukautettuja ohjeita avustajallesi. Luettavissa: <https://learn.microsoft.com/fi-fi/microsoft-copilot-studio/nlu-generative-answers-prompt-modification>. Luettu 6.4.2024.

Microsoft 2024q. Pika-aloitus: Microsoft Copilot Studio -avustajan luominen ja käyttöönotto. Luettavissa: <https://learn.microsoft.com/fi-fi/microsoft-copilot-studio/fundamentals-get-started>. Luettu: 6.4.2024.

Microsoft 2024r. Microsoft Copilot Studioin kiintiöt, rajat, sovelluksen rekisteröinti, varmenteet ja määritysarvot. Luettavissa: <https://learn.microsoft.com/fi-fi/microsoft-copilot-studio/requirements-quotas>. Luettu: 6.4.2024.

Microsoft 2024s. Microsoft Copilot Studio Community. Luettavissa: <https://powerusers.microsoft.com/t5/Copilot-Studio-Community/ct-p/PVACommunity>. Luettu: 6.4.2024.

Muchmore M. 2024. What Is Copilot? Microsoft's AI Assistant Explained. PCMag. Luettavissa: <https://www.pcmag.com/explainers/what-is-microsoft-copilot>. Luettu: 6.4.2024.

Nakanishi D. s.a. What is no-code?. NoCode.Tech. Luettavissa: <https://www.nocode.tech/article/what-is-no-code>. Luettu: 6.4.2024.

Nirala K., Singh N. 3.1.2022. A survey on providing customer and public administration based services using AI: chatbot. Springer. Luettavissa: <https://link.springer.com/article/10.1007/s11042-021-11458-y>. Luettu: 5.4.2024.

Rivera Jason. 2023. Building Solutions with the Microsoft Power Platform. O'Reilly Media. E-kirja. Luettu: 30.4.2024.

SAP s.a. Mikä on 'low-code-/no-code' sovelluskehitys? Luettavissa: <https://www.sap.com/finland/products/technology-platform/low-code/what-is-low-code-no-code.html>. Luettu 11.2.2024.

Syyttäjälaitos s.a. Organisaatio. Luettavissa: <https://syyttajalaitos.fi/syyttajalaitos>. Luettu: 13.5.2024.

Tietosuojavaltuutetun toimisto s.a. Henkilötietojen käsittely. Luettavissa: <https://tietosuoja.fi/henkilotietojen-kasittely>. Luettu: 18.2.2024.

Urrutia A. 22.1.2024. Microsoft Copilot Studio: Make your own Copilot with no code. Microsoft Reactor. Video. Katsottavissa: <https://www.youtube.com/watch?v=sJUJbYEqSo>. Katsottu: 6.4.2024.

Wieggers K. & Beatty J. 2013. Software Requirements, 3rd Edition. Microsoft Press. E-kirja. Luettu: 30.4.2024.