



Faculteit Bedrijf en Organisatie

Ontwikkeling van de Proof of Concept voor een nieuw IT asset management programma voor AZ Glorieux
gebouwd met Power Apps

Thomas Van den Hauwe

Scriptie voorgedragen tot het bekomen van de graad van
professionele bachelor in de toegepaste informatica

Promotor:
Benjamin Vertonghen
Co-promotor:
Siemen Vanneste

Instelling: —

Academiejaar: 2019-2020

Tweede examenperiode

Faculteit Bedrijf en Organisatie

Ontwikkeling van de Proof of Concept voor een nieuw IT asset management programma voor AZ Glorieux
gebouwd met Power Apps

Thomas Van den Hauwe

Scriptie voorgedragen tot het bekomen van de graad van
professionele bachelor in de toegepaste informatica

Promotor:
Benjamin Vertonghen
Co-promotor:
Siemen Vanneste

Instelling: —

Academiejaar: 2019-2020

Tweede examenperiode

Woord vooraf

Samenvatting

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris. Nam arcu libero, nonummy eget, consectetur id, vulputate a, magna. Donec vehicula augue eu neque. Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas. Mauris ut leo. Cras viverra metus rhoncus sem. Nulla et lectus vestibulum urna fringilla ultrices. Phasellus eu tellus sit amet tortor gravida placerat. Integer sapien est, iaculis in, pretium quis, viverra ac, nunc. Praesent eget sem vel leo ultrices bibendum. Aenean faucibus. Morbi dolor nulla, malesuada eu, pulvinar at, mollis ac, nulla. Curabitur auctor semper nulla. Donec varius orci eget risus. Duis nibh mi, congue eu, accumsan eleifend, sagittis quis, diam. Duis eget orci sit amet orci dignissim rutrum.

Nam dui ligula, fringilla a, euismod sodales, sollicitudin vel, wisi. Morbi auctor lorem non justo. Nam lacus libero, pretium at, lobortis vitae, ultricies et, tellus. Donec aliquet, tortor sed accumsan bibendum, erat ligula aliquet magna, vitae ornare odio metus a mi. Morbi ac orci et nisl hendrerit mollis. Suspendisse ut massa. Cras nec ante. Pellentesque a nulla. Cum sociis natoque penatibus et magnis dis parturient montes, nascetur ridiculus mus. Aliquam tincidunt urna. Nulla ullamcorper vestibulum turpis. Pellentesque cursus luctus mauris.

Nulla malesuada porttitor diam. Donec felis erat, congue non, volutpat at, tincidunt tristique, libero. Vivamus viverra fermentum felis. Donec nonummy pellentesque ante. Phasellus adipiscing semper elit. Proin fermentum massa ac quam. Sed diam turpis, molestie vitae, placerat a, molestie nec, leo. Maecenas lacinia. Nam ipsum ligula, eleifend at, accumsan nec, suscipit a, ipsum. Morbi blandit ligula feugiat magna. Nunc eleifend consequat lorem. Sed lacinia nulla vitae enim. Pellentesque tincidunt purus vel magna. Integer non enim. Praesent euismod nunc eu purus. Donec bibendum quam in tellus.

Nullam cursus pulvinar lectus. Donec et mi. Nam vulputate metus eu enim. Vestibulum pellentesque felis eu massa.

Quisque ullamcorper placerat ipsum. Cras nibh. Morbi vel justo vitae lacus tincidunt ultrices. Lorem ipsum dolor sit amet, consectetur adipiscing elit. In hac habitasse platea dictumst. Integer tempus convallis augue. Etiam facilisis. Nunc elementum fermentum wisi. Aenean placerat. Ut imperdiet, enim sed gravida sollicitudin, felis odio placerat quam, ac pulvinar elit purus eget enim. Nunc vitae tortor. Proin tempus nibh sit amet nisl. Vivamus quis tortor vitae risus porta vehicula.

Inhoudsopgave

1	Inleiding	15
1.1	Probleemstelling	15
1.2	Onderzoeksvraag	16
1.3	Onderzoeksdoelstelling	16
1.4	Opzet van deze bachelorproef	16
2	Stand van zaken	17
2.1	AZ Glorieux omgeving	17
2.1.1	Omgeving	17
2.1.2	Toekomst	17
2.1.3	IT Asset Management	18
2.1.4	Databronnen	19

2.2	LanReview	19
2.2.1	Visual Basic	19
2.2.2	LanReview werking	20
2.3	Low Code	21
2.3.1	Basics	21
2.3.2	Geschiedenis	24
2.3.3	Nood / Voordelen	25
2.3.4	Kritiek / Nadelen	27
2.3.5	Markt en Evolutie	28
2.3.6	Voorgaand onderzoek	30
2.4	Microsoft Power Platform: Power Apps	31
2.4.1	Wat?	31
2.4.2	Drie soorten apps	31
2.4.3	Belangrijke Onderdelen	32
2.4.4	IDE overzicht	34
2.4.5	Recente Wijzigingen	34
2.4.6	Pricing, Licencing	35
2.4.7	Uitbreidingsmogelijkheden	35
2.5	Microsoft Power Platform: Power Automate	35
3	Methodologie	37
3.1	Requirementsanalyse	37
3.1.1	Functioneel VS Niet-functioneel	37
3.1.2	MoSCoW-methode	38
3.1.3	Long List	39

3.1.4	Short List	40
4	Conclusie	43
A	Onderzoeksvoorstel	45
A.1	Introductie	45
A.2	State-of-the-art	47
A.3	Methodologie	48
A.4	Verwachte resultaten	49
A.5	Verwachte conclusies	50
	Bibliografie	51

Lijst van figuren

2.1	Voorbeeldweergave van de Visual Basic IDE (Speed, 2020)	19
2.2	Vergelijking van doorlooptijd (Richardson & Rymer, 2016)	25
2.4	Forrester voorspelde markt evolutie (J. Rymer, 2018)	29
2.5	Overzicht Microsoft Power Platform (Microsoft Docs, 2019d)	31
2.6	Overzicht van de formulebalk (Microsoft Docs, 2019a)	33
2.7	Overzicht van Power Apps Studio (Microsoft Docs, 2017)	34

Lijst van tabellen

2.1	Overzicht van de marktleiders	29
-----	-------------------------------------	----

1. Inleiding

De inleiding moet de lezer net genoeg informatie verschaffen om het onderwerp te begrijpen en in te zien waarom de onderzoeksvraag de moeite waard is om te onderzoeken. In de inleiding ga je literatuurverwijzingen beperken, zodat de tekst vlot leesbaar blijft. Je kan de inleiding verder onderverdelen in secties als dit de tekst verduidelijkt. Zaken die aan bod kunnen komen in de inleiding (Pollefliet, 2011):

- context, achtergrond
- afbakenen van het onderwerp
- verantwoording van het onderwerp, methodologie
- probleemstelling
- onderzoeksdoelstelling
- onderzoeksvraag
- ...

1.1 Probleemstelling

Uit je probleemstelling moet duidelijk zijn dat je onderzoek een meerwaarde heeft voor een concrete doelgroep. De doelgroep moet goed gedefinieerd en afgeleid zijn. Doelgroepen als “bedrijven,” “KMO’s,” systeembeheerders, enz. zijn nog te vaag. Als je een lijstje kan maken van de personen/organisaties die een meerwaarde zullen vinden in deze bachelorproef (dit is eigenlijk je steekproefkader), dan is dat een indicatie dat de doelgroep goed gedefinieerd is. Dit kan een enkel bedrijf zijn of zelfs één persoon (je co-promotor/opdrachtgever).

1.2 Onderzoeksvraag

Wees zo concreet mogelijk bij het formuleren van je onderzoeksvraag. Een onderzoeksvraag is trouwens iets waar nog niemand op dit moment een antwoord heeft (voor zover je kan nagaan). Het opzoeken van bestaande informatie (bv. “welke tools bestaan er voor deze toepassing?”) is dus geen onderzoeksvraag. Je kan de onderzoeksvraag verder specificeren in deelvragen. Bv. als je onderzoek gaat over performantiemetingen, dan

1.3 Onderzoeksdoelstelling

Wat is het beoogde resultaat van je bachelorproef? Wat zijn de criteria voor succes? Beschrijf die zo concreet mogelijk. Gaat het bv. om een proof-of-concept, een prototype, een verslag met aanbevelingen, een vergelijkende studie, enz.

1.4 Opzet van deze bachelorproef

De rest van deze bachelorproef is als volgt opgebouwd:

In Hoofdstuk 2 wordt een overzicht gegeven van de stand van zaken binnen het onderzoeksdomein, op basis van een literatuurstudie.

In Hoofdstuk 3 wordt de methodologie toegelicht en worden de gebruikte onderzoekstechnieken besproken om een antwoord te kunnen formuleren op de onderzoeksvragen.

In Hoofdstuk 4, tenslotte, wordt de conclusie gegeven en een antwoord geformuleerd op de onderzoeksvragen. Daarbij wordt ook een aanzet gegeven voor toekomstig onderzoek binnen dit domein.

2. Stand van zaken

2.1 AZ Glorieux omgeving

2.1.1 Omgeving

Er zitten meer dan 1000 toestellen in het domein waar controle en overzicht voor nodig is. Het gaat om een ziekenhuis. Er is geen marge voor fouten en respons moet direct kunnen zijn. Zowel Servers als client computers gebruiken Microsoft Windows besturingssysteem varianten. De softwarecatalogus is uitgebreid. Relevant is dat Office gebruikt wordt, ook Office 365. Er ligt een grote nadruk op documentering. Hiervoor wordt SharePoint gebruikt.

2.1.2 Toekomst

Migratie naar Windows 10 is een lopend proces. Veel bedrijven doen dit en proberen dit te versnellen wegens de komende end-of-life status van Windows 7 (Microsoft Support, 2020). Per case kan dit tijdrovend zijn om softwarecompatibiliteit te garanderen. Verspreiding van nieuwe technologieën: interesse in het Power Platform, implementatie van Microsoft Teams, Intune wordt steeds meer gebruikt als aanvulling van SCCM (System Center Configuration Manager). Er worden steeds meer mobiele devices opgenomen in het domein. Wat digitale transformatie betreft wordt sommige verouderde, huiseigen software vervangen. Concreet gaat het om tools gemaakt met Visual Basic. Er wordt gestreefd naar een hogere mate van automatisatie. Het is wenselijk dat zoveel mogelijk software gedeployed kan worden vanuit SCCM.

2.1.3 IT Asset Management

Het onderzoek bevindt zich in het domein van IT Asset Management. Enkele definities om dit begrip te verduidelijken.

IAITAM (g.d.) geeft een algemene definitie: 'IT Asset Management is a set of business practices that incorporates IT assets across the business units within the organization. It joins the financial, inventory, contractual and risk management responsibilities to manage the overall life cycle of these assets including tactical and strategic decision making.'

Gartner (g.d.) stelt: 'IT asset management (ITAM) provides an accurate account of technology asset lifecycle costs and risks to maximize the business value of technology strategy, architecture, funding, contractual and sourcing decisions.'

Ivanti (2018) reikt onderverdelingen aan. Er kan een onderscheid gemaakt worden tussen Fysieke, Digitale, Software, Mobile en Cloud IT Asset Management. Een beschrijving voor de voor het onderzoek toepasbare categorie, de fysieke, geldt: 'The discovery and inventory of hardware including PCs, laptops, printers, copiers, and any other device used for IT and data management purposes'.

Er werd verwacht dat de IoT (Internet of Things) trend IT Asset Management programma's zal laten evolueren om met deze devices om te kunnen gaan. Dit betekent meer assets en meer asset types. Hier zijn voordelen mee verbonden: 'Increased Operational Efficiency, Productivity Is Enhanced, Resources Are Used Efficiently, Better Checks for Safety and Compliance, Maintenance and Repair Automation' (Badnakhe, 2020).

IT asset management oplossingen zijn vaak een aanvulling op het werken van SCCM. Een belangrijke functie van SCCM is het verschaffen van informatie. In sommige gevallen wordt functionaliteit van SCCM opgeroepen of zijn database geraadpleegd. Het is de moeite om meer informatie over SCCM te geven.

Droogenbroot en Scheiris (2016) beschrijft het als volgt: 'System Center Configuration Manager, afgekort als SCCM of ConfigMgr, is een onderdeel van de Microsoft System Center Suite en staat in voor het beheer & inventarisatie van pc's en servers. Daarenboven is het mogelijk toepassingen uit te rollen, software updates te installeren en compliance te controleren en op te lossen'.

De primaire functies zijn het creëren van computer images, massa deployment van images, distribueren van silent installaties van software applicaties, cross campus managen van software en natuurlijk inventarisatie en organisatie van eigen hardware en devices.

SCCM functioneert aan de hand van een aantal sleuteltechnologieën: PXE booting, de SCCM agent geïnstalleerd op client pc's die data voed aan SCCM en Task Sequences waarmee een hele reeks taken uitgevoerd kan worden. (Spitze, 2019).

Eerder werd gesproken over de relatie tussen ITAM oplossingen en SCCM. Waar traditionele oplossingen een aanvulling of uitbreiding zijn op SCCM is LanReview eerder een vereenvoudiging. Het is toegespitst op inventarisatie en organisatie van eigen hardware en

devices.

De reden dat SCCM zo verspreid is is omdat het gratis inbegrepen is in enterprise licensing van Microsoft. Het is bijvoorbeeld inbegrepen in Microsoft 365 licentie. (Microsoft Docs, 2020a)

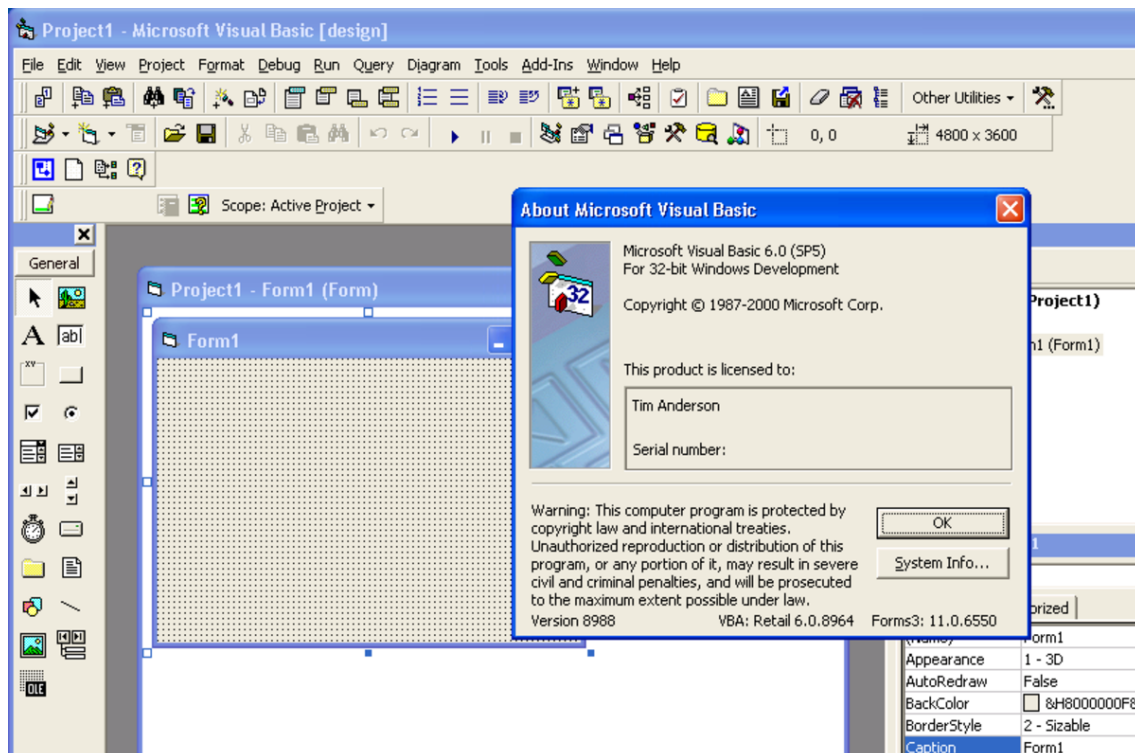
2.1.4 Databronnen

Een overzicht van enkele databronnen die AZ Glorieux gebruikt. Deze zullen later in het onderzoek aan bod komen.

- SQL Server
- Microsoft Excel en Access
- SharePoint

2.2 LanReview

2.2.1 Visual Basic



Figuur 2.1: Voorbeeldweergave van de Visual Basic IDE (Speed, 2020)

LanReview werd gebouwd in Visual Basic (VB). Dit is een event-gedreven programmeertaal en omgeving van Microsoft waarmee programmeurs code kunnen aanpassen door drag-en-drop van objecten en door wijziging van hun gedrag en uiterlijk. VB komt van de

BASIC programmeertaal. Het is een RAD (Rapid Application development) platform. Het werd voornamelijk gebruikt voor prototyping en als front-end voor databases. De laatste versie, Visual Basic 6.0, stamt van 1999.

Het grote voordeel is snelheid van ontwikkeling. Er zijn ook een aantal nadelen: Het had veel geheugen nodig en was niet geschikt voor programma's die veel proceskracht nodig hadden, zoals games. Het is ook beperkt tot het Microsoft besturingssysteem.

(Rouse, 2019)

Geschiedenis

- **1964:** BASIC geformuleerd door John Kemeny en Thomas Kurtz.
- **1987:** architect/programmeur Alan Cooper bedenkt voorloper van VB genaamd Tripod.
- **1988:** Bill Gates koopt de rechten voor Tripod.
- **1991:** Visual Basic 1.0 geïntroduceerd.
- **1998:** Visual Basic 6.0 geïntroduceerd.
- **2002:** .NET framework.
- **2008:** Einde extended support voor Visual Basic 6.0.

(Grigonis, 2014)

Vandaag

Uitvoering van VB applicaties blijft ondersteund op moderne Windows versies. Het maken van nieuwe echter niet (Microsoft Docs, 2018). Niettegenstaande heeft men manieren gevonden om alsnog te installeren. (Brust, 2015)

Er is nog een kleine maar vocale groep aanhangers van VB. Dit komt omdat ze de officiële opvolger, Visual Basic .NET, niet waardig vonden voornamelijk omdat de afhankelijkheid van .NET een extra abstractielaag was. Voorbeelden hiervan zijn:

- Visual Basic behoud blijft hoog. Opvallend aanhoudend succes voor een 'dode' tool. (ISplitter, 2014).
- Er werd een petitie gemaakt voor verdere ontwikkeling van VB. (classicvb petition, 2005)
- Er worden nog steeds applicaties mee ontwikkeld (Ippolito, 2018).

2.2.2 LanReview werking

LanReview houdt zijn informatie over assets in het domein bij in een Access databank. Deze informatie werd origineel geëxporteerd uit SCCM. De applicatie heeft twee views: de primaire view is een filterbaar overzicht van devices. voor elke entry kan een meer gedetailleerde view opgeroepen waar alle datavelden te zien zijn. Aan de hand van de filter functionaliteit worden SQL achtige queries gebouwd die dan uitgevoerd worden op de

Acces databank.

LanReview is een belangrijke tool van de Helpdesk en wordt dagelijks gebruikt.

Vaak voorkomende use cases:

- De pc overnemen van iemand in nood. Het is een startpunt voor remote management.
- Rapportage. Er kunnen voorgebouwde query's uitgevoerd worden op de data waarvan de resultaten onder andere naar Excel geëxporteerd kunnen worden.
- Filteren van de dataset. Bijvoorbeeld een overzicht van alle laptops van een belaad model tonen die nog steeds met Windows 7 werken.
- End of life beheer. Via markeringen is duidelijk of een device nog niet gebruikt wordt, in gebruik is of niet meer gebruikt wordt/in storage is.

2.3 Low Code

2.3.1 Basics

De POC zal gebouwd worden met een low-code platform. Cloud services maken hier een deel van uit. Moderne low-code platformen zijn gecategoriseerd als PaaS (Platform as a Service). In geval van IaaS (infrastructure as a service) leeft de volledige infrastructuur in de cloud. PaaS kan gezien worden als een extra laag gebouwd bovenop IaaS waarmee applicatieontwikkelaars software kunnen programmeren in de cloud om later aan te kunnen bieden als SaaS (Software as a Service). Het is een tussenlaag gefocust op programmeurs. (Nucleus, 2017)

Definiëren van wat low-code zoal inhoud kan best begonnen worden met een formele definitie:

'It is an application platform that supports rapid application development, one-step deployment, execution and management using declarative, high-level programming abstractions, such as model-driven and metadata-based programming languages. They support the development of user interfaces, business logic and data services, and improve productivity at the expense of portability across vendors, as compared with conventional application platforms.' (Vincent, Iijima, Driver & Natis, 2019)

De bedoeling van low-code is om applicaties sneller te kunnen maken en ontwikkeling voor een grotere groep toegankelijk te maken. Typisch gezien wordt er een WYSIWYG (What You See Is What You Get) interface gebruikt waarin visuele componenten geconfigureerd worden via drag-and-drop. Vaak is er ondersteuning om aangepaste toe te voegen indien het platform bepaalde functionaliteit out of the box niet ondersteund. (Kissflow, 2018a)

Low code is een software ontwikkelingsaanpak dat snellere oplevering van applicaties mogelijk maakt met minimaal manueel coderen. Dit werkt met visueel modeling in een grafische user interface om applicaties op te zetten en configureren. Hiermee slaat de ontwikkelaar de infrastructuur en her-implementatie van patronen die hun werk vertraagt en kunnen ze focussen op de unieke 10% van een applicatie. (Revell, 2020)

Kissflow legt in zijn omschrijving nadruk op een grotere toegankelijkheid en Outsystems focust meer op het kunnen vermijden van herhalend werk.

Een low-code platform bestaat typisch uit volgende delen:

- Een visuele IDE (Integrated Development Environment).
- Connectoren naar back-ends of services.
- App Lifecycle Management. Hiermee bedoelt geautomatiseerde tools voor build, debug, en deploy. Ook beheer van de app tijdens test, staging en productie.

(Revell, 2020)

low-code VS no-code

Een no-code platform is een gespecialiseerde versie van een low-code platform waar men manuele aanpassingen zoveel mogelijk wil vermijden, al dan niet door het pre-builden van de nodige visuele componenten.

Bloomberg (2017) nuanceert de verschillen door te focussen op de bedoelde gebruiker:

- low-code: Volgende generatie applicatie ontwikkeling dat het werk van professionele ontwikkelaars versneld en stroomlijnt.
- no-code: Self-service applicatie assembly voor business users die citizen developers worden.

low-code development VS traditional software development

Traditional	Low Code
<ol style="list-style-type: none"> 1 requirements bepalen 2 architectuur plannen 3 selectie van back-end framework, bibliotheken, data-stores en API's 4 selectie van front-end framework 5 keuze van deployment stack, opzetten van CI (Continuous Integration) en operatieplan creëren 6 wireframes en prototypes maken 7 ui coderen in gekozen Javascript framework 8 falende tests schrijven 9 modellen definiëren en koppelen aan data stores 10 business logic definiëren en coderen 11 views maken om JSON data te voorzien/ontvangen van en naar de front-end 12 implementeren van workflows 13 integreren van derde partij API's via gepubliceerde interface of support bibliotheek 14 herhalen tot tests slagen 15 testen voor security, performance, kwaliteit en gebruikersacceptatie 16 deploy, patch, monitor en updat tot levenseinde van de app 	<p>requirements bepalen derde partij API's selecteren app workflow, data models en user interface tekenen in de API's verbinden indien nodig handgeschreven code toevoegen aan front-end gebruikersacceptatietests deployment naar productie, hierna updates eenvoudig doorvoeren</p>

(Revell, 2020)

Citizen developers

De survey uitgevoerd door McKendrick (2017) geeft de volgende definitie:

‘we define citizen developers as business users, not part of IT departments or contracted IT services, who build and use their own scripts, programs, algorithms, or interfaces designed to perform business functions or support business processes.’

Er zijn enkele interessante statistieken uit voortgekomen:

- Snelheid van applicatie delivery en delen van data/analytics worden aanzien als zwakke gebieden van IT support.
- 76% van de respondenten zegt dat op z’n minst een deel van de gebruikte applicaties buiten het IT departement komen.
- 45% van citizen development activiteiten worden tijdens de werkuren gedaan, de grootste motivatie hiervoor (42%) is omdat het IT departement te traag is.
- In slechts 17% van de gevallen duurt ontwikkeling van een citizen developer applicatie langer dan drie maand.

Mythes

Er bestaan enkele mythes rondom low-code:

- Minder geschikt voor professionele gebruikers dan voor citizen developers
→ In praktijk wordt low-code vaker gebruikt door professionele ontwikkelaars.
- De nood om te programmeren wordt geëlimineerd
→ Het is vaak mogelijk om eenvoudigere applicaties te bouwen zonder code. Een typische business applicatie zal echter extra programmatie nodig hebben voor integratie met andere applicaties en databases, ook om aangepaste algoritmen te kunnen gebruiken. Deze use cases worden als extensies, externe programmatie of als scripts toegevoegd.
- low-code betekend kleine schaal
→ Er zijn genoeg praktijkvoorbeelden die dit tegenspreken. Neem bijvoorbeeld Sapphire HMS. (Bashar, 2017)

(Richardson & Rymer, 2016)

2.3.2 Geschiedenis

De oorsprong van low-code kan best bekeken worden vanuit het perspectief van de verranderingen in complexiteit en bruikbaarheid van programmeertalen doorheen de jaren. Er zijn enkele grote sprongen waarneembaar:

- **Fortran:** niet intuïtief, nuttig voor wetenschappelijke en numerieke computing.
- **Cobol:** Focus buiten wetenschappers of wiskundigen. Syntax staat dicht bij Engels. Het hielp bij het vinden van oplossingen naar business taken.
- **C:** Geschreven met Engelse syntax. Bruikbaar in een grote variatie aan applicaties.

- **Internet:** Kwam door het internet en een groeiende populariteit van webapplicaties. Men is kleinere en eenvoudigere scripts beginnen gebruiken in plaats van complexe programmeertalen. Er was een focus op functie. Applicaties moeten aan een snellere pas ontwikkeld kunnen worden en talen moeten eenvoudig genoeg zijn om dit te ondersteunen.

Een voorganger van low-code dat ook uit het internet is voortgekomen is het content management systeem. Wordpress is een bekend voorbeeld. Er zijn gelijkenissen in ideologie: het laat toe om de inhoud van een website aan te passen met een minimale hoeveelheid programmeren. Er worden modules gebruikt voor vaak voorkomende requirements. (Kissflow, 2018a)

(Kissflow, 2018b)

De ideologie achter low-code bestaat al een lange tijd. Er zijn cyclussen waar te nemen waarin de populariteit piekte. Een iteratie van 10 jaar geleden is 4gl (Fourth Generation Language). Een iteratie van 20 jaar geleden is RAD (Rapid Application Development).

Visuele en declaratieve development tools bestaan al decennia maar elke iteratie zijn er minder technische skills nodig om complexere business apps te kunnen maken die ook kunnen omgaan met moderne scenario's. Moderne low-code platformen overstijgen de limieten van op een aantal manieren:

Ze zijn meer 'open'. Vaak worden platform API's en adapters ondersteund. Soms is mogelijk om Java of .NET code te genereren als maatregel tegen vendor lock-in. Sommige platformen gebruiken zelf open source development frameworks zoals AngularJS, Apache Cordova en Bootstrap.

De platformen zelf zijn 'completer'. De vorige generaties waren eerder tools. Moderne low-code platformen ondersteunen de volledige applicatie levenscyclus.

Betere integratiemogelijkheden. De meeste low-code platformen laten toegang tot tot externe data toe. Vaak zijn er API's voor integratie met externe applicatie modules.

(Richardson & Rymer, 2016)

2.3.3 Nood / Voordelen

Enterprise	Result	Code	Low-code
US government (Affordable Care Act)	Document compliance module	100 person-months	5 person-months
British insurance provider	Agent portal	Unknown*	10 days to minimum viable product (MVP)
Call center operator	Customer-specific app	4 months	3 weeks
Spanish insurance provider	Web channel and administration system	2.7 years (estimated)	13 weeks

*The project was on the technology management backlog list for years with little hope of ever starting

Figuur 2.2: Vergelijking van doorlooptijd (Richardson & Rymer, 2016)

Het verkooppunt van low-code is hogere snelheid van applicatieontwikkeling. Naast deze

snelheid zijn er nog andere voordelen:

- Betere collaboratie tussen business en IT: Business shareholders zien hun visie sneller vorm nemen en kunnen rappe aanpassingen doorvoeren. Business regels zijn geïntegreerd in het bouwen van de applicatie.
- Focus op functionaliteit: Er wordt geen tijd verspild met schrijven van herhalende code.
- Transparante deployment: Er zijn geen aanpassingen meer nodig om verschillende architecturen te kunnen ondersteunen.
- Lange termijn gebruik: Apps zijn eenvoudig aan te passen.
- Enkele code base: Minder ruimte voor fouten en de code base wordt ook niet vervuild met tijd.

(van Schetsen, 2016)

Markt leidende platformen hebben de volgende eigenschappen:

- Eenvoudige visuele configuratie
- Veel integratie opties.
- Mobile compatibel.
- Schaalbaar.
- Support over de volledige app levenscyclus.

(Kissflow, 2018a)

Het perspectief van de klant, welke waardepropositie er is. Dit is duidelijker als er eerst naar de omgeving gekeken wordt. Bedrijven moeten steeds meer kunnen omgaan met disruptieve innovatie en wijzigend klantgedrag. De vraag naar applicaties is vaak groter dan wat geleverd kan worden. Er zijn strategieën om hier mee om te proberen gaan zoals outsourcing, hackathons of voorgebouwde softwareoplossingen. Low code is een nieuwe manier om er mee om te kunnen gaan. De waardepropositie voor de business en het tech management is:

- Apps visueel configureren in plaats van manueel te coderen. Ontwikkeling kan zo mogelijk met een factor 5 tot 10 versneld worden.
- Echte requirements en zo echte waarden vinden. Met low-code kunnen snel minimum viable products gebouwd worden om klantenideeën te toetsen.
- Live-trial van business ideeën tegen lage of onbestaande kost. Ideeën kunnen snel omgezet worden in een werkend prototype dat gedeployed en getest kan worden op de markt.
- Werkende prototypes kunnen in vijf minuten omgezet worden in productie apps. Het komt niet voor dat een app opnieuw gebouwd moet worden om een hoger volume of hogere diversiteit van gebruikers aan te kunnen.
- Meer development talent mogelijk. Developers zonder formele achtergrond kunnen door het visuele aspect nu ook business applicaties bouwen.

(Richardson & Rymer, 2016)

Digitale transformatie

Digitale transformatie is het proces van het gebruik van digitale technologie om nieuwe - of bestaande - business processen, cultuur en klantenexperience te maken als antwoord op veranderende business en markt requirements. Het is kortweg opnieuw bedenken van business in het digitale tijdperk. (Salesforce, g.d.) Digitale transformatie wordt steeds belangrijker als arena waarin business een voordeel kan krijgen op de competitie. Low-code wordt hierbij aanzien als een krachtig wapen in dit proces.

2.3.4 Kritiek / Nadelen

Er zijn een aantal valkuilen waar rekening mee gehouden moeten worden bij de keuze van een platform. De inzetting kan falen en het gewenste resultaat kan uitblijven als de klant geen concreet plan heeft over hoe low-code in te werken in zijn portfolio. Het is ook aangeraden om op te passen indien er een kleinere vendor gekozen wordt. Het is een veranderende markt en deze hebben een lagere stabiliteit. (Richardson & Rymer, 2016)

Een mening is dat de business kant zijn mentaliteit ten opzicht van IT moet aanpassen. Al decennia probeert met de nood van programmeren terug te drijven en in functie hiervan worden tools gemaakt. Bedrijven zijn hiertoe aangetrokken om kosten te kunnen besparen. Maar de achterliggende kennis blijft nodig, vooral als er iets fout gaat. Niet alles kan visueel gedaan worden. Development talent blijft dus nog nodig. De aanpassing in mindset die gevraagd wordt is dat business het feit aanvaard dat software ontwikkeling duur en moeilijk is en dat er geen shortcuts voor zijn. (Reselman, 2018)

Er kan gesteld worden dan niet elk low-code platform daadwerkelijk 'low-code' is en dat men hiervoor moet oppassen:

- **NIET** low-code: Het platform genereert statisch code en verstopt deze effectief voor de gebruiker. De code is niet flexibel en schaalbaar. Dit staat digitale transformatie in de weg en verhoogt technical debt in plaats van dit te verlagen.
- **WEL** low-code: Het platform is flexibel en schaalbaar. De volledige levenscyclus van de app moet ondersteund worden. Achterliggend wordt een model gedreven architectuur gehanteerd. De flexibiliteit komt doordat de model componenten geabstraheerd worden naar XML. De applicatie kan ook makkelijker gewijzigd worden en is eenvoudig schaalbaar.

(Shiah, 2018)

Shadow IT

Gebruik van no-code tools in de handen van citizen developers kan mogelijk leiden tot shadow IT. Dit komt voor wanneer de IT organisatie te traag en proces-gebonden is om snel genoeg te kunnen reageren op de noden van business en business users buiten IT gaan om zelf een app bouwen. Zonder overzicht en coördinatie kunnen security kwetsbaarheden en compliance overtredingen verspreiden. Deze apps kunnen ook redundant en van lage

kwaliteit zijn. (Bloomberg, 2017)

2.3.5 Markt en Evolutie



(a) Forrester Wave (J. R. Rymer & Koplowitz, 2019) (b) Gartner Magic Quadrant (Vincent, Iijima, Driver & Natis, 2019)

Het business model van low code platformen is anders dan dat van traditionele platformen zoals Java of .NET. Er wordt uitgegaan van geleverde business waarde in plaats van belofde. Zo zijn veel platformen gratis of tegen lage kost toegankelijk. Inkomst wordt gegenereerd uit benoemde gebruikers, gedeployde apps en nodige capaciteit. (Richardson & Rymer, 2016)

Wat betreft marktsegmentatie heeft er zicht op relatief korte tijd veel evolutie voorgedaan. Er kunnen drie grote stappen onderscheiden worden:

1. 5 marktsegmenten waarvan de grootste general purpose is. Het doel is grote variatie aan web en mobiele applicaties kunnen bouwen. Declaratieve tooling gebruikt met aandacht voor creatie, integratie, deployment life-cycle management en distributie van applicaties. Hiernaast waren ook nog segmenten voor process, database, request-handling en mobiele apps.
2. Samensmelting naar general purpose
3. Splitsing om te focussen op gebruiker. Business user of professionele ontwikkelaars.

Er worden een aantal marktontwikkelingen verwacht:

- **Consolidatie.** Grote verkopers zullen low-code platformen overnemen.
- **Marktsegment veranderingen.** Low code for mobile verdwijnt en low-code for process groeit naar general purpose segment.
- De volgende innovatie zal **ondersteuning voor IoT** zijn.

(Richardson & Rymer, 2016)

J. R. Rymer en Koplowitz (2019) identificeert 5 marktleiders. Opvallend is dat Microsoft in

Forrester	Salesforce	Microsoft	Mendix	Outsystems	Kony
Gartner	Salesforce	Microsoft	Outsystems	Mendix	Appian

Tabel 2.1: Overzicht van de marktleiders

een korte periode naar een leidingspositie is gesprongen. De bevindingen van Richardson en Rymer (2016) sluiten hierbij aan.

Prospecten voor de komende vijf jaar:

- Tegen 2024 zal drie vierde van grot enterprises ten minste vier low code development platformen gebruiken voor zowel IT app development en citizen development initiatieven.
- tegen 2024 zal low-code applicatie development verantwoordelijk zijn voor 65% van totale applicatie ontwikkeling activiteit.

(Vincent e.a., 2019)

De verschuiving van traditionele naar low-code software ontwikkeling is terug te vinden in huidige jobaanbiedingen¹².

Er wordt verwacht dat de low-code markt tegen 2022 21 biljoen dollar waard zal zijn. Er wordt ook voorspeld dat de jaarlijkse groei van 50% zich verder zal zetten. (J. Rymer, 2018)

Projected low-code development platform market growth

Year-over-year growth rate



(F) = Forrester forecast

Note: The figure for 2016 is estimated.

Source: Forrester estimates, vendor reporting

Figuur 2.4: Forrester voorspelde markt evolutie (J. Rymer, 2018)

¹ <https://stagebank-hbo-ict.irp.nl/internships/12543/>

² <https://stagebank-hbo-ict.irp.nl/internships/12378/>

2.3.6 Voorgaand onderzoek

Er is al redelijk wat praktisch onderzoek gevoerd die het potentieel van low-code illustreert maar het grootste succesverhaal is dat van Sapphire HMS. Dit is een low-code applicatie binnen gezondheidszorg. Koeweit had nood aan een nieuw ziekenhuis managementsysteem. Na initieel falen om iets manueel te bouwen werd gekeken naar low-code platformen en was het mogelijk om de eerste implementatie na 6 maanden in gebruik te nemen. (Bashar, 2017)

Proof of Concepts

Er werd onderzoek gedaan naar de mogelijke meerwaarde van low-code in een KMO omgeving. Uit vier marktleidende platformen werd Outsystems gekozen. Er werden POC's voor vaak voorkomende scenario's mee opgesteld. De bouw was snel, eenvoudig en er waren geen obstructies. De nodige investering van tijd, geld en kennis was beperkt (Devloo, 2018).

Microsoft Power Apps werd gebruikt om enkele administratieve bedrijfsprocessen van verhuisbedrijf De Borger te optimaliseren. Het resultaat werd effectief in gebruik genomen de tijds winst sinds implementatie werd berekend en significant bevonden. De prijs was aanvaardbaar. Er was slechts 5 euro per maand nodig om een aangepaste connector te kunnen gebruiken. (Spriet, 2019)

In een ander geval werd Power Apps gebruikt om een mobiele reporting applicatie te maken voor gebruik in de bouwsector. Er werd ook gebruik gemaakt van Power Automate om afbeeldingen te verwerken en SharePoint om deze op te slaan. (Aho, 2018)

Binnen de gezondheidszorg werd er een case study gedaan naar hoe onderzoeksdata verzameld kon worden door gebruik van een low-code platform. Er werd voor Mendix gekozen. De veiligheid van het platform werd als goed bevonden, relevant privacywetten waren gerespecteerd en authenticatie werd ondersteund en was eenvoudig te implementeren. Er werd verhoogde productiviteit waargenomen. (Totterdale, 2018)

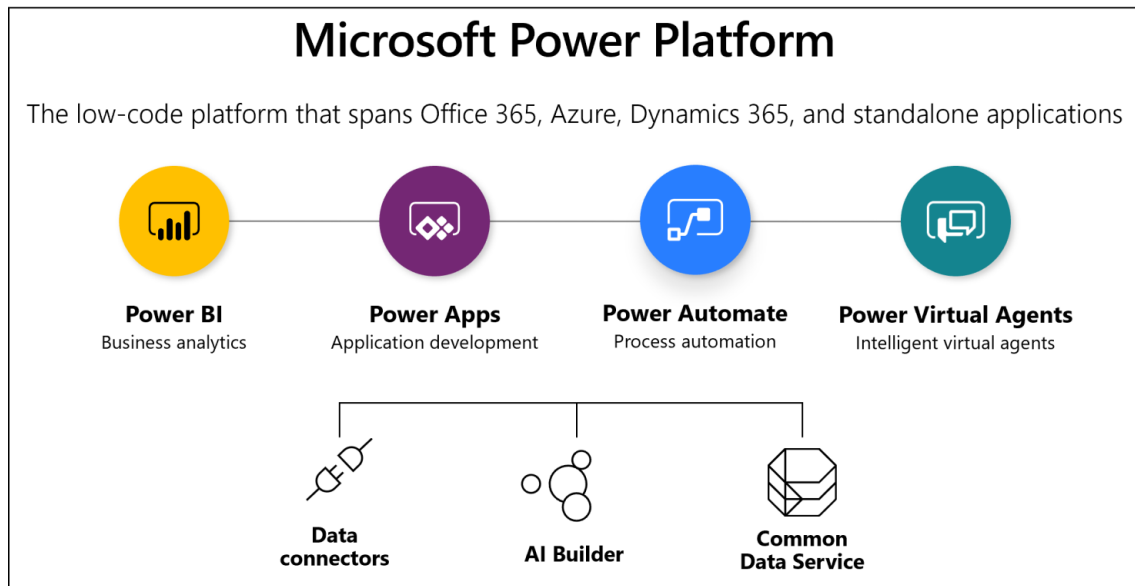
Enquetes

Er werd onderzocht of het mogelijk zou zijn om een EHR (electronic Health Records) systeem te bouwen met low-code om te gebruiken op nationaal niveau. Dit was naar aanleiding van het Akson planningsproject voor een nieuw EHR systeem in Noorwegen. Er werden kwalitatieve interviews afgenomen bij mensen met expertise in low-code en mensen met kennis van het Akson project. (Ness & Hansen, 2019)

Low-code en traditionele softwareontwikkeling werden met elkaar vergeleken aan de hand van interviews met medewerkers bij een Fins software consultancy bedrijf. Er kon besloten worden dat zowel de medewerkers als de klanten oplossingen gebouwd met low-code verkozen. (Virta, 2018)

2.4 Microsoft Power Platform: Power Apps

2.4.1 Wat?



Figuur 2.5: Overzicht Microsoft Power Platform (Microsoft Docs, 2019d)

Power Apps is een suite met apps, services en connectors. Het is een gegevensplatform die een snelle ontwikkelomgeving biedt om aangepaste apps te bouwen voor zakelijke behoeften. Men kan verbinden met zakelijke gegevens die zijn opgeslagen ofwel in het onderliggend gegevensplatform (Common Data Services), ofwel in de verschillende online en on-premises gegevensbronnen (SharePoint, Excel, Office 365, Dynamics 365, SQL Server enzovoort). De apps hebben een responsief ontwerp en kunnen zowel in een browser of op mobiele apparaten (telefoon of tablet) uitgevoerd worden. (Microsoft Docs, 2019g)

2.4.2 Drie soorten apps

Canvas apps

Populairste en meest toegankelijke variant. Er is een focus op het grafische. Zo worden de apps via een WYSIWYG (What You See Is What You Get) manier gebouwd. Daarmee is er volledige controle over de User Experience. Er is zowel tablet als mobile layout mogelijk. Men kan verder connecteren met meerdere data bronnen. Het resultaat zijn taak- en rol gebaseerde apps. (Pragmatic Works, 2019)

Model Driven apps

In tegenstelling tot Canvas apps begint men hier bij het data model. Deze data moet ook in de CDS leven. Het doel is meestal om complexe business processen te visualiseren. Het resultaat zijn dan ook end-to-end business applicaties. (Pragmatic Works, 2019)

Portals

Waar Canvas en Model apps naar binnen, naar internet business gebruikers gericht is, is Portals naar buiten gericht. Met portals kunnen extern gefocuste websites gebouwd worden waarmee gebruikers buiten de organisatie kunnen inloggen met een variëteit aan identiteiten, data in de Common Data Services kunnen bekijken of inhoud anoniem browsen. Er zijn zowel interne als externe gebruikers mogelijk. (Microsoft Docs, 2020c)

2.4.3 Belangrijke Onderdelen

Common Data Service

Met de common data service kan data gebruikt door business applicaties opgeslagen en gemanaged worden. Data wordt opgeslagen binnen een set entiteiten. Een entiteit is een set van records om data op te slaan, zoals tables data opslaan in een database. Er zijn standaard entiteiten aanwezig voor typische scenario's, er kunnen ook aangepaste entiteiten gemaakt worden. Data wordt opgeslagen in de cloud. Er is ook logica en validatie mogelijk in de vorm van berekende velden, business regels, workflows en business process flows. Het 'common' aspect: De databronnen zijn gelinkt, als data in de ene bron aangepast wordt zal de relevante data in de andere bron automatisch aangepast worden. (Microsoft Docs, 2019d)

Connectors

Data die een app nodig heeft is opgeslagen in een data bron, deze data naar de app brengen wordt gedaan door er een connectie voor te maken. Deze connectie heeft een specifieke connector nodig om te kunnen praten met de data bron. Power Apps heeft connectors voor enerzijds populaire services en anderzijds on-premises data opslag. Een connector kan in de app zelf dan ofwel tabellen of acties voorzien.

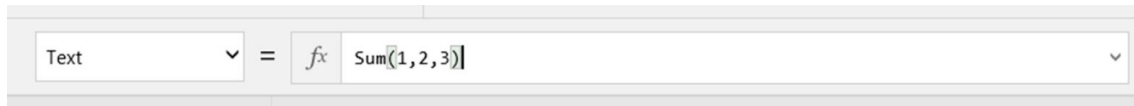
Een overzicht van de belangrijkste connectors:

- Common Data Service
- Cloud storage: Voor elke grote provider. Bijvoorbeeld Google Drive of OneDrive.
- Excel: om excel data te kunnen tonen in de app. Requirements zijn dat de data geformatteerd moet worden als tabel en het bestand in de Cloud opgeslagen moet zijn.
- Office 365 (Outlook): Onder andere mogelijk om mails te tonen, versturen, beantwoorden of te deleten.
- Office 365 (Users): Toegang tot de gebruikersprofielen in de organisatie.
- Oracle DB
- Power BI
- SharePoint
- SQL Server
- Twitter

(Microsoft Docs, 2020b)

On-premises data gateway: Om on-premises data bronnen te kunnen gebruiken binnen een Power is moet er een data gateway voor aanwezig zijn. Dit is een soort van brug tussen de on-premises data en de Microsoft cloud services die veilige data overdracht garandeert. (Microsoft Docs, 2019f)

Formules



Figuur 2.6: Overzicht van de formulebalk (Microsoft Docs, 2019a)

Logica en werken met gegevens is mogelijk met Excel-achtige expressies. In Excel worden formules gebruikt om cellen te vullen of grafieken en tabellen te maken. In Power Apps worden vergelijkbare formules gebruikt om besturingselementen in plaats van cellen te configureren. Concreet ook om om te gaan met user input. De granulariteit is: een app bestaat uit UI controls → elke control heeft properties → per property kunnen formule(s) ingesteld worden. (Microsoft Docs, 2019a)

Overig

Nadat werk aan een app opgeslagen is moet deze gepubliceerd worden om bruikbaar te zijn voor ander leden van de organisatie.

Versiebeheer is beperkt aanwezig. Er kan eenvoudig teruggegaan worden naar eerder gepubliceerde versies van een app.

UI Tests zijn mogelijk met de Test Studio maar op moment van schrijven is dit nog experimenteel. Power Apps Test Studio is een low-code oplossing om tests te schrijven, organiseren en automatiseren voor canvas apps. In de Test Studio kunnen tests geschreven worden via Power Apps expressies of door gebruik van een recorder om app interactie op te slaan en expressies automatisch te genereren. De tests kunnen hierna terugspeeld worden binnen de Test Studio om app functionaliteit te valideren.

Gebruikte concepten en terminologie komen overeen met gangbare testing frameworks zoals bijvoorbeeld JUnit voor Java.

Meer uitleg aan de hand van deze terminologie.

- **Test Cases:** Test cases bestaan uit een serie instructies of acties genaamd test stappen. Test cases worden uitgevoerd om te controleren of apps of specifieke features in de app werken zoals verwacht. Deze test stappen zijn geschreven in de Power Apps expressie taal.
- **Test Suites:** Gebruikt om test cases samen te groeperen.
- **Test Assertions:** gebruikt om te valideren of het verwacht resultaat overeen komt met het verkregen resultaat. Evalueert naar true of false.

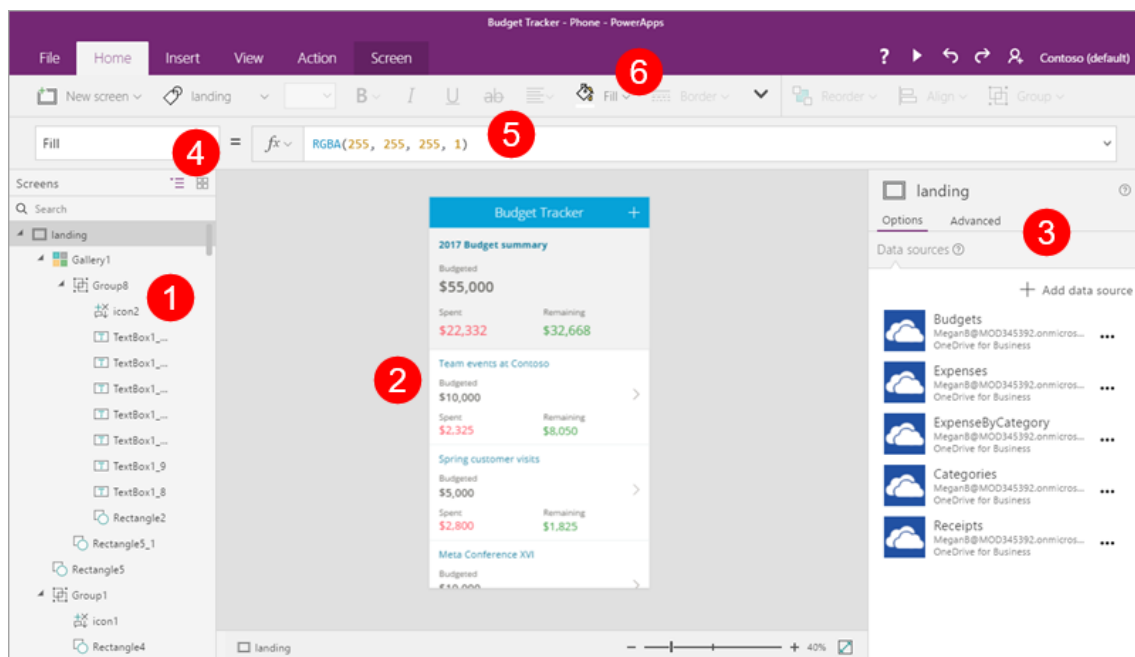
(Microsoft Docs, 2019c)

Tot op heden werd is er nog geen IT Asset Management app gebouwd in Power Apps die als template terug te vinden is. Wel bestaat er een Asset checkout app. Dit staat er dicht genoeg bij. (Meganathan, 2019)

2.4.4 IDE overzicht

De onderdelen van Power Apps Studio uitgelicht volgens Figuur 2.7:

1. Links: Hiërarchische view van de app screens en controls.
2. Centraal: Huidige app scherm.
3. Rechts: Geavanceerde layout en data bronnen.
4. Property lijst voor geselecteerde control.
5. Formulebalk.
6. Ribbon.



Figuur 2.7: Overzicht van Power Apps Studio (Microsoft Docs, 2017)

2.4.5 Recente Wijzigingen

De Test studio valt hierbij (zie subsectie 2.4.3).

De AI Builder is in publieke preview sinds 10 juni 2019. Dit maakt het mogelijk om AI te gebruiken in Power Apps met minimale technische kennis.

Een voorbeeld van het typische verloop: Een AI model type kiezen → Data verbinden (uit CDS) → AI model aanpassen naar noden (data filteren, scheduling) → AI model trainen (gebeurt automatisch) → De inzichten van het AI model gebruiken doorheen het Power

Platform.

Wat AI modellen betreft zijn er enkele keuzes: enerzijds kan er aan aangepast model gebruikt gemaakt worden (voorspelling, form processing, object detectie, text classificatie), anderzijds kan geopteerd worden voor een voorgebouwd. In dat geval is er geen data nodig, het is gebouwd (en getraind) door Microsoft. Mogelijke keuzes zijn: Business card lezer, sleutel zin extractor, taal detectie, text herkenning, gevoelsanalyse. (Microsoft Docs, 2019e)

2.4.6 Pricing, Licencing

- Power Apps voor Office 365.
- Plan 1
- Plan 2
- Power Apps for Dynamics.

(Pohl, 2019)

Elke entry in de lijst heeft meer functionaliteit dan de voorgaande.

In Power Apps for office is het enkel mogelijk Canvas apps te maken. Het aantal connectors is beperkt. De on-premise connectors bijvoorbeeld zijn niet aanwezig.

2.4.7 Uitbreidingsmogelijkheden

Power Apps studio is bedoelt als no code omgeving gefocust op business gebruikers. Binnen deze ui is er geen optie om aangepaste code te gebruiken. De enige manier om dit mogelijk te maken is via Custom API's, Azure functions of Azure API apps³.

Praktisch gezien is dit mogelijk door een project te maken in Visual studio, een OpenAPI definitie te voorzien. Als deze app in de Azure cloud staan kan deze ook opgeroepen worden vanuit Power Apps (Jugo, 2019)

2.5 Microsoft Power Platform: Power Automate

Power automate is een service voor het maken van geautomatiseerde workflows tussen apps en services. Een flow kan gemaakt worden door stappen aaneen te schakelen. Per stap kan een actie of conditie ingesteld worden. In deze condities kunnen data operaties geconfigureerd worden, er zijn ook expressies mogelijk. Naast het zelf bouwen van een flow zijn er sjablonen aanwezig voor gangbare automatisatie cases. Flows kunnen gebruikt worden vanuit Power Apps. (Microsoft Docs, 2019b)

Er zijn vijf soorten flows:

³Volgens het antwoord van een Microsoft medewerker op het Power App forum: <https://powerusers.microsoft.com/t5/Power-Apps-Ideas/add-your-own-js-in-powerapps-or-call-external-js-file-easily/idi-p/869>

- Geautomatiseerde flow: de flow wordt gestart vanuit een bepaalde trigger.
- Button flows: de flow wordt manueel gestart. Vaak wordt dit gebruikt vanop een mobiele device.
- Scheduled flows: De flow herhaald / op een schema uitvoeren.
- Approval flows
- UI flows: Een vorm van UI automatisatie, een optie indien er geen API of connector aanwezig is (bijvoorbeeld oudere toepassingen). Voor gebruik met Windows desktop en web applicaties. Momenteel nog in experimentele fase. Het kan gezien worden als een eenvoudige manier om een script te bouwen. De werking is dat de gebruiker een actie/set stappen opneemt, hier wordt dan een aanpasbare flow voor gegenereerd. (Microsoft Power Automate, 2019)

Beperkingen

Er is geen aangepaste code of scripting ondersteund.

3. Methodologie

3.1 Requirementsanalyse

De bedoeling van deze analyse is uitvinden hoe goed Power Apps voldoet aan de requirements en mogelijk ontdekken of er een alternatief bestaat dat overwogen kan worden. Als PowerApps in de toekomst effectief gebruikt zal worden, voornamelijk door het IT en Helpdesk team, is het dan niet logisch om dit alternatief te laten conformeren aan de bedoelde eindgebruiker, zijnde professionele IT'ers? Dit idee zal mee spelen bij de besluitvorming.

3.1.1 Functioneel VS Niet-functioneel

De requirements werden opgesteld tijdens herhaalde gesprekken met zowel Siemen als een lid van het Helpdesk team dat de app effectief zou gaan gebruiken. De resultaten zijn hieronder opgesplitst in functionele¹ en niet-functionele² requirements.

- **Functionele requirements**

- Overzicht kunnen geven van belangrijkste info voor elk toestel in het netwerk.
- Rapporten kunnen genereren.
- PC's bedienen vanop afstand (Remote Desktop Protocol kunnen oproepen).
- Gerichte/basis taken kunnen automatiseren.
- Bruikbaar zijn buiten het domein.
- Mobiel bruikbaar zijn.

¹Functionele requirement: gewenst gedrag van het systeem

²Niet-functionele requirement: kwaliteitseis waaraan het systeem moet voldoen

- Nieuwe types toestellen opnemen.
- Randapparatuur opnemen (= koppeling tussen toestellen).
- Revisie van elk dataveld per toesteltype.
- Data opslaan in SharePoint (cloud).
- Samenwerken met SCCM en/of synchroniseren met en data uit de SQL databank kunnen gebruiken.
- Barcode's kunnen scannen (toegang hebben tot camera, barcode functionaliteit ingebouwd).
- AI functionaliteit.
- Het Ping commando kunnen oproepen.
- Command line toegang hebben tot PC's.
- Intune integratie.
- **Niet-functionele requirements**
 - Prijs (geen prijsstijging voor de te implementeren case of voor het aantal gebruikers).
 - Future proof³ zijn.
 - Diverse GUI verbeteringen (specifiek kleur markeringen, tabbladen)/robuust GUI ontwerp ondersteunen.
 - Performant zijn (specifiek met grote hoeveelheden entries/rijen kunnen omgaan).
 - Leercurve moet degelijk zijn (het gebruik ervan moet aanslaan na het onderzoek).
 - Veiligheid (een must nodig omwille van gevoelige bedrijfsdata).

3.1.2 MoSCoW-methode

De volgende stap is prioriteiten stellen onder de requirements. Hiervoor wordt de MoSCoW-methode⁴ gebruikt.

- **Must have**
 - Prijs.
 - Overzicht kunnen geven van belangrijkste info voor elk toestel in het netwerk.
 - Rapporten kunnen genereren.
 - PC's bedienen vanop afstand (Remote Desktop Protocol kunnen oproepen).
 - Mobiel bruikbaar zijn.
 - Future proof zijn.
 - Performant zijn.
 - Veiligheid.
- **Should have**
 - Gerichte/basis taken kunnen automatiseren.
 - Bruikbaar zijn buiten het domein.

³Hiermee bedoelt een algemene combinatie van achterliggende technologie, innovatie, consistentie en marktaanwezigheid

⁴Must have: eis moet terugkomen in het eindresultaat, Should have: eis is zeer gewenst maar het product is bruikbaar zonder, Could have: eis komt aan bod als er genoeg tijd is, Won't have: komt niet aan bod, is voor de toekomst of een vervolgproject (Wikipedia, 2020)

- Nieuwe types toestellen opnemen.
- Randapparatuur opnemen.
- Revisie van elk dataveld per toesteltype.
- Data opslaan in SharePoint (cloud).
- Leercurve moet degelijk zijn.
- **Could have (nice to have)**
 - Samenwerken met SCCM en/of synchroniseren met en data uit de SQL databank kunnen gebruiken.
 - Diverse GUI verbeteringen/robuust GUI ontwerp ondersteunen.
 - Barcode's kunnen scannen.
 - AI functionaliteit.
 - Het Ping commando kunnen oproepen.
- **Won't have**
 - Command line toegang hebben tot PC's.
 - Intune integratie.

3.1.3 Long List

Er zijn veel platformen om uit te kiezen⁵. Degene hieronder opgelijst werden besproken in de Forrester Wave (J. R. Rymer & Koplowitz, 2019) en het Gartner Magic Quadrant (Vincent e.a., 2019). De low-code markt kent een sterke groei en is zeer bewogen (zie sectie 2.3.5). Het eerste gekozen criteria om de lijst te filteren is bedoelt om hiermee om te gaan, het kan immers aangenomen worden dat een platform bestempelt door Gartner of Forrester als leider voldoende future proof is. Het andere gekozen criteria betreft het licentie model. Power Apps is opgenomen in het Office 365 licentie van AZ Glorieux, initieel gebruik ervan betekend geen dus geen meerprijs, een concurrent moet met andere woorden iets gelijkaardig kunnen aanbieden.

Naam	Omschrijving	Leider		
		Forrester	Gartner	Freemium
AgilePoint				
Appian			x	
Betty Blocks				
bpm'online				
Clear Software				
GeneXus				
Kony		x		x
K2				
Kintone				
MatSoft				x
Mendix		x	x	x
Microsoft		x	x	(x)
Outsystems		x	x	x
Oracle				

⁵ 57 volgens een oplijsting van TrustRadius

Pega			
Progress Software			
ProntoForms			
Quick Base			
Salesforce	x	x	
ServiceNow			x
Skuid			
Thinkwise			
TrackVia			x
WaveMaker			
Zoho			x

Het resultaat is dat er van de leiders (zie tabel 2.1 voor een gefocuste weergave) slechts enkelen overblijven die in aanmerking komen.

3.1.4 Short List

Hier worden de overgebleven platformen in diepte met elkaar vergeleken. Maar in plaats van elke requirement af te toetsen werden deze eerst herleid naar hun achterliggende begrippen (bijvoorbeeld: mobile ondersteuning, automatisatie functionaliteit, security, leercurve en dergelijke).

Gemaakte stellingen zijn afkomstig uit PCmagazine.com reviews van elk platform en zijn aangevuld met informatie uit de officiële documentatie per platform.

Microsoft

Deel van het Power platform. Het bestaat sinds 2016 en is daarmee het jongste platform in de vergelijking maar heeft een snelle ontwikkeling ondergaan en is op slechts enkele jaren tijd een marktleider geworden.

Werd diepgaand behandeld in de (in sectie 2.4). Een kort overzicht:

Forrester:

Gartner:

Licentiëring:

Outsystems

Outsystems werd opgericht in Lissabon in 2001, zit momenteel in 11 landen en telt 1228 medewerkers.

Er is een focus op professionele ontwikkelaars, het is gemakkelijker om met code te werken.

Het is mogelijk om op elk moment te switchen van de grafische omgeving naar de code editor. Iets anders dat dit professioneel aspect ondersteund is dat de IDE (genaamd Service Studio) offline geïnstalleert dient te worden. De layout ervan komt sterk overeen met die van PowerApps en Mendix. Andere grote onderdelen zijn de Forge (respository voor gebruikersgemaakte apps en plug-ins) en de Integration Studio waar extensions geschreven worden.

Een nadeel van de technische focus is dat er geen volledig cloud no-code omgeving aanwezig is. Een ander nadeel is de hogere leercurve maar er is een uitgebreid aanbod van documentatie, tutorials online cursussen en webinars aanwezig in de OutSystems University om hiermee om te gaan.

Het platform is gebouwd in .NET. Bouwen van mobiele of webapplicaties is ondersteund maar er moet gekozen worden tussen de twee.

Een app is gestructureerd in modules: mobile (of web), service, library en extension modules. Dit om hergebruik aan te moedigen.

Automatisatie functionaliteit is beperkt tot Timers. Het is enkel mogelijk om bepaalde acties op een tijdschema uit te voeren. Wat AI betreft zijn er connectors naar Azure Luis en Azure ML. Inhuis voorziet OutSystems.AI taal analyse componenten (sleutelzin detectie, gevoelsanalyse).

UI design is minder geavanceerd dan bij de concurrentie en er is meer werk nodig om het gewenste resultaat te krijgen.

SharePoint gebruik is enkel mogelijk via de REST API. Dit is een grote beperking ten opzichte van Power Apps, waar het een standaard Connector is.

Het platform is veilig en voldoet aan volgende certificeringen: Om gerust te stellen voor vendor lock-in⁶ is het mogelijk om de applicatie te genereren in .NET.

Forrester:

Gartner:

Licentiëring:

Mendix

Mendix werd opgericht in Nederland in 2005, werd in 2018 overgenomen door Siemens voor 730 miljoen dollar. Er zijn enkele grote partnerships aangegaan, onder andere met SAP die Mendix resellt als het SQL Cloud Platform.

Net als bij OutSystems is de doelgroep eerder professionele ontwikkelaars. Er is zowel een volledige no-code cloud omgeving aanwezig (Mendix Studio) als een uitgebreidere variant (Mendix Studio Pro) die offline geïnstalleerd moet worden, gericht op developers. Het coderen gaat met Java.

Er is een groot aanbod aan voorgebouwde templates en componenten die Microsoft en OutSystems evenaren. De UI filosofie is om te starten met design en wireframes, dan het

⁶Gedwongen een product of service moeten blijven gebruiken, onafhankelijk van kwaliteit, omdat het (financieel) niet praktisch is om er vanaf te stappen (Cloudflare, g.d.)

model te maken met logica en workflows die in dat model passen. De apps zijn responsief. Het scherm wordt automatisch aangepast tussen smartphone, tablet of desktop views.

Database integratie is beter dan bij Outsystems. Database wijzigingen worden automatisch herkend in de app, bij Outsystems is dit niet het geval. Specifiek naar SharePoint toe is verbinding net als bij Outsystems beperkt tot een REST API.

App creatie is algemeen gezien gestroomlijnder dan bij Outsystems. Er kan na afloop gedeployed worden naar verschillende cloud omgevingen. App aanpassen zijn eenvoudig (versionering is voorzien) maar voor elke nieuwe versie van het platform moet er gemigreerd worden.

Testing en analytics zijn geavanceerder dan bij concurrenten. Waar testing voor PowerApps nog in beta zit is het bij Mendix meer volwassen, tests worden bijvoorbeeld automatisch uitgevoerd.

Over specifieke onderdelen: Buzz is een portaal dat dient als sociaal intranet waar collaboratie begint. SCRUM is ingebouwd in het platform. Integraties, plugins, gerbuikersapps zijn in de Mendix App Store te vinden, dit aspect is even matuur als varianten van Microsoft en Outsystems.

Wat AI functionaliteit betreft is er integratie mogelijk met IBM Watson.

Security is als verwacht voor een marktleider:

Forrester:

Gartner:

Licentiëring:

Salesforce

Er is geen gratis licentie aanwezig en het platform is eerder gefocust op CRM⁷

PCmag (**bibid**) was minder gunstig over het platform en hekelde zich vooral aan de feature bloot en UI clutter. De tutorials zouden overigens niet up to date zijn.

Om deze redenen is het geen valide optie maar omdat de marktaanwezigheid zo groot is was het de moeite om kort te bespreken.

Appian

Net als Salesforce een grote speler. functioneel even sterk als Outsystems en Mendix maar helaas is er geen gratis licentie mogelijk.

Temenos Quantum (Kony)

Dit platform is door Gartner bestempeld als een leider en heeft een gratis licentie. Wat serieuze overweging tegenhoud is de recente overname (en naamswijziging).

⁷https://nl.wikipedia.org/wiki/Customer_relationship_management

4. Conclusie

Curabitur nunc magna, posuere eget, venenatis eu, vehicula ac, velit. Aenean ornare, massa a accumsan pulvinar, quam lorem laoreet purus, eu sodales magna risus molestie lorem. Nunc erat velit, hendrerit quis, malesuada ut, aliquam vitae, wisi. Sed posuere. Suspendisse ipsum arcu, scelerisque nec, aliquam eu, molestie tincidunt, justo. Phasellus iaculis. Sed posuere lorem non ipsum. Pellentesque dapibus. Suspendisse quam libero, laoreet a, tincidunt eget, consequat at, est. Nullam ut lectus non enim consequat facilisis. Mauris leo. Quisque pede ligula, auctor vel, pellentesque vel, posuere id, turpis. Cras ipsum sem, cursus et, facilisis ut, tempus euismod, quam. Suspendisse tristique dolor eu orci. Mauris mattis. Aenean semper. Vivamus tortor magna, facilisis id, varius mattis, hendrerit in, justo. Integer purus.

Vivamus adipiscing. Curabitur imperdiet tempus turpis. Vivamus sapien dolor, congue venenatis, euismod eget, porta rhoncus, magna. Proin condimentum pretium enim. Fusce fringilla, libero et venenatis facilisis, eros enim cursus arcu, vitae facilisis odio augue vitae orci. Aliquam varius nibh ut odio. Sed condimentum condimentum nunc. Pellentesque eget massa. Pellentesque quis mauris. Donec ut ligula ac pede pulvinar lobortis. Pellentesque euismod. Class aptent taciti sociosqu ad litora torquent per conubia nostra, per inceptos hymenaeos. Praesent elit. Ut laoreet ornare est. Phasellus gravida vulputate nulla. Donec sit amet arcu ut sem tempor malesuada. Praesent hendrerit augue in urna. Proin enim ante, ornare vel, consequat ut, blandit in, justo. Donec felis elit, dignissim sed, sagittis ut, ullamcorper a, nulla. Aenean pharetra vulputate odio.

Quisque enim. Proin velit neque, tristique eu, eleifend eget, vestibulum nec, lacus. Vivamus odio. Duis odio urna, vehicula in, elementum aliquam, aliquet laoreet, tellus. Sed velit. Sed vel mi ac elit aliquet interdum. Etiam sapien neque, convallis et, aliquet vel, auctor non, arcu. Aliquam suscipit aliquam lectus. Proin tincidunt magna sed wisi. Integer blandit

lacus ut lorem. Sed luctus justo sed enim.

Morbi malesuada hendrerit dui. Nunc mauris leo, dapibus sit amet, vestibulum et, commodo id, est. Pellentesque purus. Pellentesque tristique, nunc ac pulvinar adipiscing, justo eros consequat lectus, sit amet posuere lectus neque vel augue. Cras consectetur libero ac eros. Ut eget massa. Fusce sit amet enim eleifend sem dictum auctor. In eget risus luctus wisi convallis pulvinar. Vivamus sapien risus, tempor in, viverra in, aliquet pellentesque, eros. Aliquam euismod libero a sem.

Nunc velit augue, scelerisque dignissim, lobortis et, aliquam in, risus. In eu eros. Vestibulum ante ipsum primis in faucibus orci luctus et ultrices posuere cubilia Curae; Curabitur vulputate elit viverra augue. Mauris fringilla, tortor sit amet malesuada mollis, sapien mi dapibus odio, ac imperdiet ligula enim eget nisl. Quisque vitae pede a pede aliquet suscipit. Phasellus tellus pede, viverra vestibulum, gravida id, laoreet in, justo. Cum sociis natoque penatibus et magnis dis parturient montes, nascetur ridiculus mus. Integer commodo luctus lectus. Mauris justo. Duis varius eros. Sed quam. Cras lacus eros, rutrum eget, varius quis, convallis iaculis, velit. Mauris imperdiet, metus at tristique venenatis, purus neque pellentesque mauris, a ultrices elit lacus nec tortor. Class aptent taciti sociosqu ad litora torquent per conubia nostra, per inceptos hymenaeos. Praesent malesuada. Nam lacus lectus, auctor sit amet, malesuada vel, elementum eget, metus. Duis neque pede, facilisis eget, egestas elementum, nonummy id, neque.

A. Onderzoeksvoorstel

Het onderwerp van deze bachelorproef is gebaseerd op een onderzoeksvoorstel dat vooraf werd beoordeeld door de promotor. Dat voorstel is opgenomen in deze bijlage.

A.1 Introductie

Het IT Team van AZ Glorieux heeft een oog op de toekomst. Twee grote voorbeelden hiervan zijn een push om nodige toestellen te migreren naar Windows 10 en een geleidelijke adoptie van Intune als aanvulling van System Center Configuration Manager (SCCM). Tijdens mijn stage heb ik hier een bescheiden bijdrage aan kunnen leveren. Het beoogde onderwerp voor mijn bachelor proef is verbonden met deze vernieuwing. LanReview, een binnenhuis ontwikkeld assetmanagement programma dat onontbeerlijk is voor de IT Helpdesk, is aan vernieuwing toe. De laatste ontwikkeling is van enige tijd geleden en het is niet logisch meer om dit terug op te nemen voornamelijk wegens de verouderde code base (Visual BASIC 3.0). De vervangende applicatie moet gemaakt worden met Microsoft Power Apps en ondersteunend ook Power Automate.

Meer achtergrond over de IT voorziening van AZ Glorieux verklaart deze keuze. Het is een groot regionaal ziekenhuis met een netwerk dat meer dan 1000 toestellen telt. Het domein is gebouwd met System Center Configuration Manager. Zowel Clients als Servers gebruiken vormen van het Microsoft Windows besturingsstelsel. Microsoft Office wordt op de meeste pc's gebruikt. Power Apps is ook opgenomen in het Office365 plan.

Een overzicht van de functies van LanReview die zeker ook in zijn opvolger aanwezig dienen te zijn:

- Overzicht geven van belangrijke informatie voor elke toestel in het domein.
- Rapporten kunnen genereren.
- Pc's bedienen vanop afstand.

Er zijn ook enkele volledig nieuwe functies beoogd:

- Mobiel bruikbaar zijn.

Overige functionaliteit:

- Grotere mate van Automatisatie introduceren met Power Automate.
- Buiten het domein bruikbaar zijn.
- Nieuwe types toestellen opnemen.
- Relevante Randapparatuur opnemen, deze in relatie stellen met de gekoppelde PC.
- Revisie van datavelden per toestel.
- Diverse Grafische User Interface (GUI) verbeteringen: kleuren gebruiken, tabbladen.
- Met barcode's kunnen werken. Bv de mac adres barcode inscannen om een nieuwe entry in de applicatie te maken.

Er zal een requirementsanalyse uitgevoerd worden via de MoSCoW-methode (Wikipedia, 2020). Dit staat verder beschreven in Methodologie.

Verdere uitleg van de hoofdfunctionaliteiten van LanReview via enkele use cases:

- Er komt een telefoon binnen van een dokter die een probleem heeft met één van de medische programma's. Het computernummer wordt doorgegeven. We zoeken op dit nummer in LanReview en krijgen een overzicht van de PC in kwestie. Het is nu mogelijk te rechtsklikken en de pc over te nemen om het probleem verder op te lossen.
- Er werden nieuwe laptops aangekocht. We nemen deze op in LanReview door op z'n minst een niet toegewezen computernummer te kiezen samen met het mac-adres. Aanvullende zaken zijn: een beschrijving, model, locatie, ip-nummer.
- Een model laptop dient geupdatet te worden naar Windows 10. Binnen LanReview queryen we op dit model en krijgen we een overzicht van elk toestel terug.

De hoofddoelstelling bestaat uit het opstellen van een competente Proof of Concept. Hiernaast moet verdere ontwikkeling zo toegankelijk mogelijk gemaakt worden. Met andere woorden de proef en bijbehorende documenten moeten een hoge mate van praktisch nut hebben voor het IT Team om niet gerealiseerde functionaliteit in de proof of concept en toekomstige functionaliteiten toe te kunnen voegen.

De onderzoeksvragen liggen in lijn met het bovenstaande:

- Is het mogelijk een vervanger voor LanReview te bouwen met Power Apps die elk de vier hoofdfunctionaliteiten ondersteunt en op z'n minst drie vierde van de overige functionaliteiten kan ondersteunen?
- Is Power Apps werkelijk de beste keuze hiervoor of is er meerwaarde in een volledig

gerealiseerd IT asset management pakket? Is er alternatief een beter geschikt low-code platform?

- Is de Proof of Concept eenvoudig uit te breiden met nieuwe functionaliteiten? Is dit aanvaardbaar voor het IT team van AZ Glorieux?
- Is Power Apps robuust genoeg om meer complexe functionaliteit te ondersteunen. Is het met een zelf geschreven uitbreiding bijvoorbeeld mogelijk om de remote desktop functionaliteit te verwezenlijken?
- Is de gebruikte methode op z'n minst deels bruikbaar om andere applicaties voor de IT van AZ Glorieux te bouwen? Er worden twee cases onderzocht: één voor een telefoonboek (legacy business applicatie met een lage moeilijkheidsgraad) en één om het potentieel van Power Apps te demonstreren. Dit is opgenomen in de volgende onderzoeksvraag.
- Is er een use case voor nieuwe of experimentele functionaliteiten in Power Apps zoals integratie met Teams of beperkt toepassen van AI via de AI Builder?
- De Proof of Concept zal nauw samenwerken met SCCM, is het concreet mogelijk om de nabije toekomst ook samen te werken met Intune?

A.2 State-of-the-art

LanReview is eigenlijk een view op SCCM. Het is uit die databank dat de gegevens per toestel komen. Deze gegevens werden aangevuld met extra data velden en opgeslaan in SharePoint lijsten. Om te rapporteren worden SQL achtige queries gebruikt op deze lijsten. Het is mogelijk om op elk veld van de entries te filteren. De data van elke entry kan ook aangepast worden. Er is een optie aanwezig om pc's over te nemen. Het onderliggende remote desktopprotocol wordt opgeroepen vanuit LanReview om dit te verwezenlijken.

De term assetmanagement werd gebruikt om LanReview te beschrijven maar waar LanReview een view is op SCCM kan aan traditioneel IT Asset Management Program meestal gezien worden als alternatief voor SCCM. In dergelijke programma's is buiten inventarisatie ook voorziening voor netwerk discovery, analytics, security en meer. Gartner (Gartner, 2020) beschrijft het als volgt: *IT asset management (ITAM) provides an accurate account of technology asset lifecycle costs and risks to maximize the business value of technology strategy, architecture, funding, contractual and sourcing decisions.*

Het IT asset management landschap is al enige tijd in evolutie door verspreiding van smartphone technologie en IoT (Badnakhe, 2020), dit maakt de keuze voor een nieuwe en flexibele technologie als Power Apps logisch.

PowerApps is een low code ontwikkelingsplatform dat niet programmeurs toelaat om business apps te maken. Het is mogelijk om canvas of modelgestuurde applicaties te bouwen (Knight, 2019). Indien iets niet visueel gedaan kan worden is gebruik van een Excel achtige query taal mogelijk (Owen, Lindhorst & Vivek, 2019). Uitbreiding is mogelijk via Connectoren naar externe services. Uitbreidingen zijn ook volledig zelf te bouwen in C Sharp (Vivek, Hecke, Guderian, Daly & Lattimer, 2019). Recente innovaties binnen het Power platform zijn introductie van AI mogelijkheden aan de hand van Virtual

Agents en een grotere integratie met Teams (Cunningham, 2019). Vooral dit laatste is interessant omdat AZ Glorieux recent ook Teams is beginnen introduceren.

Automatisatie in een PowerApp kan verwezelijkt worden via Power Automate, dat recent een naamwijziging heeft ondergaan van Flow (Weare, 2019) Volledig binnen de GUI is het mogelijk een automatisatieproces te maken bestaand uit aaneengeschakelde acties en condities.

Een populaire databron voor een PowerApp is SharePoint. Integratie gebeurt eenvoudigweg via de Connector ervoor. (Owen, Vivek, Manwaring, Blythe & Delano, 2019)

A.3 Methodologie

Er gaan twee onderzoekstechnieken gebruikt worden: een Proof of Concept voor de vervanger van LanReview gebouwd met PowerApps en een vergelijkende studie tussen beide applicaties. De vergelijking zal focussen op analyseren van requirements, bekijken hoe deze uitgewerkt kunnen worden en hierna ook illustrerend vergelijken met behulp van simulaties.

Er zal gewerkt worden in fasen.

1. De Basics: Een globaal beeld schetsen van de IT omgeving van AZ Glorieux en hierna uitdiepen wat LanReview nodig heeft om te kunnen werken (Specifiek de gebruikte databronnen). Achtergrondinformatie toereiken voor de gevonden technologieën en de technologieën die we zullen gebruiken om onze POC te bouwen (Power Apps, Power Automate).
2. LanReview Reviewen: Hiermee bedoelt het interne werken van LanReview in kaart brengen tot in de nodige details. Helaas zullen we hierbij niet op codeniveau geraken omdat de broncode niet beschikbaar is. Dit betekent ook dat geen code overgenomen kan worden naar de POC maar dat enige originaliteit nodig zal zijn bij het uitwerken van de functionaliteiten. Deze en de vorige stap zal ondersteund worden met afbeeldingen. Er zal in de eerste stappen gefocust worden op toegankelijkheid en duidelijkheid.
3. Requirementsanalyse: Met behulp van de MoSCoW-methode gaan er prioriteiten gesteld worden in het aanzienlijk aantal beoogde functionaliteiten. De basis hiervan werd reeds gelegd tijdens gesprekken specifiek hiervoor gehouden tijdens de Stageperiode met de co-promotor en collega's die de applicatie praktisch zouden gaan gebruiken.
4. Alternatieven: Hoewel er hoogstwaarschijnlijk niet afgeweken zal worden van PowerApps zal de nodige aandacht geven worden aan alternatieven. Welke volledig uitgewerkte proprietary softwarepakketten bestaan er die LanReview kunnen vervangen? Uit de vergelijking van Finances Online blijkt dat er keuze genoeg is (FinancesOnline, 2020). Is er serieuze concurrentie voor PowerApps vanuit het Open Source kamp? Binnnen PowerApps: is SharePoint de juiste technologie om onze

data op te slaan? De bedoeling hiervan is hoofdzakelijk inspiratie op te doen voor het uitwerken van de Proof of Concept. Als er echter uit dit onderzoek een superieure oplossing gevonden wordt zal dit voorgelegd worden aan de opdrachtgever.

5. Bouwen van de Proof of Concept: voor de praktische ontwikkeling van de POC laten we ons leiden door de voorganger en door de gebruikelijke technieken opgedaan in stap 1. De testfase zal uitvoerig zijn, de POC moet namelijk op termijn inzetbaar zijn binnen AZ Glorieux. Testen van code is niet aan de orde in een low-code ontwikkelplatform buiten de zelf geschreven uitbreidingen ervoor. Geautomatiseerd de UI testen is ook nog niet mogelijk doordat het Power Apps test Framework nog in de experimentele fase zit (Microsoft, 2020b). Tests zullen dus manueel opgesteld en uitgevoerd moeten worden. Indien toegelaten zou het ook behulpzaam zijn als de eindgebruikers bij tussenmomenten de Proof of Concept uitprobeerden en feedback gaven over hun ervaring.
6. Een vergelijking tussen LanReview met de POC. Wat zijn de gelijkenissen en verschillen? De uitwerking van belangrijke functionaliteiten zullen onderling vergeleken worden. In het bijzonder: Zijn er beperkingen in Power Apps gevonden waardoor ingeboet werd aan functionaliteit?
7. Aandacht voor de toekomst. De uitbreidingsmogelijkheden moeten gepeild worden. Extra functionaliteit moet zo vlot mogelijk geïntroduceerd kunnen worden door het IT Team. Is het mogelijk PowerApps elders toe te passen in AZ Glorieux via een soortgelijke methode als voor onze POC. Mogelijk wordt dit uitgelegd via een specifieke case. Hoe moet de POC aangepast worden om op termijn samen te kunnen werken met Intune zoals het nu nauw met SCCM samenwerkt? In dit gedeelte of één apart ervoor zal ook een praktisch aspect aanwezig moeten zijn voor het IT Team van AZ Glorieux dat als documentatie zal moeten dienen of alternatief, verwijzingen moet hebben naar bestaande documentatie.

Praktischere zaken:

Wat tools betreft moet specifiek een PowerApps ontwikkelomgeving opgezet worden. Dit kan via de webapplicatie voor PowerApps als deel van het Office365 pakket. Alternatief zou het ook mogelijk moeten zijn om het softwarepakket "PowerApps Studio" te installeren. Als het maken van klassediagrams niet uitgebreid mogelijk zou zijn vanuit PowerApps zal Visual Paradigm gebruikt worden.

Over AZ Glorieux, een belangrijke noot:

Als tijdens het uitwerken van de proef het AZ Glorieux netwerk of iets ermee te maken nodig zou zijn werd reeds voorgesteld dat ik terplaatse mag werken.

A.4 Verwachte resultaten

Tastbare resultaten:

- Een comprehensieve requirementsanalyse. Als hier niet de nodige aandacht wordt gegeven bestaat het risico van de "verkeerde" POC te bouwen. Een goede basis is

belangrijk dus er wordt verwacht dat de requirementsanalyse uitgebreid is.

- Logischerwijze zullen niet alle requirements opgenomen kunnen worden in de POC, toch wordt verwacht dat de belangrijkste uitgewerkt zijn. Ruimer gezien wordt verwacht dat de POC met weinig aanpassing de originele applicatie, LanReview, kan gaan vervangen.
- Een bijkomend resultaat is een set instructies of een stappenplan, de vorm kan nog wijzigen maar er zal een document gebouwd worden dat het IT Team kan gebruiken om verder te werken met PowerApps, om ontwikkeling van de POC over te kunnen nemen.

Met betrekking tot de POC:

Er wordt verwacht dat het opzetten van een PowerApps applicatie weinig problemen geeft, dit is uiteindelijk het doel van low code platformen zoals PowerApps. Er wordt wel enige uitdaging verwacht bij het uitwerken van functionaliteit die niet direct voorzien is. Uitbreidingen schrijven, vooral als het op automatisatie aankomt, kan zich moeilijk tonen. Er moet rekening mee gehouden worden in de planning.

Er wordt overigens verwacht dat er niet afgeweken zal worden van een combinatie van PowerApps, Power Automate en SharePoint voor het bouwen van de POC.

Indien dit gemeten wordt zal er vermoedelijk geen significant verschil zijn in performantie tussen de POC en LanReview.

A.5 Verwachte conclusies

De conclusies, geformuleerd als antwoorden op de onderzoeksvragen, zijn als volgt:

- Er wordt verwacht dat een vervanger voor LanReview gebouwd kan worden met LanReview. SharePoint kan gebruikt blijven als data bron maar SCCM (dus SQL Server) verbinding is mogelijk beperkt door licentie wijzigingen. (Microsoft, 2020a)
- Power Apps is de beste keuze in ons scenario. Zelf als hiervoor betere software gevonden werd zal de bijkomende kostprijs niet verantwoord kunnen worden.
- Het is mogelijk de POC af te leveren op een manier waarmee verder ontwikkeling ondersteund wordt.
- PowerApps is robuust genoeg om de hoofdfunctionaliteiten te ondersteunen, al dan niet met zelfgemaakte uitbreidingen.
- Er wordt verwacht dat de gebruikte techniek ook andere toepassingen kan vinden binnen AZ Glorieux.
- Een aanpassing naar Intune in plaats van of in combinatie met SCCM is doenbaar. De nodige tijds en moeite investering is aanvaardbaar.

Samenvattend wordt verwacht dat de POC een meerwaarde is voor AZ Glorieux en dat deze kan groeien tot waardige opvolger voor LanReview.

Bibliografie

- Aho, L. (2018). *Mobile data collection and storing solution for Microsoft environments*.
- Badnakhe, R. (2020, januari 23). Why Are More Businesses Adopting IoT for Asset Management? Verkregen 6 april 2020, van <https://dzone.com/articles/why-more-businesses-are-adopting-iot-for-asset-man>
- Bashar, L. (2017, juni 28). Introducing Sapphire, the Hospital Management System Built with OutSystems. Verkregen 6 april 2020, van <https://www.outsystems.com/blog/introducing-sapphire-hospital-management-system.html>
- Bloomberg, J. (2017, juni 22). Don't Confuse Low-Code with No-Code. Verkregen 6 april 2020, van <https://intellyx.com/2017/06/22/dont-confuse-low-code-with-no-code/>
- Brust, D. (2015, september 14). Installing Visual Basic/Studio 6 on Windows 10. Verkregen 6 april 2020, van <http://blog.danbrust.net/2015/09/14/installing-visual-basic-studio-6-on-windows-10/>
- classicvb petition. (2005, maart 8). A PETITION FOR THE DEVELOPMENT OF UNMANAGED VISUAL BASIC AND VISUAL BASIC FOR APPLICATIONS. Verkregen 6 april 2020, van <https://web.archive.org/web/20141024003217/http://www.classicvb.org/petition/>
- Cloudflare. (g.d.). What Is Vendor Lock-In? | Vendor Lock-In and Cloud Computing. Verkregen 18 april 2020, van <https://www.cloudflare.com/learning/cloud/what-is-vendor-lock-in/>
- Cunningham, R. (2019, november 4). Power Apps Top 10: innovation areas that are reshaping app development. Verkregen 14 februari 2020, van <https://powerapps.microsoft.com/en-us/blog/power-apps-top-10-innovation-areas-that-are-reshaping-app-development/>
- Devloo, M. (2018). *Kan een low-code development-platform een substantiële meerwaarde betekenen voor kleine en middelgrote bedrijven in de IT-sector?*

- Droogenbroot, W. & Scheiris, J. (2016, april 11). Efficiënt beheer van uw PC-park met Microsoft System Center Configuration Manager. Verkregen 6 april 2020, van <https://www.savaco.com/nl/nieuws/detail/efficient-beheer-van-uw-pc-park-met-microsoft-system-center-configuration-manager>
- FinancesOnline. (2020, februari 27). Top 12 Most Popular IT Asset Management Software. Verkregen 2 maart 2020, van <https://financesonline.com/it-asset-management/>
- Gartner. (g.d.). It Asset Management (itam). Verkregen 6 april 2020, van <https://www.gartner.com/en/information-technology/glossary/it-asset-management-itam>
- Gartner. (2020). It Asset Management (itam). Verkregen 14 februari 2020, van <https://www.gartner.com/en/information-technology/glossary/it-asset-management-itam>
- Grigonis, R. (2014, maart 27). Where Is the Successor to Visual Basic? Verkregen 6 april 2020, van <https://www.newsmax.com/RichardGrigonis/Visual-Basic-Microsoft-programmers/2014/03/27/id/562161/>
- IAITAM. (g.d.). What is IT Asset Management (ITAM)? Verkregen 6 april 2020, van <https://iaitam.org/what-is-it-asset-management/?q=what-is-it-asset-management/&v=d3dcf429c679>
- Ippolito, I. (2018, september 20). VB6 Desktop Icon Size Changer. Verkregen 6 april 2020, van <http://www.planetsourcecode.com/vb/scripts/ShowCode.asp?txtCodeId=75311&lngWId=1>
- ISplitter. (2014, januari 14). Visual Basic 6.0: A giant more powerful than ever. Verkregen 6 april 2020, van <https://www.codeproject.com/Articles/710181/Visual-Basic-6-0-A-giant-more-powerful-than-ever>
- Ivanti. (2018, oktober 10). What Is IT Asset Management? Verkregen 6 april 2020, van <https://www.ivanti.com/blog/what-is-it-asset-management>
- Jugo, A. (2019, december 1). Extending PowerApps and Flow, part 2: Extending PowerApps language with Azure API Apps. Verkregen 8 april 2020, van <https://blog.sharedove.com/adisjugo/index.php/2018/12/01/extending-powerapps-and-flow-part-2-extending-powerapps-expression-language-with-azure-api-apps/>
- Kissflow. (2018a, april 17). Low-Code Development Platforms – Launchpads to Build Powerful Apps Easily. Verkregen 6 april 2020, van <https://kissflow.com/rad/low-code/low-code-development-platform-launchpad-build-apps-easily/>
- Kissflow. (2018b, mei 2). The History of Low-Code Platforms: How Development Changed Forever. Verkregen 6 april 2020, van <https://kissflow.com/rad/low-code/history-of-low-code-development-platforms/>
- Knight, B. (2019, mei 14). PowerApps Canvas vs Model-driven Applications. Verkregen 14 februari 2020, van <https://blog.pragmaticworks.com/powerapps-canvas-vs-model-driven-applications>
- McKendrick, J. (2017). *THE RISE OF THE EMPOWERED CITIZEN DEVELOPER - 2017 LOW-CODE ADOPTION SURVEY*. Unisphere Research.
- Meganathan, A. (2019, juli 26). Build Asset Checkout App Using Microsoft PowerApps. Verkregen 8 april 2020, van <https://www.c-sharpcorner.com/article/build-asset-checkout-app-using-microsoft-powerapps/>
- Microsoft. (2020a, januari 7). Power Apps and Power Automate licensing FAQs. Verkregen 2 maart 2020, van <https://docs.microsoft.com/en-us/power-platform/admin/powerapps-flow-licensing-faq>

- Microsoft. (2020b, januari 10). Power Apps test framework. Verkregen 2 maart 2020, van <https://docs.microsoft.com/en-us/power-platform-release-plan/2019wave2/microsoft-powerapps/test-framework>
- Microsoft Docs. (2017, juni 12). Create a canvas app to manage projects. Verkregen 8 april 2020, van <https://docs.microsoft.com/en-us/powerapps/maker/canvas-apps/sharepoint-scenario-build-app>
- Microsoft Docs. (2018, januari 18). Support Statement for Visual Basic 6.0 on Windows. Verkregen 6 april 2020, van <https://docs.microsoft.com/en-us/previous-versions/visualstudio/visual-basic-6/visual-basic-6-support-policy>
- Microsoft Docs. (2019a, januari 3). Aan de slag met het canvas-app-formules in Power apps. Verkregen 8 april 2020, van <https://docs.microsoft.com/nl-nl/powerapps/maker/canvas-apps/working-with-formulas>
- Microsoft Docs. (2019b, november 5). Get started with Power Automate. Verkregen 8 april 2020, van <https://docs.microsoft.com/en-us/power-automate/getting-started>
- Microsoft Docs. (2019c, november 18). Test Studio (experimental). Verkregen 8 april 2020, van <https://docs.microsoft.com/en-us/powerapps/maker/canvas-apps/test-studio>
- Microsoft Docs. (2019d, juni 12). Wat is Common Data Service? Verkregen 8 april 2020, van <https://docs.microsoft.com/nl-nl/powerapps/maker/common-data-service/data-platform-intro>
- Microsoft Docs. (2019e, september 6). What is AI Builder? Verkregen 8 april 2020, van <https://docs.microsoft.com/en-us/ai-builder/overview>
- Microsoft Docs. (2019f, oktober 16). What is an on-premises data gateway? Verkregen 8 april 2020, van <https://docs.microsoft.com/en-us/powerapps/maker/canvas-apps/gateway-reference>
- Microsoft Docs. (2019g, juli 15). What is Power Apps? Verkregen 8 april 2020, van <https://docs.microsoft.com/en-us/powerapps/powerapps-overview>
- Microsoft Docs. (2020a, februari 12). Frequently asked questions for Configuration Manager branches and licensing. Verkregen 6 april 2020, van <https://docs.microsoft.com/en-us/configmgr/core/understand/product-and-licensing-faq>
- Microsoft Docs. (2020b, maart 19). Overview of canvas-app connectors for Power Apps. Verkregen 8 april 2020, van <https://docs.microsoft.com/en-us/powerapps/maker/canvas-apps/connections-list>
- Microsoft Docs. (2020c, januari 17). What is Power Apps portals? Verkregen 8 april 2020, van <https://docs.microsoft.com/en-us/powerapps/maker/portals/overview>
- Microsoft Power Automate. (2019, november 4). Learn about robotic process automation in Power Automate. Verkregen 12 april 2020, van https://www.youtube.com/watch?v=bZrrdoytTH0&feature=emb_title
- Microsoft Support. (2020, januari 15). Ondersteuning voor Windows 7 beëindigd op 14 januari 2020. Verkregen 12 april 2020, van <https://support.microsoft.com/nl-be/help/4057281/windows-7-support-ended-on-january-14-2020>
- Ness, C. & Hansen, M. E. (2019). *Potential of low-code in the healthcare sector* (master-scriptie, Norwegian School of Economics).
- Nucleus. (2017, januari 16). Wat is het verschil tussen IaaS, PaaS, SaaS en UaaS? Verkregen 6 april 2020, van <https://www.nucleus.be/blog/uptime-as-a-service/verschil-iaas-paas-saas-en-uas/>

- Owen, A. F., Lindhorst, G. & Vivek, K. (2019, december 19). Formula reference for Power Apps. Verkregen 14 februari 2020, van <https://docs.microsoft.com/en-us/powerapps/maker/canvas-apps/formula-reference>
- Owen, A. F., Vivek, K., Manwaring, M., Blythe, M. & Delano, L. (2019, maart 4). Connect to SharePoint from a canvas app. Verkregen 14 februari 2020, van <https://docs.microsoft.com/en-us/powerapps/maker/canvas-apps/connections/connection-sharepoint-online>
- Pohl, T. (2019, september 27). Overview of the new pricing models for Microsoft PowerApps and Flow. Verkregen 8 april 2020, van <https://blog.atwork.at/post/Microsoft-PowerApps-and-Flow-licensing>
- Polleffiet, L. (2011). *Schrijven van verslag tot eindwerk: do's en don'ts*. Gent: Academia Press.
- Pragmatic Works. (2019, april 3). PowerApps Canvas vs Model-Driven Applications. Verkregen 8 april 2020, van https://www.youtube.com/watch?v=fhIIC1mSO_4&feature=emb_title
- Reselman, B. (2018, januari 31). Why the promise of low-code software platforms is deceiving. Verkregen 6 april 2020, van <https://searchsoftwarequality.techtarget.com/opinion/Why-the-promise-of-low-code-software-platforms-is-deceiving>
- Revell, M. (2020, januari 16). What Is Low-Code? [2020 Update]. Verkregen 6 april 2020, van <https://www.outsystems.com/blog/what-is-low-code.html>
- Richardson, C. & Rymer, J. R. (2016, januari 15). *Vendor Landscape: The Fractured, Fertile Terrain Of Low-Code Application Platforms*. Forrester.
- Rouse, M. (2019, juli). Visual Basic (VB). Verkregen 6 april 2020, van <https://searchwindevelopment.techtarget.com/definition/Visual-Basic>
- Rymer, J. (2018, augustus 8). Why You Need To Know About Low-Code, Even If You're Not Responsible For Software Delivery. Verkregen 12 april 2020, van <https://go.forrester.com/blogs/why-you-need-to-know-about-low-code-even-if-youre-not-responsible-for-software-delivery/>
- Rymer, J. R. & Koplowitz, R. (2019, maart 13). *The Forrester Wave: Low-Code Development Platforms For ADD Professionals, Q1 2019*. Forrester.
- Salesforce. (g.d.). What Is Digital Transformation? Verkregen 6 april 2020, van <https://www.salesforce.com/products/platform/what-is-digital-transformation/>
- Shiah, J. (2018, april 27). Why "low-code development" isn't always truly "low-code". Verkregen 6 april 2020, van <https://www.itproportal.com/features/why-low-code-development-isnt-always-truly-low-code/>
- Speed, R. (2020, maart 12). Microsoft throws a bone to those unable to leave the past behind: .NET 5 support on the way for Visual Basic. Verkregen maart 2020, van https://www.theregister.co.uk/2020/03/12/visual_basic_net_5/
- Spitze, P. (2019, oktober 7). Microsoft SCCM and the alternatives in 2019. Verkregen 6 april 2020, van <https://www.software2.com/resource-centre/microsoft-sccm-and-the-alternatives-in-2019>
- Spriet, J. (2019). *Hoe kunnen de administratieve systemen van verhuizingen De Borgerge-optimaliseerd en efficiëntergemaakt worden?*
- Totterdale, R. L. (2018). CASE STUDY: THE UTILIZATION OF LOW-CODE DEVELOPMENT TECHNOLOGY TO SUPPORT RESEARCH DATA COLLECTION. *Issues in Information Systems*, 19(2), 132–139.

- van Schetsen, A. (2016, mei). Guest View: Five reasons low-code development is no longer optional. Verkregen 6 april 2020, van <https://sdtimes.com/agile/guest-view-five-reasons-low-code-development-no-longer-optional/>
- Vincent, P., Iijima, K., Driver, M. & Natis, Y. (2019, augustus 8). *Magic Quadrant for Enterprise Low-Code Application Platforms*. Gartner.
- Virta, T. (2018). *Relation of low-code development to standard software development: Case Biit Oy* (masterscriptie, Lappeenranta University of Technology).
- Vivek, K., Hecke, P., Guderian, B., Daly, J. & Lattimer, J. (2019, maart 27). Use plug-ins to extend business processes. Verkregen 14 februari 2020, van <https://docs.microsoft.com/en-us/powerapps/developer/common-data-service/plugin-ins>
- Weare, K. (2019, december 23). Microsoft Rebrands Flow Service to Power Automate, Adds Robotic Process Automation Capabilities. Verkregen 14 februari 2020, van <https://www.infoq.com/news/2019/12/Power-Automate-RPA/>
- Wikipedia. (2020, maart 1). MoSCoW-methode. Verkregen 2 maart 2020, van <https://nl.wikipedia.org/wiki/MoSCoW-methode>