



Vijf tips om succesvol te innoveren met low-code

Whitepaper

@blisdigital



Unleash
the power of
technology

Blis
Digital



Inhoudsopgave

- 5 Wat is low-code?
- 7 Waarom kiezen organisaties low-code?
- 8 Voordelen van low-code
- 9 Microsoft Power Platform
- 10 Onderdelen van het Power Platform
- 11 Tip 1 tot en met 5
- 25 Resources



Waarom wij dit hebben geschreven voor jou...

Low-code is hot. En het Microsoft Power Platform is misschien nog wel hotter. Met het Power Platform levert Microsoft een hele interessante oplossing voor organisaties die al met Microsoft 365-applicaties werken en die digitaal willen innoveren.

Hoewel het Power Platform je in staat stelt om snel en eenvoudig oplossingen te bouwen, zijn de oplossingen en het beheer allesbehalve eenvoudig en er zijn een aantal belangrijke randzaken waar je goed over na moet denken, wil je het platform écht succesvol kunnen inzetten. Als grote Power Platform fans én ervaren software developers, combineren wij

graag de laagdrempeligheid en pragmatiek van low-code, met de principes en best practices van professioneel software development.

In dit whitepaper hebben wij de vijf belangrijkste principes hiervan beschreven en willen wij je een waardevol hulpmiddel bieden, zodat je beslagen ten ijs komt wanneer je met het Power Platform aan de slag gaat. Veel leesplezier!

Blijft inmiddels

Wat is low-code technologie?

Het Power Platform is het low-code platform van Microsoft. Voor we specifiek inzoomen op het Power Platform, eerst een paar woorden over low-code in het algemeen. Wat is het en waarom zou je het gebruiken?

Focus op de oplossing

No-code is het bouwen van applicaties in een grafische interface, zonder dat daar enige code bij komt kijken. Met low-code doelen we op een manier van software bouwen die ook grotendeels grafisch is, maar die daarnaast de ruimte geeft om complexe logica vorm te geven in makkelijk te begrijpen scripts en formules. Dat maakt low-code ontwikkeling een relatief eenvoudige manier om apps en oplossingen te bouwen voor zakelijke uitdagingen. Het zorgt dat je snel kunt ontwikkelen, het biedt een manier om bouwstenen te gebruiken (vergelijkbaar met Lego) en is gebaseerd op een datamodel. Lowcode ontwikkeling is een vorm van 'declarative programming'.



Unleash
the power of
technology

De tegenhanger van low-code noemen we ook wel 'high-code'. High-code is een imperatieve vorm, waarbij als het ware turn-by-turn instructies aan de computer worden gegeven met behulp van code.

Citizen developers

Low-code-platformen zijn vooral een uitkomst voor zogenaamde citizen developers. Citizen developers zijn gebruikers uit de business met IT-affiniteit. Zij zijn 'officieel' geen software developer, maar kunnen dankzij low-codeplatformen toch zelf oplossingen bouwen, zonder software developers te hoeven inschakelen. Low-code gebruikt eenvoudige expressies voor het maken van configuraties en formules, vergelijkbaar met good old Excel en oude bekende Microsoft Access.



Waarom kiezen organisaties voor low-code?

Het is dus relatief eenvoudig om apps en toepassingen te bouwen met behulp van lowcode. Maar dat is zeker niet de enige reden waarom low-code technologie zo populair is. Organisaties kiezen doorgaans om de volgende twee redenen om low-code technologie te omarmen.

Innovatiebehoefte

De behoefte aan flexibele IT-platformen en applicaties neemt de laatste jaren enorm toe. Met de klassieke manier van software ontwikkeling, gefocust op controle, stabiliteit en veiligheid, kun je niet altijd meebewegen met de behoefte van de business om snel te innoveren. Juist daar is low-code een uitkomst. Of het nu gaat om het automatiseren businessprocessen, het lanceren van digitale innovaties of het moderniseren en verSaaS-en van bestaande software: low-code is een interessant alternatief of juist een aanvulling op bestaande (legacy) bedrijfsapplicaties of ERP-systemen.

Krapte op arbeidsmarkt

Low-code draagt ook bij aan een oplossing voor de enorme krapte op de arbeidsmarkt in het algemeen en die van software developer in het bijzonder. Het opleiden van een citizen developer vraagt minder tijd dan een full-stack software developer. Daarnaast kun je deze twee rollen mooi samen laten werken; de citizen developer doet de eenvoudiger 'standaard' dingen, terwijl de full-stack developer zich met de technisch complexere zaken bezig kan houden. Ook voor arbeidsproductiviteit biedt lowcode een uitkomst. Door het werk meer te automatiseren met 'slimme' oplossingen kan een medewerker meer doen, in minder tijd.



De voordelen van low-code:

1. **Minder code** dus minder bugs en een kortere ontwikkeltijd.
2. **Nauwelijks professionele** ontwikkelingsvaardigheden vereist.
3. **Je apps draaien op het low-code platform**, een gecontroleerde omgeving waarbinnen functionaliteiten aanwezig zijn om zaken als governance, security en monitoring in te regelen.
4. **Het is eenvoudig** om te integreren met legacy-systemen en bestaande databronnen.
5. **Standaard krijg je een geweldige gebruikersinterface** en je apps zijn te direct te gebruiken op verschillende devices, zoals PC, tablet en telefoon.

Power Platform, het low-code platform binnen Microsoft 365

Verschillende low-code platformen

Betty Blocks, Appian, Mendix, OutSystems en Microsoft hebben allemaal hun eigen mogelijkheden en beperkingen. Wij zijn groot fan van het Power Platform, het low-code-platform van Microsoft, dat beschikbaar is als een cloud-only offering en draait op Microsoft Azure.

Integratie met Microsoft 365

Het Power Platform is een interessante aanvulling op de bestaande Microsoft 365-applicaties die veel organisaties inzetten, zoals Word, Excel, SharePoint en Teams. Dit platform maakt het mogelijk om eenvoudig en zonder al te veel technische kennis, apps te bouwen, bedrijfsprocessen te automatiseren en veel voorkomende handmatige taken te digitaliseren. Je bepaalt als organisatie helemaal zelf het tempo en de schaal

van je groeipad. Met het Power Platform maak je net zo makkelijk een eenvoudige goedkeuringsflow voor declaraties als een geavanceerde virtuele assistent voor de servicedesk.

Van eenvoudig dashboard tot mission critical app

Of het nu gaat om een eenvoudig dashboard voor financiële inzichten of een bedrijfskritische toepassing, met het Power Platform kan het allemaal. Het is zelfs mogelijk om ergens tijdens het bewandelen van jouw groeipad, te besluiten een applicatie alsnog om te zetten van Power Platform low-code naar een full stack-app, die door professionele ontwikkelaars wordt doorontwikkeld.

Unleash
the power of
technology

De vier onderdelen van het Power Platform

Het Power Platform is een standaard onderdeel van de meeste Microsoft 365-licenties en integreert naadloos met applicaties als Teams, Dynamics 365, SharePoint en Excel. Het Power Platform bestaat uit vier aparte toepassingen:

1. **Power BI** voor het maken van rapportages, interactieve dashboards en data-analyses
2. **Power Apps** voor het maken van apps en formulieren
3. **Power Automate** voor het automatiseren van processen
4. **Power Virtual Agents** voor het maken van chatbots

Naast deze toepassingen maakt Power Platform op de achtergrond gebruik van verschillende Azure-diensten voor data-connectoren, authenticatie, logging en AI-functies.

Elk van de Power Platform-toepassingen heeft een andere invalshoek, maar allemaal zijn ze gebaseerd op het low-code-principe: je kunt er eenvoudig en zonder veel technische kennis mee beginnen.

Je kunt de toepassingen onderling verbinden, maar ook met data uit Microsoft 365, Dynamics 365 en Azure of niet-Microsoft bronnen, door gebruik te maken van data-connectoren.

Bliss

inmiddels

Tip 1: meer low-code dan je denkt!

Forrester introduceerde de term low-code in 2014 om het nieuwe denken over appontwikkeling aan te duiden. Maar wie denkt dat low-code iets heel nieuws is, komt bedrogen uit.

Ook dit is low-code

Sinds 1993 is het al mogelijk om in Excel macro's met behulp van Visual Basic for Applications (VBA) te programmeren. En wat denk je van al die slimme oplossingen die te vinden zijn in Access databases, SharePointlijsten en -workflows en InfoPath-forms. Wat ons betreft zijn ook dat allemaal vormen van low-code.

Meer bedrijfskritisch dan je denkt

Deze vorm van low-code is voor heel veel gebruikers uit de business al jaren een feestje, maar voor veel bedrijven is het vandaag de dag een enorm pijnpunt. Op deze manier zijn veel functionaliteiten gecreëerd waar de business inmiddels op vertrouwt, maar die niet door IT-professionals beheerd worden, waarvan niemand meer weet wie ze heeft bedacht, hoe ze zijn gemaakt en waar zijn gedocumenteerd.



Unleash
the power of
technology

Over lifecyclemanagement van deze functionaliteiten is nooit nagedacht, maar als iets niet meer werkt, dan is het ineens wel het probleem van de IT-afdeling. Kortom, hier zul je dus iets mee moeten.

Het begint met inzicht

Het inzichtelijk maken van wat er al in je omgeving is, is een belangrijke eerste stap. Nu je weet welke low-code oplossingen er allemaal bestaan, kun je per oplossing bepalen wat er nodig is om weer 'in control' te komen. En daar veel van de eerder genoemde low-code vormen inmiddels verouderd zijn, is het waarschijnlijk een goed idee om deze oplossingen op nieuw te realiseren met behulp van het Power Platform.



Tip 2: Ook voor low-code moet je ALM inrichten

Net als elke andere IT-oplossing, moeten ook low-code-oplossingen beheerd worden. Je wilt immers grip en controle houden over de oplossingen in jouw organisatie. Bovendien wil je kwaliteit en beschikbaarheid van de oplossingen borgen en bedrijfskritische data veilighouden. Door het Power Platform als volwaardig onderdeel op te nemen in je IT-strategie, voorkom je ongewenste verrassingen. Application Lifecycle Management (ALM) is een (deel van het) antwoord op deze zoektocht.

Wat is ALM?

Application Lifecycle Management (ALM) gaat over de combinatie van mens, beleid en technologie die nodig is om een softwareoplossing gedurende de hele levenscyclus te beheersen, van concept tot en met uitfasering. Er zijn verschillende modellen om ALM in te richten, maar één ding hebben ze allemaal gemeen, ze zijn er opgericht om de complexe stappen uit softwareontwikkeling te vangen in een eenduidig en herhaalbaar proces. Daarbij wordt dus niet alleen rekening gehouden met de techniek, maar ook hoe je als organisatie om gaat met het beheer en het gebruik van de applicatie.

Ook voor oplossingen die gemaakt worden met het Power Platform zijn ALM-principes essentieel. Immers, je wil het Power Platform duurzaam inzetten.

ALM binnen het Power Platform

Als je Power Platform-oplossingen duurzaam in wilt zetten, is het dus belangrijk de ALM-principes te hanteren. ALM kent een aantal verschillende fases, elk met hun eigen kenmerken. Door deze fases te volgen, ben je in staat de juiste keuzes te maken en weet je altijd waar jouw oplossing is qua levensduur. Laten we elke fase eens nader bekijken.

Plan

Voor je begint een oplossing te maken met het Power Platform, is het van belang om goed na te denken over de licenties die je nodig hebt. Binnen het Power Platform zijn diverse licentie(s) (modellen) beschikbaar waarvan de kosten flink uitéén kunnen lopen.

Om ongewenste verrassingen te voorkomen, begin je met het in kaart brengen van de benodigde functionaliteiten en het mappen hiervan op de verschillende licentieopties. De [Power Platform Licensing Overview](#) is hierbij een handig hulpmiddel om te bepalen welke componenten uit het Power Platform het meest geschikt zijn, eventueel aangevuld met diensten uit Azure. Als onderdeel hiervan moet je ook nadenken over het gebruik van de oplossing. Hoeveel data wordt er opgeslagen? Hoeveel API's ga je aanroepen? Allemaal zaken die effect hebben op de kosten.

Develop

Als je eenmaal gaat bouwen, hou je uiteraard rekening met de naamgevingsconventies en maak je keuzes over hoe je straks de toegang tot de applicatie regelt. Het ontwikkelen zelf gebeurt in een iteratief proces, dat je samen met de eindgebruikers doorloopt, en hangt samen met het testen en uitrollen van je

oplossing. Denk hierbij ook aan het inzetten van verschillende Power Platform-omgevingen, om een deployment-strategie te ondersteunen met gescheiden ontwikkel-, test-, acceptatie- en productie-omgevingen (OTAP). Maar ook om te voldoen aan specifieke data-beveiligingseisen kan het gebruik van verschillende omgevingen een uitkomst zijn.

Test

Op het testen van low-code-applicaties komen we later terug, maar dat ze getest moeten worden, is een ding dat zeker is. Belangrijk is om alle drie de testfasen te doorlopen. Requirements worden vaak in de Plan-fase opgesteld. Al in die fase is het belangrijk een softwaretester aan te haken om vroegtijdig eventuele onduidelijkheden of tegenstrijdigheden in die requirements te identificeren. Vervolgens is het de taak van een softwaretester om integraties en functionaliteit van het proces of oplossing te testen. Ook helpt de softwaretester bij de gebruikerstest. Als deze stappen succesvol zijn doorlopen kun je de oplossing met een gerust hart in gebruik nemen.

Release

Het uitrollen van low-code-oplossingen gebeurt vaak door mensen direct toegang te geven tot een app. Op die manier is het niet meer mogelijk in een later stadium eenvoudig aanpassingen

door te voeren – én te testen – zonder de gebruikers hiermee lastig te vallen. Elke aanpassing is namelijk direct ‘beschikbaar’ voor je gebruikers. Daarom is het altijd verstandig om minimaal een gescheiden test- en een productieversie van een oplossing uit te rollen.

Het Power Platform biedt de mogelijkheid om met verschillende omgevingen logische scheidingen aan te brengen. Met behulp van Power Platform Solutions kun je je oplossing eenvoudig bundelen en uitrollen een andere omgeving. En met behulp van Power Platform Build Tools for Azure DevOps kun je er zelfs voor zorgen dat zaken automatisch uitgerold worden middels een zogenaamde ‘deployment pipeline’.

Operate

Zodra een oplossing eenmaal operationeel is, wil je natuurlijk weten hoe het staat met het gebruik en de status van die oplossing. In de Operatefase gaat het dus vooral om inzicht. Wat gebeurt er en waar wordt de oplossing voor ingezet? Doet het wat het moet doen? Zijn er issues of neemt het gebruik exponentieel toe en moet je daarop gaan anticiperen?

De Power Platform Center of Excellence Starter Kit geeft je inzicht en informatie over veel van deze componenten. Het Power BI-dashboard maakt in één oogopslag alle Power Platform-

oplossingen in jouw tenant inzichtelijk. Ook die oplossingen die buiten jou om zijn gebouwd.

Learn

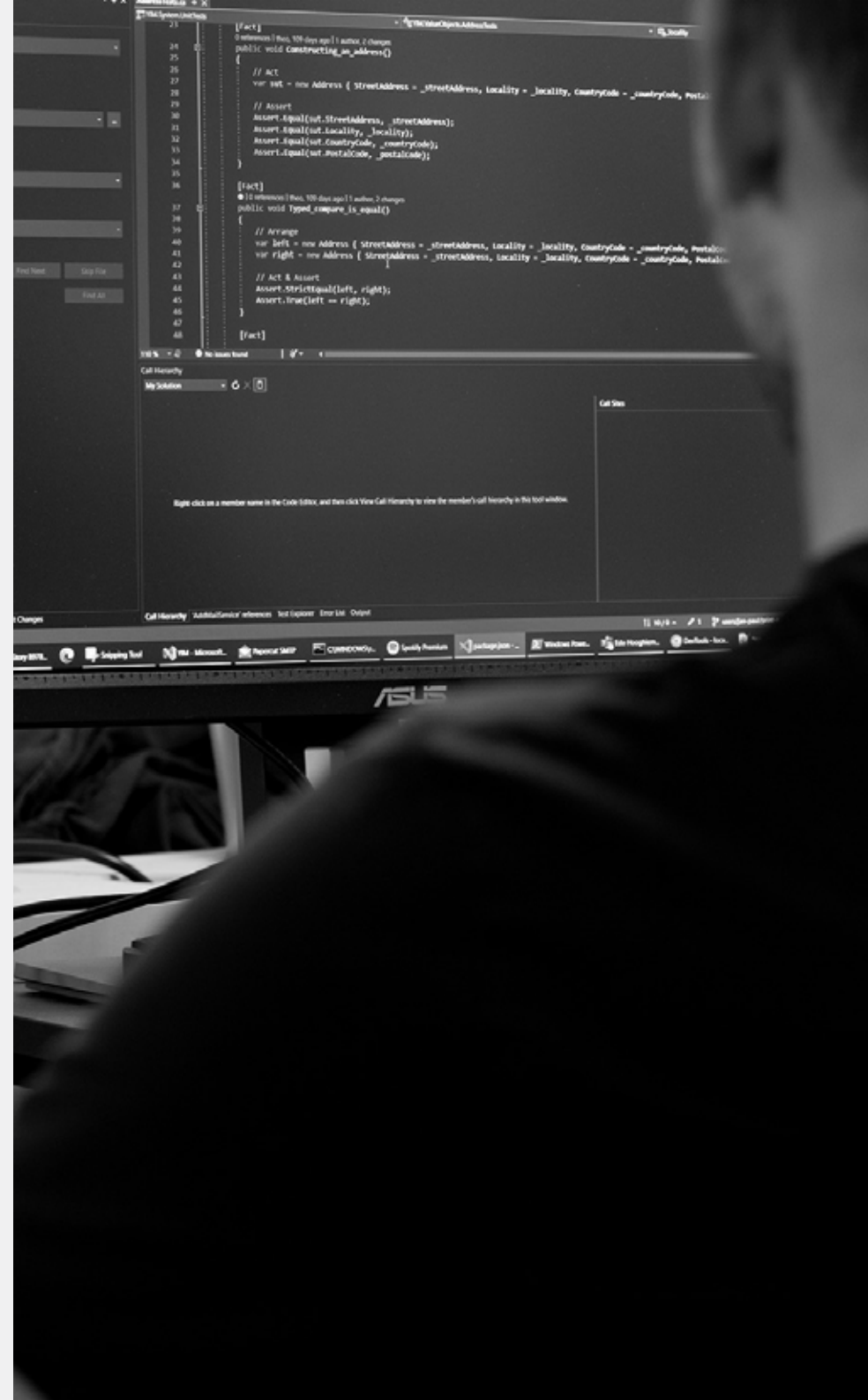
De inzichten en informatie die je uit de Operate-fase haalt, gecombineerd met gebruiksrapportages van de verschillende oplossingen in jouw omgeving, helpen je overzicht te houden. Zo kom je van reactief beheer (break/fix) naar proactief beheer (in control) om problemen voor te zijn en verstoringen te voorkomen. In samenspraak met de gebruikers ontwikkel je oplossingen continue verder en ben je in staat om steeds weer in te spelen op alle nieuwe vragen uit de business.

Van onbeheerd naar beheerd

Veel oplossingen in het Power Platform beginnen als een work-around van een bepaald proces of handmatige taken. Vaak zonder verdere gevolgen of keuzes rondom een formeler proces. Wil je echter het Power Platform inzetten zonder dat de omgeving vervuild raakt, of dat je gevoelige data verliest door onzorgvuldig gebruik, dan moet je nadenken over het beheren van die oplossingen. Alles open laten staan is geen optie, dat werkt chaos in de hand. Maar alles dichtzetten is ook niet handig, want dan gaan gebruikers ‘buiten’ shoppen.

Samenvatting

Stap één is dus het verkrijgen van inzicht: wat gebeurt er allemaal al? Stap twee is bewust keuzes maken om oplossingen wel of niet te beheren en als ze worden beheerd een formeler proces te hanteren. Zonder bewuste keuzes en afwegingen loop je vroeg of laat tegen problemen aan. Zorg dat je ze voorblijft!



Tip 3: Met Fusion Development overbrug je de kloof

Technische details worden bij low-code-platformen zo veel mogelijk afgeschermd. Dat zorgt dat de zogenaamde citizen developers zich niet druk hoeven te maken om dergelijke complexiteit en heeft als resultaat dat apps sneller klaar zijn en gebruikt kunnen worden. Helaas, je kunt met low-code niet alles bouwen. Soms mis je iets. Je hebt bijvoorbeeld een koppeling nodig met een extern systeem om data binnen te halen. Dat is waar fusion development om de hoek komt kijken.

Fusion what?

Microsoft heeft, als ze het heeft over fusion development, over het 'overbruggen van de kloof' tussen gebruiker en softwarebouwer. Fusion development brengt eindgebruikers, citizen developers, full-stack developers en IT-beheerders bij elkaar. Zo ontstaat een multidisciplinair team met maar één doel: snel en laagdrempelig oplossingen maken voor businessuitdagingen. Binnen dit team heeft iedereen baat bij de voordelen van low-code.

Fusion development in de praktijk

Hoe ziet zo'n samenwerking binnen een fusion development team eruit? Citizen developers gaan aan de slag met het Power Platform. Hebben zij aan de standaard functionaliteiten, componenten of (data-)koppelingen niet voldoende, dan schakelen zij een full-stack-ontwikkelaar in. De IT Pro'er zorgt er op zijn beurt voor dat de apps en data-integraties op de juiste manier uitgerold worden, en dat alles veilig en gecontroleerd gebruikt kan worden. Zo breid je de functionaliteit van het Power Platform samen uit, wissel je efficiënt kennis uit, doet iedereen waar hij goed in is en wordt al je development- en IT-capaciteit optimaal ingezet..

Opschalen wanneer nodig

En dreigt het echt een succes te worden? Dan kun je eenvoudig opschalen in Azure en andere diensten inpluggen om te zorgen dat jouw app schaalbaar en toekomstbestendig blijft.

Power Platform zelf uitbreiden

Fusion development helpt je dus om meer te bereiken met het Power Platform, doordat je connectoren en functionaliteit kunt toevoegen. Maar het zou zonde zijn om die maar in één project te gebruiken, dus kun je connectoren uitrollen binnen je hele organisatie. De data die je zo hebt ontsloten is dan beschikbaar voor alle andere Power Platform-ontwikkelaars. Veel ontwikkelaars delen hun uitbreidingen ook extern in het Independent Publisher Connector Program van Microsoft. Hierdoor kunnen ook andere organisaties jouw connector gebruiken.

Beheer zonder code

Fusion development houdt echter niet op bij het ontwikkelen van apps met het Power Platform. Ook het beheer van applicaties kun je voor een steeds groter deel aan citizen developers overlaten. Door een app te verpakken in een 'solution', kunnen zij hem zelf beheren delen met andere omgevingen of afdelingen

of hergebruiken als deel van een nieuwe oplossing. Allemaal zonder code te schrijven. Azure DevOps voorziet deze solutions van een versie geschiedenis en geeft ontwikkelaars de mogelijkheid om ontwikkel-, test-, acceptatie- en productie-omgevingen in te richten. Een dergelijk gestroomlijnd proces helpt een organisatie om professioneel met het Power Platform om te gaan en zorgt dat Fusion development als versneller fungeert bij het bedenken, testen, bouwen en uitrollen van digitale innovaties.

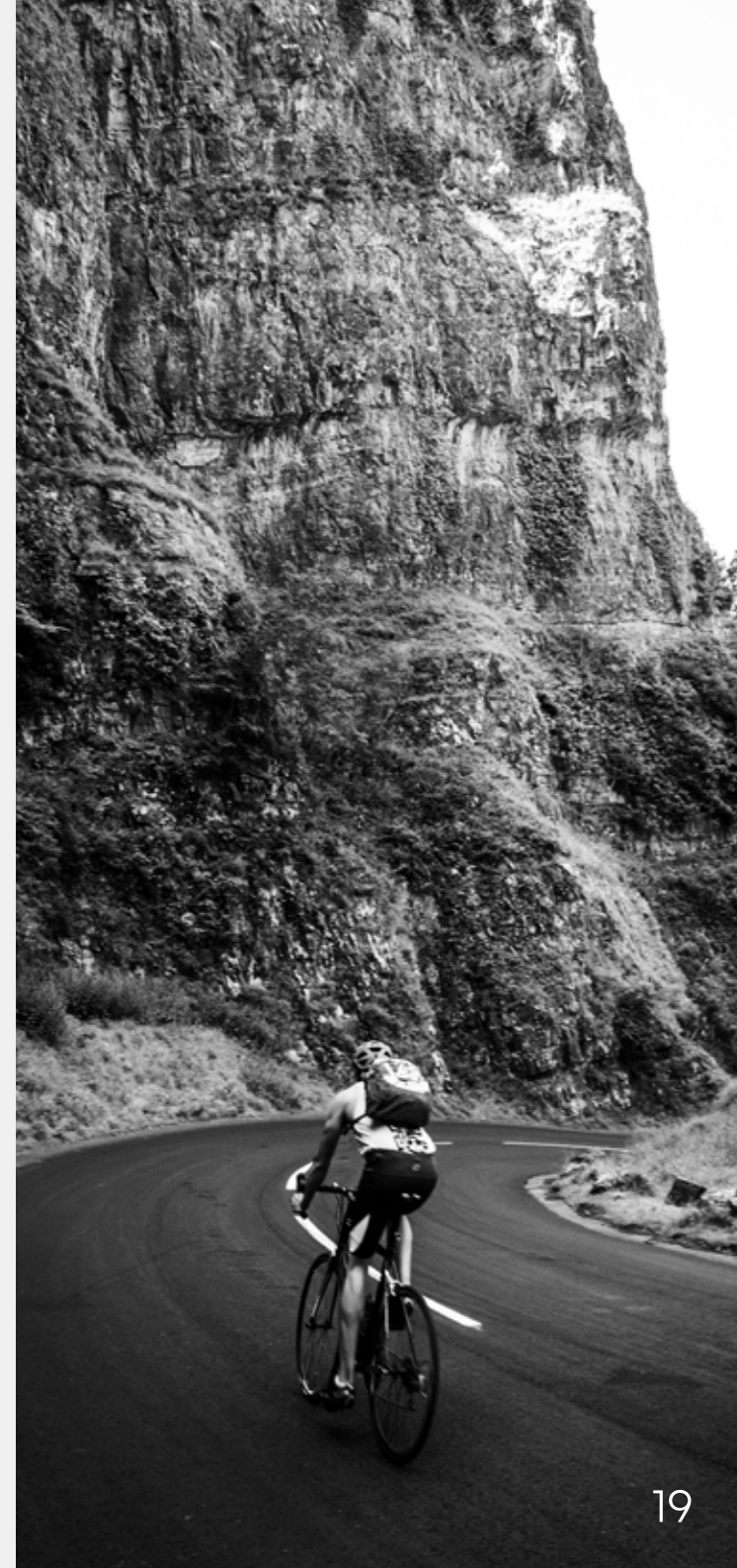


Tip 4: ook low-code oplossingen moet je testen

Het Microsoft Power Platform maakt softwareontwikkeling voor veel meer mensen toegankelijk. Van handige e-mail-hack tot enterprise-level bedrijfsproces: het is vaak te realiseren zonder code te schrijven. En als je geen code schrijft, hoef je ook niet te testen, toch?

Je klikt een app bij elkaar en hop, het kan naar productie. Dat is dus een misverstand. Of het nu gaat om low-code of traditioneel coderen, zonder testen iets in productie zetten is geen veilige en beheersbare manier van software ontwikkelen.

Heb je een team van twintig testers nodig voor je low-code-apps? Waarschijnlijk niet. Toch moet je wel serieus testen voor een Power Platformtoepassing in gebruik genomen wordt. Je wilt tenslotte kwaliteit afleveren. Maar waarom moet je software die al half af is voordat je begint, toch testen?



Ruis op de lijn

Het antwoord zit hem in communicatie. Als je een oplossing wil maken, begint dat met een idee in je hoofd. We weten allemaal hoe moeilijk het is om een beeld in je hoofd te vertalen naar papier. Zo werkt het ook met het bouwen van low-codeoplossingen. Stel: een team van negen mensen bedenkt dat een bepaald proces slimmer kan. Slechts een van hen gaat daadwerkelijk met de oplossing aan de slag. Ergens in dat proces zit een vertaalslag, en daar ontstaat ruis. Simpelweg omdat mensen elkaar nu eenmaal soms verkeerd begrijpen. Daarom is het altijd goed om af te spreken dat, als de app eenmaal af is, er nog een keer met een frisse blik naar te kijken. Ook bij low-code.

Het yin en yang van software testen

Software testers worden er vaak van beschuldigd dat ze alleen maar problemen zien, maar dat is eigenlijk niet hoe het werkt. Een softwaretester is vooral iemand die zich bij ieder stukje businesslogica afvraagt: Wat betekent dit eigenlijk? Het volledig doordenken van de functionaliteit van je app en uitwerken wat er nodig is om al je logica onder alle omstandigheden te laten werken, is een vak apart. Dus ja, een tester vindt dingen die mis zouden kunnen gaan. Dat is de negatieve kant. Aan de positieve kant: een tester zorgt voor kwaliteit. Jij maakt dingen. Een tester helpt je om het echt goed te maken. Dat schept vertrouwen in

het product en zorgt voor een probleemloze uitrol en productief gebruik. Als het goed is, zijn deze twee dingen in balans. Een beetje als yin en yang.

Drie testfasen

Ook het testen van low-code-applicaties vindt plaats op drie niveaus:

1. Requirements testen

Voor je begint, moet je een plan hebben. Ook bij low-code zijn goede requirements erg belangrijk. Zoals we al schreven in het hoofdstuk over Application Lifecycle Management, is het slim in deze fase al een tester mee te laten kijken. Hij of zij is er goed in om hiaten in requirements te herkennen die later voor problemen kunnen zorgen.

2. Integratie- en functionele testen

Het meest gedetailleerde, technische testniveau (de 'unittest') kun je bij low-code meestal overslaan. Maar het testen van functionaliteit van de app en integraties met andere systemen is natuurlijk wel belangrijk. Praten de modules met elkaar? Komt de data aan? Aannames doen op dit gebied is gevaarlijk en kan je in de volgende testronde, of in productie, voor serieuze problemen en vertragingen zorgen. Bij een functionele test gedraagt een

tester zich als gebruiker en voert alle handelingen uit zoals ze in de requirements zijn uitgewerkt. Zo weet je of de app ook écht doet wat de gebruiker gevraagd heeft.

3. Gebruikerstest

In de laatste testronde laat je de gebruiker zelf ook beoordelen of de app doet wat hij of zij gevraagd heeft. Sla deze fase nooit over! Een gebruiker kijkt met een hele andere blik naar software dan de techneuten. Ze zullen dus zeker nog opmerkingen hebben.

Hoe belangrijker de app, hoe meer testrondes er nodig zijn

Een veelvoorkomende misvatting is dat gebruikerstesten voor low-code genoeg zijn. Maar gebruikers testen software vanuit hun eigen perspectief en meestal niet erg systematisch. Dat betekent dat je gevaarlijke aannames uit het ontwikkelproces niet altijd boven water krijgt. Een gebruiker neemt bewust of onbewust aan dat een ontwikkelaar aan alles gedacht heeft. Maar 'alles' is, ook voor een ervaren ontwikkelaar, best veel en die aanname hoeft dus niet te kloppen.

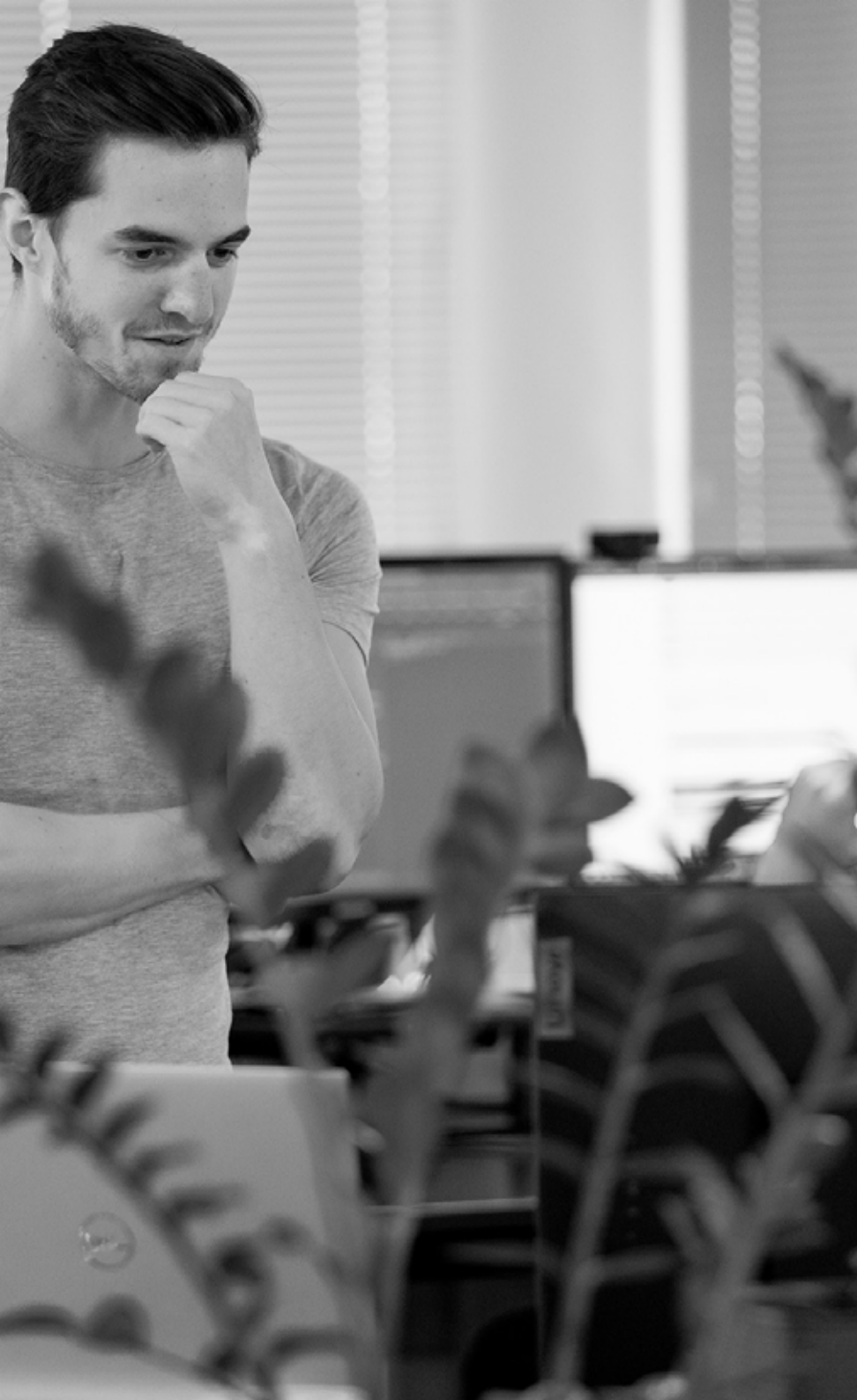
Low-code voedt dit soort aannames, omdat het ontwikkelproces minder technisch is en het dus makkelijk is om te veronderstellen dat 'alles automatisch' gaat. Hoeveel testrondes je nodig hebt, hangt natuurlijk wel af van de complexiteit van de app en het

aantal gebruikers. Heb je een appje voor jezelf gemaakt? Dan kun je inderdaad eens kritisch door je app heen klikken en constateren dat het aardig doet wat je wilt. Maar gaan meer gebruikers met de app werken, dan is er meer nodig. Zelfs een app die door vijf mensen gebruikt wordt, kan al verrassingen opleveren.

Hoe belangrijker en omvangrijker de app en hoe meer gebruikers, hoe groter de kans op onbedoelde effecten. Hoe strenger je kwaliteitseisen dus moeten zijn. Daarnaast heb je ook andere zaken nodig, die je ongetwijfeld herkent uit full-stack ontwikkelprocessen: documentatie, handleidingen en governanceafspraken over naamgeving, rechten, toegang en het opruimen van data. Ook deployment- en updateprocessen moet je vastleggen als je een low-code app breed wilt gaan uitrollen.

Specifieke tests voor low-code

Het mooie van Power Platform is dat heel veel dingen al klaar zijn. Microsoft zorgt dat je app altijd start en dat de infrastructuur op orde is. Daar hoeft je je niet druk over te maken. Standaardintegraties en connecties die je in je app hebt gebruikt, zullen het ook gewoon doen. Echter, voor de toegankelijkheid van de data zijn de gebruikers zelf verantwoordelijk. Dat kan misgaan. Wat als je een Teams team gebruikt in je app en dat Team wordt verwijderd? Wat als bepaalde mappen niet toegankelijk zijn?



Dat zijn uitzonderingen die je moet testen. Ook verwerking en presentatie van de data moeten getest worden. Bij low-code heb je meestal niet volledig in beeld welke functionaliteit ondersteund wordt. Platformen veranderen nou eenmaal iedere dag en je kunt niet alles weten. Je kunt bijvoorbeeld wel bedenken dat je bij slecht weer een regenemoji in de titel van je app wilt hebben, maar test dan wel of dat op alle apparaten werkt en of er geen onverwachte effecten optreden. Het zou zonde zijn als mails daardoor niet aankomen. Houd er ook rekening mee dat het gebruik van een app kan groeien. Drie gebruikers die al hun bestanden opslaan in één map, dat gaat misschien nog wel goed. Met 3.000 gebruikers is dat echt een ander verhaal.

Samengevat

Testen is onderdeel van het ontwikkelproces. Ook bij low-code. Hoe belangrijker en omvangrijker de app en hoe meer gebruikers, hoe belangrijker het testproces is. Gelukkig is er binnen het Power Platform al veel geregeld, maar ook bij low-codeapplicaties is – net als bij full-stack-softwareontwikkeling – de kwaliteit van je oplossing het resultaat van een doordacht testproces.

Tip 5: Adoptie van low-code moet je stimuleren

Met behulp van low-code-platformen kunnen mensen uit de business met IT-affiniteit dus zelf eenvoudig en zonder veel technische kennis apps bouwen. Maar hoe laagdrempelig een lowcode- platform als Microsoft Power Platform ook is, om te zorgen dat het echt omarmd wordt, moet je ook aandacht schenken aan het begeleiden, ondersteunen en opleiden van de citizen developers.

Het [Power Platform Adoption Maturity Model](#) is hier voor een nuttig hulpmiddel. Het geeft je een gefaseerde aanpak om klein te beginnen en stap voor stap te groeien naar een organisatie-brede uitrol van het Power Platform. Zie ook dit artikel op Microsoft.com [Repeatable patterns for successful Power Platform adoption](#).

Power Platform als groeimodel

Voor je begint met het promoten van het Power Platform moet je inzichtelijk maken welke low-code toepassingen er al in je organisatie gebruikt worden. Bepaal vervolgens per toepassing

wat de (technische) kwaliteit is en hoe toekomstvast de oplossing is. Waarschijnlijk kom je tot de conclusie dat herbouw van de toepassing met behulp van het Power Platform verstandig is.

Aan de hand van het Power Platform Adoption Maturity Model kun je goed bepalen wat je als organisatie moet inregelen om het gebruik van Power Platform tot een succes te maken.

Probeer niet gelijk op Level 500 in te stappen, dat is nergens voor nodig. Door op een beheersbare manier te groeien, voorkom je vervelende groeistuipen.

Unleash
the power of
technology

Nog meer lezen? Deze resources zijn een bezoekje waard:

- [Gartner Insights: Low-Code Application Platforms 2022](#)
- [Forrester Wave ranks top platforms for developers](#)
- [Transform your business apps with fusion development](#)
- [Ebook: Fusion development approach to building apps](#)
- [Microsoft Power Platform Center of Excellence Starter Kit](#)
- [Power Platform Build Tools for Azure DevOps](#)
- [Microsoft Power Platform CLI](#)
- [Power Platform PowerShell Modules](#)
- [Repeatable patterns for successful adoption](#)

Bliss

Training

Wij zijn Blis Digital!

Het Power Platform van Microsoft stelt je in staat om eenvoudig oplossingen te bouwen. Daarmee biedt het enorm veel kansen. Toch moet je ook over een aantal zaken goed nadenken voor je het Power Platform gaat inzetten. Realiseer je dat het niet iets is dat je er 'even bij doet'. Het is niet zomaar een tooltje waarmee de business leuke toepassingen kan gaan maken. Als je het power platform inzet, moet het een strategisch onderdeel zijn van je IT-landschap.

Gewoon 'aanzetten' van een low-code-platform zorgt geheid voor ellende, verlies van controle en brengt potentieel zelfs grote gevaren met zich mee op het vlak van beveiliging, datalekken en verstoringen. Om het Power Platform echt succesvol in te zetten, moet je dus vooraf over een aantal belangrijke zaken nadenken, zoals governance, security, documentatie, testen, support en ALM. Kortom, bezint eer gij begint. We hopen dat we je daarbij hebben geholpen met dit whitepaper.



Unleash
the power of
technology



Meer weten? Check onze
website blisdigital.com of
volg ons @blisdigital



Unleash
the power of
technology