

Low code / no code

Kennismaken met snelle en efficiënte applicatieontwikkeling

VOOR BETERE ICT BESLISSINGEN



Low code / no code

Kennismaken met snelle en efficiënte applicatieontwikkeling



Uitgegeven door het ICT informatiecentrum, Houten

© 2022 ICT informatiecentrum

Alle rechten voorbehouden. Het is de ontvanger van deze publicatie verboden de inhoud ervan geheel of gedeeltelijk te vereenvoudigen, openbaar te maken, digitaal te verspreiden of op welke wijze dan ook te distribueren, zonder voorafgaande toestemming van de uitgever. Hoewel deze uitgave met zorg is samengesteld, aanvaardt de uitgever geen enkele aansprakelijkheid voor schade ontstaan door het gebruik ervan en fouten of onvolkomenheden in de gepubliceerde teksten.

Inleiding

Low code en no code staan volop in de belangstelling. Gestimuleerd door de sterke behoefte aan en noodzaak voor verdere digitalisering, de grote vraag naar maatwerk applicaties, steeds kortere ontwikkeltijden voor bedrijfssoftware, de snel gewenste vervanging of aanpassing van legacy software en de enorme schaarste aan ervaren softwareontwikkelaars, neemt het aantal toepassingen van low code en no code softwareontwikkeling de laatste jaren enorm snel toe. U krijgt of heeft er dan ook ongetwijfeld mee te maken.

In dit e-boek maakt u kennis met wat low code en no code is en wat de eigenschappen, toepassingen, voordelen en nadelen ervan zijn. U leest over de verschillen tussen beide ontwikkelmethoden en waarom low code en no code steeds belangrijker worden.

Kortom, een e-boek dat u laat kennismaken met de actuele ontwikkelingen rond low code en no code en dat duidelijk maakt dat beide technieken voortaan een serieus alternatief zijn voor het realiseren van bedrijfssoftware en andere applicaties.

Wij wensen u alvast alle succes toe met uw low code of no code project!

ICT informatiecentrum

Kennis uit de praktijk

De inhoud van dit boekje is tot stand gekomen met medewerking van diverse specialisten voor dit thema. Zij leveren u de kennis uit de praktijk. Wij noemen ze onze 'kennispartners'. Staat een kennispartner als bron bij een artikel vermeld, dan kunt u het artikel plaatsen in de context van die kennispartner en zijn of haar activiteiten.

Het is goed om te weten dat niemand ons betaalt voor publicatie van zijn of haar bijdrage aan deze publicatie. Ook betaalt het ICT informatiecentrum niemand voor een tekstuele bijdrage. Zo zijn en blijven wij volledig onafhankelijk in de samenstelling van onze publicaties.

Uiteraard danken wij onze kennispartners voor hun medewerking. Mede dankzij hun ervaringen, visies, tips en adviezen worden ICT beslissingen iedere dag weer een beetje beter.

Inhoud

Inleiding	3
Kennis uit de praktijk	4
Wat is low code?	6
No code versus low code	13
Low code en no code is de toekomst	16
Snel en efficiënt applicaties ontwikkelen met low code	21
No code: evolutie of revolutie?	25
Low code helpt om competitief en vernieuwend te blijven	28
Standaard, maatwerk of low code?	29
No code software als aanvulling op je standaardsysteem	32
Waarmee kunnen wij u helpen?	34
Kennispartners	35

Wat is low code?

Bron: Apronto | [appronto.nl](https://www.appronto.nl)

Slimme applicaties vormen een steeds belangrijker onderdeel van het IT arsenaal dat moderne bedrijven in stelling brengen om competitief te blijven. De behoefte aan innovatie en een snelle time-to-market vraagt om ontwikkelmethoden die sneller en gemakkelijker zijn dan het traditionele 'code kloppen'.

Dan kom je al gauw uit bij low code en no code, twee vormen van softwareontwikkeling die je in staat stellen om met een minimum aan handmatige code innovatieve en kwalitatief hoogwaardige applicaties te bouwen. Maar wat zijn de belangrijkste verschillen tussen low code en no code? Wat levert low code development op?

Wat is low code development?

De definitie van low code die Forrester Research hanteert, luidt:

“Products and/or cloud services for application development that employ visual, declarative techniques instead of programming and are available to customers at low or no cost in money and training time to begin, with costs rising in proportion of the business value of the platforms.”

Een hele mond vol. In wat compacter en begrijpelijk Nederlands kunnen we low code development omschrijven als een methode van softwareontwikkeling die je in staat stelt om (zakelijke) apps snel en met een minimum aan handmatige code te ontwikkelen. Een belangrijk kenmerk van low code app development is dat het ontwikkelen op een visuele manier geschiedt. Developers maken gebruik van een grafische gebruikersomgeving, waarin configuraties grotendeels de plaats innemen van traditioneel geschreven computercode.

Kenmerkend voor low code is dat applicaties worden opgebouwd uit visuele blokken. Je kunt deze modules hergebruiken en eventueel aanvullen met handgeschreven code. De eerste fundamenten voor low code ontstonden in de jaren negentig van de vorige eeuw in de vorm van programmeertalen van de vierde generatie, modelgedreven ontwerpen en automatische codegeneratie.

Wat is no code?

“The future of coding is no coding at all”. Die uitspraak is afkomstig van GitHub CEO Chris Wanstrath en geeft eigenlijk in een notendop de essentie van no code development weer. No code gaat net als low code uit van visueel ontwikkelen, maar voert dit principe nog een stapje verder door dan low code.

Ontwikkelaars kunnen middels de methode van drag & drop, functionele componenten selecteren uit een digitale bibliotheek. De elementen die je nodig hebt om een applicatie te bouwen, breng je vervolgens onder in een visuele workflow. No code modelleert bedrijfsprocessen in de vorm van flowcharts. Aan de hand van die informatie wordt met behulp van een visuele composer een eerste versie van een applicatie gebouwd, die gelijk ingezet kan worden. Die prototypes kunnen door gebruikers van het platform heel gemakkelijk tussentijds worden gewijzigd, bijgesteld of geoptimaliseerd.

Wat is het verschil tussen low code en no code?

Hoewel low code en no code veel raakvlakken hebben en deels gebruikmaken van dezelfde principes, zijn er wel verschillen tussen de twee ontwikkelwijzen. De belangrijkste verschillen tussen de twee zijn:

- No code legt een nog sterkere nadruk op snelheid en innovatie.
- No code vergt nog minder specialistische IT- en programmeerkennis dan low code. Vandaar ook dat no code development een

belangrijke plek inruimt voor zogenoemde 'citizen developers'. Dit zijn mensen die wel heel handig zijn in het ontwerpen van applicaties, maar geen vakgeschoolde of professionele IT'ers zijn.

- De gebruikte technologieën verschillen enigszins van elkaar. Het gros van de no code platforms (Betty Blocks, WEM, AppSheet) steunt bijvoorbeeld op HTML en Javascript. Low code platforms als Mendix, OutSystems of Thinkwise zijn daarentegen geënt op Java en .NET.
- Low code is sterker gericht op ontwikkelaars, no code primair op eindgebruikers.
- Zet je een low code platform als Mendix tegenover een no code platform als Betty Blocks? Dan zie je dat low code development meer mogelijkheden biedt voor maatwerk op elk niveau.
- Met low code kun je complexe applicaties ontwikkelen. No code heeft op het gebied van complexiteit meer beperkingen en is vooral toegesneden op het bouwen van eenvoudige apps.

Trends en cijfers

Hoe zit het met de populariteit van beide ontwikkelvormen? De markt voor low- en no code is de afgelopen jaren met miljarden euro's gegroeid. De verwachting is dat de markt voor low code development in 2025 een totale waarde heeft van 45,5 miljard dollar. Die groei is het gevolg van een toenemende behoefte aan korte ontwikkelcycli en lagere ontwikkelingskosten.

Forrester ziet in het verlengde hiervan tevens een toenemende behoefte aan flexibele platforms. Die bieden namelijk de ontwerp- en ontwikkeltools die je nodig hebt om apps helemaal af te stemmen op de wensen en behoeften van gebruikers. Noord-Amerika is nog steeds de grootste markt, terwijl de Aziatisch-Pacifische markt het snelst groeiende afzetgebied is. Mendix en OutSystems zijn de actuele marktleiders op het gebied van low- en no code development.

De wereld van low- en no code blijft dynamisch en is momenteel in de ban van een paar belangrijke trends:

- De behoefte van klanten aan spannende, op maat gemaakte customer journeys via verschillende touchpoints. Dit vraagt in toenemende mate om multi- en omnichannel ervaringen.
- Organisaties breiden met no- en low code platforms hun kernsystemen uit om datagedreven en gebruiksvriendelijke ervaringen aan te bieden. Het resultaat: nieuwe marktkansen. Analisten schatten het huidige aantal ontwikkelingsprojecten dat gebruikmaakt van low code op 10 tot 20 procent van de totale bedrijfsactiviteit. Dit aandeel zal naar verwachting binnen 4 jaar stijgen naar zo'n 65 procent.
- De behoefte aan en het onderzoek naar AI-gedreven software zit in de lift. Deze trend zal nog wel een tijdje aanhouden als we kijken naar de nog steeds groeiende databerg die we met zijn allen produceren.
- Ontwikkelteams worden steeds meer een uitgebalanceerde mix van professionele ontwikkelaars en citizen developers. Zo creëer je bij het ontwikkelen van een applicatie de beste balans tussen technische vereisten en zakelijke behoeften.

Wat levert low code development op?

Er zijn verschillende redenen die een keuze voor low code development rechtvaardigen. Tijd om eens te kijken wat low code je als organisatie concreet oplevert.

Lagere ontwikkelkosten

Ten opzichte van traditioneel ontwikkelen met handmatig geschreven codes is low code development goedkoper. Omdat je op een visuele manier ontwikkelt en deels terug kunt vallen op herbruikbare bouwstenen (modules), gaat het ontwerpen,

ontwikkelen, aanpassen, opschalen en onderhouden van een business-app een stuk sneller. Zo kun je een ontwikkeltraject dat normaal een half jaar duurt al gauw terugbrengen tot een of twee maanden. Het spreekt voor zich dat je dit ook merkt in je portemonnee.

Meer flexibiliteit en wendbaarheid

Low code geeft je organisatie extra flexibiliteit en wendbaarheid. IT innovaties verlopen tegenwoordig extreem snel, waardoor ook de behoeften vanuit de markt continu veranderen. Technologieën als kunstmatige intelligentie (AI) en machine learning veranderen bijvoorbeeld van een luxe in een absolute noodzaak, zeker omdat de hoeveelheden data die organisaties gebruiken alsmaar toenemen.

Als je een low code platform gebruikt, kun je applicaties heel gemakkelijk aanpassen. Het resultaat is navenant: omdat je veel sneller kunt inspelen op veranderende marktomstandigheden en dynamische zakelijke behoeften, wordt je als organisatie flexibeler en wendbaarder. Zo betaalt low code development zich dus ook uit nadat een applicatie eenmaal in gebruik is genomen.

Minder bugs

Minder code betekent in de regel ook minder bugs. En minder bugs leiden weer tot minder irritaties bij netwerkbeheerders en eindgebruikers. Bovendien scheelt het tijd en werk qua aanpassing en onderhoud. Daarnaast hebben toonaangevende low code platforms features die debugging een stuk gemakkelijker maken. Mendix heeft bijvoorbeeld een standaard feedback-knop waarmee je bugs eenvoudig kunt melden. Een ontwikkelaar weet dan precies waar en bij wie een bepaald probleem zich heeft voorgedaan.

Meer ruimte voor innovatie

Als je minder tijd hoeft te spenderen aan het ontwikkelen en managen van je applicaties, houd je meer ruimte over voor innovatie. Vooruitdenken, doorontwikkelen en nieuwe ideeën uittesten; het wordt gewoon gemakkelijker. Low code development betekent ook dat je verder kijkt dan de techniek achter een applicatie. De focus ligt vooral op de zakelijke kant en praktische meerwaarde. Zo krijg je applicaties die concrete problemen en uitdagingen oplossen en helemaal toegespitst zijn op de strategische doelstellingen van een organisatie.

Je kunt vrijwel elk type applicatie bouwen

Met low code kun je vrijwel elke denkbare applicatie bouwen. Bestaande legacy systemen uitbreiden en koppelen aan je applicatielandschap? Met een platform als Mendix gaan voor native mobile? Het kan allemaal.

Low code platforms zijn cloudvriendelijk

De gang naar de cloud is een actuele IT ontwikkeling die niet meer te stoppen is. Organisaties plaatsen steeds meer data en infrastructurele onderdelen in de cloud, terwijl veel fonkelnieuwe applicaties cloud-only zijn. Het goede nieuws is dat low code platforms erg cloudvriendelijk zijn. Hun PaaS structuur (platform-as-a-service) is volledig geoptimaliseerd voor gebruik in de cloud.

Minder afhankelijk van schaarse IT'ers

De IT markt is een krappe. De behoefte aan goede IT'ers overtreft de beschikbaarheid, waardoor het niet gemakkelijk is om gespecialiseerde IT'ers aan je bedrijf te binden. Ze hebben namelijk veel keus en zijn doorgaans verwend met hoge salarissen en goede arbeidsvoorwaarden. Met low code (en met no code nog meer) wordt het bouwen van applicaties een stuk gemakkelijker, omdat je minder mensen met specialistische programmeerkennis nodig hebt.

Extra schaalbaarheid

Door no code of low code platforms te gebruiken, wordt je IT infrastructuur schaalbaarder. Je kunt projecten en applicaties gemakkelijker in de gaten en up-to-date te houden. Door het verkorten van de ontwikkeltijd en door het gebruiken van een gezamenlijke infrastructuur, kun je een uitgebreide applicatieportfolio onderhouden. Dit gaat bovendien niet ten koste van essentiële zaken als transparantie en veiligheid.

Je betreft stakeholders meer bij het ontwikkelproces

Low code app development maakt het mogelijk om features binnen enkele uren of dagen te tonen aan stakeholders en eindgebruikers. De visuele ontwikkelomgeving zorgt er bovendien voor dat iedereen de applicatie in wording echt in actie ziet. Het resultaat: meer betrokkenheid van alle partijen omdat je iedereen kunt meenemen op de spannende reis richting een goed werkend eindproduct.

No code versus low code

Bron: Jump | jump.nl

Steeds meer organisaties transformeren hun IT landschap van standaard softwarepakketten naar applicaties op basis van low code of no code. Wat betekenen low code en no code? En wat is het verschil tussen deze twee?

No code en low code platforms zijn upcoming. Ze sluiten aan bij digitalisering van de samenleving. Er is een grote vraag naar developers, waardoor veel bedrijven tekorten hebben. Dankzij low code kunnen ontwikkelaars sneller web apps en mobiele apps ontwikkelen. Vaak is er binnen enkele dagen een applicatie te bouwen die een probleem oplost. Dit minimal viable product is snel inzetbaar, levert feedback op voor toekomstige doorontwikkeling en kan uiteindelijk verder verfijnd worden. Zo is er meer focus op innovatie en zijn ontwikkelaars minder tijd kwijt aan debuggen.

Er zijn verschillende low code partijen. De leading low code platforms zijn volgens de The Forrester Wave: Mendix, OutSystems, Salesforce, Kony en Thinkwise. Ook zijn er naast low code platform een aantal no code platforms, zoals Airtable, AppSheet, FileMaker, Kintone en BettyBlocks.

Low code en no code: dicht bij elkaar...

Er is geen hard onderscheid tussen no code en low code. Ze zijn onderdeel van een spectrum van software development, dat van no code via low code naar full code loopt. No code en low code platforms zijn dus vergelijkbare oplossingen, die het ontwikkelen van software makkelijker en sneller maken dan traditioneel programmeren. We zien dat low code en no code platforms wel eens in elkaars vaarwater zit. Zo is BettyBlocks een no code platform, maar is het wel mogelijk om eigen blokken te ontwikkelen op basis van webtechnologieën. Dat terwijl low code platform Mendix een eigen no code oplossing heeft gepresenteerd: Mendix Studio.

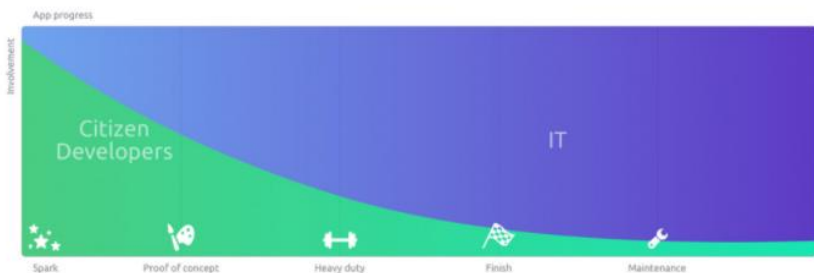
Het ontwikkelen van software gaat voor zowel no code als low code platforms via een grafische interface met drag & drop tools. Veel aspecten van een applicatie worden automatisch aangemaakt op basis van de schermen, zoals de datastructuur. Ook kunnen logica en validaties eenvoudig worden toegevoegd, meestal via een visuele flow. Randzaken als dependency management, testen en deployment worden geautomatiseerd.

Door de grafische interface is programmeren via low code en no code platforms makkelijker dan traditioneel programmeren. Er is minder diepgaande kennis nodig om software te ontwikkelen. Hierdoor is het mogelijk om sneller te programmeren, al neemt die snelheidswinst af als een applicatie complex in elkaar zit.

...maar ook heel verschillend

Natuurlijk zijn er ook verschillen. No code legt de focus op innovatie. Gebruikers worden in staat gesteld om hun ideeën direct om te zetten in applicaties. De zogenoemde citizen developers zorgen voor een brede innovatie binnen een bedrijf. Deze citizen developers zijn mensen in je bedrijf zonder programmeerkennis, maar die wel affiniteit hebben met Excel. Dankzij no code zijn zij in staat net die extra stap te zetten.

No code is gericht op medewerkers met slimme ideeën. Zij kunnen deze ideeën zelf omzetten in een werkende app. Toch is dit niet oneindig, zoals in



dit schema te zien is. Uiteindelijk zijn er IT'ers nodig die de applicatie van een 'proof of concept' omzetten in een operationele app.

Bij low code ligt de focus juist op het gemakkelijker maken van applicatieontwikkeling, maar wel met alle vrijheid die traditioneel programmeren ook had. Ideaal dus voor programmeurs die ontzorgd willen worden of minder bezig willen zijn met devops.

Wat verder opvalt is dat de gebruikte technologieën verschillen. De meeste no code platforms zijn gebaseerd op webtechnologieën als HTML en Javascript, terwijl low code gebaseerd is op Java en .NET.

Welk platform past beter bij je wensen?

Kies niet blindelings voor een platform, maar laat je begeleiden om een goede afweging te maken. Vaak heeft low code een hoger kostenplaatje, maar hebben ontwikkelaars meer vrijheid om alles te kunnen maken. Wil je jouw ontwikkelaarsteam verbeteren of wil je (in een later stadium) complexe applicaties bouwen, dan is low code zeer geschikt.

No code is vaak vriendelijker geprijsd. Wel is de oplossing meestal geschikt voor één type app, zoals mobiele apps. Dankzij no code kun je medewerkers een tool geven om snel te prototypen. Dit stimuleert innovatie.

Kortom, start met low code als je wil digitaliseren. Werk je inmiddels volledig digitaal? Dankzij no code stel je iedereen in staat om te innoveren.

Low code	No code
Applicatieontwikkeling & sneller programmeren	Innovatie
IT'ers	Iedereen (citizen developers)
Java & .NET	HTML & Javascript
Eén app met veel functies	Veel losse apps

Low code en no code is de toekomst

Bron: Software voor jou | softwarevoorjou.nl

De softwarewereld is constant in beweging, maar welke richting heeft de toekomst? In dit artikel nemen we je mee in de richting die de toekomst heeft: het ontwikkelen op basis van low code en no code platforms.

Low code en no code

Low code en no code behoren beide tot dezelfde familie. Tot dezelfde familie behoort ook het full code of high code ontwikkelen. Tussen low code en no code zijn er geen grote verschillen. De verschillen liggen voornamelijk in de details en de uiteindelijk softwareoplossing die ontwikkelt gaat worden. Deze bepaalt welk platform gebruikt gaat worden. Low code en no code platforms zijn vergelijkbare oplossingen, die ervoor zorgen dat het ontwikkelen van software sneller en gemakkelijker gemaakt wordt dan bij het ouderwets programmeren. Omdat ze zo vergelijkbaar zijn willen ze nog wel eens in elkaars vaarwater zitten.

No code

Bij no code wordt de focus gelegd op innovatie. Gebruikers hebben de mogelijkheid om hun ideeën direct om te zetten in een softwareoplossing, bijvoorbeeld een applicatie. Bij no code hebben de zogenoemde citizen developers binnen je bedrijf de mogelijkheid om een brede innovatie te zorgen. Deze citizen developers hoeven niet beschikken over programmeerkennis. Wel belangrijk is enige affiniteit met Excel. Met no code zijn zij in staat om hun slimme ideeën zelf om te zetten in een werkende app. Denk hierbij bijvoorbeeld aan de Microsoft Power Apps. Bij no code ligt de focus dus op medewerkers met slimme ideeën. Met no code hebben zij de mogelijkheid om deze ideeën zelf om te zetten in een werkende app. Toch zijn de mogelijkheden voor no code niet oneindig. Er zullen altijd IT'ers binnen je bedrijf nodig zijn die de applicatie van een eerste versie omzetten

in een daadwerkelijk operationele app. No code platforms zijn ideaal voor het ontwikkelen van niet al te complexe applicaties voor intern gebruik. Denk hierbij bijvoorbeeld aan een simpel CRM systeem of een applicatie voor het koppelen van formulieren aan achterliggende tabellen en datastromen. Of een applicatie om een leadlijst met verschillende rollen en rechten bij te houden. Het gaat voornamelijk over datastromen die je in Excel kan beheren, maar door hier een eenvoudige applicatie voor te maken heb je meer inzicht en grip op deze gegevens. Voor eenvoudige (interne) toepassingen is no code een prima oplossing.

Low code

Wil je een stap verder zetten dan mogelijk is met een no code platform, dan kom je uit bij een low code platform. Om met een low code platform te kunnen werken is er enige programmeerkennis vereist. Low code is dan ook vooral geschikt voor zeer handige citizen developers en natuurlijk voor 'echte' developers. Voor onervaren gebruikers is een low code platform al snel veel te complex. Low code platforms bieden meer mogelijkheden en functionaliteiten dan no code systemen. Toch worden de mogelijkheden van deze platformen vaak niet optimaal benut. Hoeveel meer mogelijkheden er zijn is afhankelijk van het platform dat je kiest. Eenvoudige codes, zoals bijvoorbeeld in Excel het optellen en vermenigvuldigen, begrijpen de meeste mensen wel en worden daarom ook veel gebruikt. Worden de vergelijkingen moeilijker, dan zullen de meeste 'normale' gebruikers al snel afhaken. Maar heb je veel kennis van Excel of ben je zelf een developer, en kun je dus alles uit een low code systeem halen, dan kom je best ver. Ondanks dat de hoeveelheid code in een low code beperkt is, kan je vaak binnen een paar dagen een goed werkende app ontwikkelen. Low code is daarom erg geschikt voor applicaties die intern gebruikt gaan worden. Denk hierbij aan een bestel app waarmee medewerkers bijvoorbeeld gereedschap, een nieuwe laptop of andere artikelen voor hun werk kunnen aanvragen.

Er zijn ook nadelen aan werken met low code platforms. Zo is het relatief simpel om applicaties te ontwikkelen. Echter moeten deze vaak gekoppeld

worden aan bestaande systemen, zoals interne CRM of ERP systeem. Voor het ontwikkelen van een dergelijke koppeling is ontwikkelkennis vereist.

Werken met een low code platform is daarom niet altijd eenvoudig. Enige sturing en begeleiding vanuit een dedicated team zal dus nodig zijn. Waar je ook rekening mee moet houden bij een low code platform is dat er bijna altijd gewerkt wordt met een standaard user interface, die vaak niet naar je eigen wensen kan worden aangepast. Denk goed na voordat je een app ontwikkelt of het bijvoorbeeld belangrijk is om je eigen huisstijl te gebruiken in de uitstraling, is dat het geval dan is het belangrijk om low code platform te kiezen waarbij dat wel mogelijk is of toch te kiezen voor een maatwerk oplossing.

High code oftewel het ‘echte’ coderen

Naast no code en low code platforms is er natuurlijk ook het ‘echte’ coderen van scratch, oftewel high code. Professionele ontwikkelaars hebben vaak eigen high code platforms vormgegeven waarmee zij complexe maatwerk applicaties ontwikkelen. Het gaat hier over ‘echt’ coderen. Het nadeel van high code platformen is dat alleen ervaren developers genoeg vakkennis hebben om er gebruik van te kunnen maken.

Een high code platform is de beste oplossing als je een optimale gebruikerservaring wil waarborgen of als de processen zo complex en uniek zijn dat je deze niet kan oplossen met een no code of low code platform. Maar ook als je een klantportaal nodig hebt om alle informatie uit je ERP systeem aan je eindgebruikers beschikbaar te stellen. Zijn je processen nog helemaal niet gedigitaliseerd, dan is het met een high code platform mogelijk om vanaf scratch een applicatie op maat te laten ontwikkelen. Maar ook als sommige processen al gedigitaliseerd zijn, dan kan je deze toevoegen aan je maatwerkapplicatie. Volstaat een no code of low code platform niet omdat je processen te complex zijn of omdat je jouw applicatie extern wil gaan gebruiken, dan is een high code platform voor jou de beste keuze. Let er dan op dat je de juiste ervaren developers in dienst hebt, aanneemt of inhuurt.

Wanneer past een low code of een no code platform beter bij je wensen?

Het is belangrijk voordat je een keuze maakt om goed na te denken waarvoor een dergelijk platform binnen je onderneming gebruikt gaat worden. Vaak is een low code platform duurder, maar geeft deze de ontwikkelaars meer vrijheid om alles te kunnen ontwikkelen. Een low code platform is zeer geschikt als je jouw IT ontwikkelaars team wil verbeteren, maar ook als je complexere applicaties wilt ontwikkelen of dat op een later moment wil gaan doen. Een no code platform is vaak goedkoper en biedt minder mogelijkheden. Het platform is meestal geschikt voor één type app, bijvoorbeeld enkel de mogelijkheid om mobiele applicaties te ontwikkelen. Echter kan je met een no code platform je medewerkers (citizen developers) een tool geven om snel mee te prototypen. Uiteindelijk zal dit innovatie binnen je onderneming stimuleren.

Even kort door de bocht; start met low code als je wil digitaliseren. Zijn al je processen inmiddels al volledig digitaal, dan stel je met een no code platform je medewerkers in staat om voor innovatie te zorgen.

In het overzicht zie je de grootste verschillen. Wellicht kan je met behulp van deze verschillen de voor jou beste keuze maken;

Toekomst voorspellen?

Een aantal grote partijen in de software wereld voorspellen dat Low code applicatie platformen in 2024 verantwoordelijk zijn voor 65% van alle ontwikkelde applicaties. Dit betekent dat het overgrote deel van alle applicaties gebouwd gaan worden met behulp van dergelijke platformen en overige tools die het mogelijk maken om apps te ontwikkelen.

No-Code	Low-Code	High-Code
Geen specifieke programmeerkennis nodig (citizen developers)	Geschikt voor developers/IT'ers	Geschikt voor ervaren developers
Ter vervanging van interne excel documenten	Meerdere variabelen en logisch formules	Sterk wisselende data met complexe formules
Veel losse apps	1 app met veel functies	
Simpele applicatie voor intern gebruik	Simpele applicatie voor (intern) gebruik	Complexe maatwerk applicaties
Standaard User Interface	Standaard User Interface	User Interface is naar eigen wens in te richten
Standaard functionaliteiten	Schaalbaar	Schaalbaar
Standaard security certificering	Standaard security certificering	Maatwerk security oplossingen
Innovatie	Sneller ontwikkelen	
	Koppelen aan klantportaal	Koppelen aan klantportaal
	Data uit meerdere bronnen kunnen koppelen	Data uit meerdere bronnen kunnen koppelen
		Optimalisatie mogelijkheden voor SEO & conversie

Snel en efficiënt applicaties ontwikkelen met low code

Bron: Quint | quintgroup.com

Wil je als organisatie nieuwe applicaties ontwikkelen, dan kun je in zee gaan met een pakketleverancier of zelf een maatwerkoplossing (laten) ontwikkelen. Een interessante hybride oplossing is een low code platform. Daarmee kunt u een applicatie naar wens bouwen zonder dat programmeerkennis nodig is. Maar waar moet u op letten bij de keuze en implementatie van een low code platform? Hoe zorgt u dat het platform aansluit bij uw IT omgeving, dat de investering voldoende opbrengt en dat u niet overgeleverd bent aan één leverancier?

Low code platforms (ook rapid application platforms ofwel RAD genoemd) of no code platforms zijn niet nieuw. Ze zijn al lange tijd op de markt en hebben zich bewezen als volwassen en robuuste producten. Voorbeelden van platforms zijn OutSystems, Mendix en Betty Blocks. Ook hebben grotere software/SaaS pakketten zoals ServiceNow, Salesforce en Pega uitgebreide low code mogelijkheden. Zowel bij mid market als bij enterprise organisaties winnen low code platforms aan populariteit.

Bij low code maken ontwerpers gebruik van een grafische gebruikersomgeving en het instellen van configuraties in plaats van het schrijven van traditionele computercode. Dat maakt het ontwikkelen niet alleen eenvoudiger en mogelijk voor niet-IT'ers, maar ook sneller, omdat applicaties worden opgebouwd uit visuele blokken. Daarnaast blijft het vaak mogelijk om handgeschreven broncode in te voegen.

De voordelen van low code

Dat klinkt als het beste van twee werelden en dat kan het ook zijn. Low code heeft, naast het grote gebruiksgemak, een aantal voordelen vergeleken met alles zelf ontwikkelen of een kant-en-klare oplossing afnemen.

Lage time-to-market en investering

Aangezien low code platforms een productiviteit hebben die twee tot vijf keer zo snel is als traditionele softwareontwikkeling, kunnen nieuwe apps, websites en enterprise applicaties in zeer korte tijd worden ontwikkeld. Door de lage time-to-market is het mogelijk opdrachtgevers, managers en andere stakeholders snel resultaat te laten zien. Door zaken als Agile en continuous delivery (in korte cycli snel nieuwe (deel)producten opleveren) is het mogelijk snel te reageren op kansen in de markt. Het gevolg van sneller ontwikkelen is ook dat de investeringskosten significant lager zijn.

Laag projectrisico

Veel functionaliteit is standaard aanwezig in low code platforms, zoals security, bescherming van data, zoekfuncties en de integratie met andere systemen. Ook voor specifieke sectoren of bedrijfsfuncties zijn vaak standaardfuncties aanwezig. Omdat het ontwikkelproces transparant is, worden eindgebruikers eerder bij het project betrokken en komen eventuele inconsistenties en andere problemen vroeg aan het licht en kunnen ze direct worden opgelost. Dat alles zorgt voor minder risico's en uw organisatie kan zich concentreren op de businessdoelstellingen.

Automatisch klaar voor browser toegang en mobiele devices

De meeste low code platforms maken de applicaties die erin worden gemaakt automatisch op maat voor smartphones en tablets. Ook zijn applicaties eenvoudig aan te passen, zodat ze via een browser toegankelijk zijn.

Niet alleen hosanna

Hoewel low code absoluut aantrekkelijke kanten heeft, zijn er ook uitdagingen. Ten eerste moet een low code platform worden aangeschaft. Dat stelt organisaties voor de keuze welk platform te kiezen en om toch een out-of-pocket investering te doen, voordat er kan worden begonnen.

In de relatieve eenvoud die low code platforms bieden, schuilt ook een gevaar: in plaats van dat doordachte oplossingen worden ontworpen door IT professionals, kan het low codeontwikkelaars verleiden tot quick fixes. Goede IT professionals zijn gewend om ontwikkelde software volgens een architectuur te ontwikkelen, te testen én om tijd te nemen om nut en noodzaak te bepalen. Door het gemak van low code kunnen ongewenste shortcuts worden genomen. Men spreekt dan van 'powertools for powerfools'.

Schaalbaarheid vormt tevens een belangrijk aandachtspunt. Een flexibel low code platform maakt gebruik van een cloudgebaseerde architectuur die schaalbaar is qua design (webscale architecture). Deze architectuur stelt een organisatie in staat om op een Agile manier systemen te bouwen die een hoge mate van beschikbaarheid en flexibiliteit bieden en een continuous delivery werkwijze ondersteunen. Dit wordt een must-have in de ontwikkeling van enterprise software, maar niet alle leveranciers bieden deze mogelijkheid. In de afweging welk low code platform te kiezen is dit dus een belangrijke overweging in hoeverre schaalbaarheid en flexibiliteit een integraal onderdeel vormen van de platform functionaliteit.

Rondom low code is het dus belangrijk om doordachte keuzes te maken. We noemen een paar zaken waarop u in ieder geval zou moeten letten.

Voorkom vendor-lock-in

De platformkeuze, de bijbehorende specifieke ontwikkeltooling en de benodigde expertise hieromtrent brengt het risico van een vendor lock in met zich mee. Om dit te voorkomen, is het opstellen

van een risicoanalyse nodig. Zo is het belangrijk dat de interne organisatie voldoende overzicht kan houden over dat wat op het platform is ontwikkeld en goede documentatie hierover bijhoudt om te voorkomen dat een black box ontstaat. Ook is het aan te raden om een bepaalde mate van modulariteit of gelaagdheid toe te passen, zoals een data laag die gescheiden blijft van een workflow laag. Zo is niet alle software gebaseerd op één platform en blijft het mogelijk in de toekomst voor bepaalde modules van low code platform te wisselen.

Ontwikkel een strategie

De keuze om een low code platform in te richten, heeft verstrekkende gevolgen. Daarom is het aan te bevelen een strategie te ontwikkelen en bewust te kiezen voor de juiste partners.

Werk volgens DevOps

Als u low code volledig wilt benutten, raden we u aan om uw organisatie in te richten op basis van DevOps. Met deze manier van werken komen de snelheid en beheersbaarheid die low code biedt beter tot hun recht.

No code: evolutie of revolutie?

Bron: Kotuur | kotuur.com

Er is een discussie gaande over de rol van no code software. Kamp A zegt dat no code simpelweg hoort bij de ontwikkelingen in softwareland van de afgelopen jaren. De vraag en behoefte in de markt zorgden ervoor dat het zich langzaam zo heeft gevormd; “geleidelijke ontwikkeling of groei”, zoals Van Dale de evolutie beschrijft. Kamp B zegt echter dat dit helemaal niet zo langzaam is gegaan en dat no code niet alleen een softwareontwikkeling is, maar een ommekeer voor organisaties. Oftewel, een revolutie; 'een plotselinge verandering in de bestaande toestand'. Laten we kijken naar waar deze uitspraken vandaan komen en wie weet kunnen we aan het eind een conclusie trekken over de vraag: is digitalisering met no code een evolutie of een revolutie in software?

No code software als evolutie

Volgens de evolutietheorie heeft al het huidige leven op aarde één gemeenschappelijke voorouder: de bacterie. Door natuurlijke selectie en mutatie zijn bepaalde soorten uitgestorven en andere soorten juist uit elkaar ontstaan. Alles is door de jaren heen doorontwikkeld dankzij de evolutie. Zo zijn mensen volgens deze theorie ontstaan uit aapachtigen.

Evolutietheorie toegepast op software

Ondanks dat software niet uit zichzelf doorontwikkelt, zou je wel kunnen stellen dat het proces hetzelfde verloopt als met evolutie. Alle software begon tientallen jaren geleden met één eerste versie en sindsdien zijn de talen en tools om oplossingen mee te maken continu veranderd. Sommigen zijn ‘uitgestorven’ doordat er geen behoefte meer aan was (natuurlijke selectie). Anderen zijn doorontwikkeld tot de versies die we vandaag de dag kennen (mutatie). Zo gingen we van de programmeertaal 3GL ook naar 4GL, vervolgens kwam low code en sinds enkele jaren is no code in opkomst.

Gebruikers hadden door ontwikkelingen in het dagelijks leven behoefte aan ontwikkelingen in softwareland. Ontwikkelingen die software sneller, makkelijker of flexibeler maakten. Maar zoals we al zeiden: vanaf de eerste versie gingen hier tientallen jaren overheen. Er is als je het zo bekijkt dus geen sprake van een revolutie. Dit kunnen we een software-evolutie noemen. Of toch niet...?

No code software als revolutie

Als we even afstappen van de evolutietheorie en ons verdiepen in de gebeurtenissen van de 16e t/m de 18e eeuw, dan zien we hele andere dingen gebeuren. Dan zien we namelijk de industriële revolutie en de wetenschappelijke revolutie. Twee plotselinge veranderingen in de tijd die zijn veroorzaakt door evolutionaire ontwikkelingen, namelijk de stoommachine en wetenschappelijke inzichten. Deze ontwikkelingen zijn ontstaan doordat huidige zaken steeds verder ‘gemuteerd’ werden. James Watt had de eerste werkende stoommachine bijvoorbeeld nooit kunnen maken als de natuurkundige theorieën die dit mogelijk maakten niet al waren bedacht. Deze evolutionaire ontwikkelingen behoren vandaag de dag echter tot een ‘revolutie’ doordat zij ervoor zorgden dat de samenleving totaal veranderde.

Softwarerevolutie

Als we deze manier van denken toepassen op no code software, dan kunnen we onszelf misschien wel overtuigen dat het ook een revolutie aanwakkert. No code software is op zichzelf inderdaad een IT ontwikkeling die voortkomt uit andere ontwikkelingen, een voortvloeiende evolutie. Maar de gevolgen zijn grandioos; een totale verandering van organisaties.

Het disruptieve effect van no code komt door de mogelijkheden tot citizen development en citizen adoption vanuit IT. De gevolgen hiervan zijn een veel dynamischere bedrijfsomgeving en een organisatie die agile is in plaats van alleen agile werkt. Er komt een hele andere rolverdeling. Normale

bedrijfscollega's nemen de taak om applicaties te maken van de IT afdeling over, terwijl de technische experts van de IT afdeling een actievere rol gaan spelen in het ontwikkelproces en overzicht kunnen houden.

Daarnaast is no code een manier om organisaties naar de cloud te helpen en zo schaduw-IT te verhelpen en one single source of truth te genereren. Iedereen kan zo realtime naar dezelfde applicatie kijken en eraan werken terwijl het gebouwd en getest wordt. Deze samenwerking en eenheid is net zo belangrijk als de korte time-to-market en de hoge return on investment (ROI). Deze eenheid zorgt voor een revolutie in de manier van softwareontwikkeling. Dit zorgt voor waarde in organisaties.

Conclusie: een revolutie in de evolutie

Volgens John Bratincevic, een senior analyst bij Forrester, moet software een expressie van je bedrijf zijn voordat je jezelf echt een digitale organisatie kunt noemen. Je kunt dit label niet meer plakken als je alleen wat digitale processen hebt lopen. Alles van je bedrijf, dus je cultuur, je beleid, je data; alles moet worden omgezet in software en moet mee veranderen met jouw bedrijf. Dat is wat digitaal zijn echt inhoudt.

Citizen developers zorgen ervoor dat de manier van denken binnen je organisatie verandert. Door citizen developers zal je business worden omgezet in software en zal je een echt digitale organisatie worden. No code software is de evolutionaire ontwikkeling die de revolutionaire beweging van citizen development mogelijk maakt.

Digitalisering met no code software: een revolutie in de evolutie.

Low code helpt om competitief en vernieuwend te blijven

Bron: Conclusion Low Code Company | [lowcodecompany.nl](https://www.lowcodecompany.nl)

Wereldwijd onderzoek in opdracht van Mendix laat zien dat veel bedrijven last hebben van een tekort aan softwareontwikkelaars en van achterblijvende financiering van IT initiatieven. Hierdoor gaan veel business managers zelf op zoek naar een oplossing voor hun problemen.

95% van de beslissers gaf aan dat het betrekken van IT in belangrijke strategische initiatieven een belangrijke meerwaarde levert. 70% vindt dat de IT organisatie waarde toevoegt. Aan de andere kant geeft 50% van de IT managers aan dat hun budgetten te laag zijn, tegenover 32% van de business stakeholders. IT managers (59%) maken zich zorgen over het onderhoud van legacy systemen. Dat is bijna twee keer zoveel als business managers zich hier zorgen over maken (31%). Daarentegen geeft 61% van de business managers aan dat IT maar de helft van hun ideeën implementeert.

Dat laatste zorgt ervoor dat de business meer en meer z'n eigen gang gaat om concurrerend te kunnen blijven. 69% van de business managers denkt dat het goed is om buiten IT om te acteren. Terwijl 91% van de IT managers aangeeft dat het gevaarlijk is wanneer applicaties worden ontwikkeld zonder dat men de impact van die applicaties (en het onderhoud) helemaal begrijpt.

Werken met low code kan een deel van deze problemen ondervangen. De ondervraagden zien vooral de integratie in de backend systemen als een belangrijk voordeel van low code. Daarnaast geeft meer dan 50% aan dat werken met low code platforms snel is en men verwacht veel van de eenvoudige integratie met nieuwe technologie als AI, IoT en ML die low code mogelijk maakt.

Standaard, maatwerk of low code?

Bron: Software voor jou | softwarevoorjou.nl

Tot enkele jaren geleden had je bij de keuze voor het ontwikkelen van software twee opties: standaardsoftware aanschaffen of maatwerksoftware laten ontwikkelen. In de keuze voor standaard of maatwerk koos het mkb in het verleden vaak voor de standaardoplossing, omdat maatwerk al snel te duur was. Maar waarom zou je kiezen voor een standaardoplossing als er ook geen ‘standaard’ ondernemingen zijn? Er is goed nieuws, want dankzij nieuwe ontwikkeltechnieken is een maatwerkoplossing tegenwoordig ook voor het mkb bereikbaar.

Hoge advieskosten bij standaardsoftware

Bij de keuze voor een standaardsoftwarepakket kwamen er vaker wel dan niet hoge advieskosten bij. Die extra advieskosten werden gebruikt om de standaardsoftware zodanig in te richten dat de unieke bedrijfsprocessen hiermee ook echt kunnen worden ondersteund. Of de unieke bedrijfsprocessen zelf zouden moeten worden aangepast wat resulteert in het opleiden van je medewerkers hoe te werken met de veranderde processen.

Maar wat gebeurt er als de markt of het bedrijf zelf verandert? Dan begint het hele keuzeproces weer opnieuw. Op deze manier kan je als bedrijf niet snel inspelen op de laatste trends op de markt. Het is dus belangrijk dat je kiest voor software dat flexibel en snel met je mee kan bewegen op de steeds veranderende markt.

Maatwerk bereikbaar voor iedereen

Dankzij een aantal ontwikkelingen in de wereld van software is (semi)-maatwerk voor bijna alle ondernemers bereikbaar geworden. Door die nieuwe ontwikkelingen is het mogelijk om snel, toegankelijk en tegen niet al

te hoge kosten maatwerksoftware te ontwikkelen die jouw bedrijfsprocessen ondersteunen, en niet andersom wat bij standaardsoftware het geval is.

Eén van de factoren waarom maatwerktrajecten voor meer ondernemers bereikbaar zijn geworden is het gebruik van de cloud. Dankzij cloud technologieën wordt het voor de ontwikkelaars steeds eenvoudiger om de maatwerksoftware naadloos aan te laten sluiten bij jullie unieke bedrijfsprocessen.

Low code en no code development

(Semi)maatwerk software wordt steeds vaker ontwikkeld met low code en af en toe zelfs met no code ontwikkelplatforms. Ontwikkelaars die ontwikkelen met behulp van dergelijke platforms worden ‘citizen developers’ genoemd.

Met een low code of no code platform wordt software visueel ontwikkelt met al bestaande ‘blokken’ software of halffabrikaten. Ontwikkelen op deze manier levert maatwerk, maar omdat de ‘blokken’ software al ontwikkelt zijn is het minder kostbaar dan het ontwikkelen van scratch.

Serverless computing

Naast het ontwikkelen op basis van een low code of no code platform is er nog een andere ontwikkeling om maatwerksoftware bereikbaar te maken voor een groter publiek, namelijk serverless computing. Ondanks dat hier nog steeds geprogrammeerd wordt met code hoeft de softwareontwikkelaar geen rekening meer te houden met de server beheertaken.

Het komt erop neer dat de ontwikkelaar alleen gebruikmaakt van ‘rekenkracht’ die door cloudprovider ongemerkt ergens wordt ondergebracht. Je betaalt dus alleen voor de rekenkracht die je gebruikt. In plaats van zelf losse externe modules te ontwikkelen kunnen we gebruik maken van serverless functions. Bij complexere situaties zorgt het voor sneller en goedkoper ontwikkelde software.

Maatwerk wordt de standaard

Op basis van low code, no code en serverless computing wordt maatwerk de standaard. Op basis van deze ontwikkeltechnieken krijg je snel ontwikkelde maatwerksoftware dat ook nog betaalbaar is. De meerwaarde van maatwerk betaalt zich op die manier snel uit. Moet je de keuze maken voor nieuwe software denk dan eerst goed na voordat je een of ander duur ERP systeem of CRM systeem aanschaft.

No code software als aanvulling op je standaardsysteem

Bron: Kotuur | [kotuur.com](https://www.kotuur.com)

Je hoort steeds meer geluiden om je heen over no code software: 'het kan zoveel en is zo eenvoudig!' Gaan voor no code of niet kan echter als een grote keuze voelen. De keuze lijkt vaak in te houden dat je al je huidige standaardsoftware eruit moet doen om met no code aan de slag te gaan. Maar dit is niet per sé het geval. No code is, zoals veel mensen denken, geen vervanging voor developers of data scientists. Het kan juist dienen als aanvulling of uitbreiding van je huidige interne capaciteiten en functionaliteiten. Dit kun je op een snelle en flexibele manier doen.

Waarom standaardsoftware tekortschiet

Standaardsoftware is over het algemeen vrij goedkoop en het bevat veel functionaliteit. Toch schiet het regelmatig tekort binnen bestaande bedrijfsprocessen, want:

- Zo'n 20 procent van bedrijfsprocessen zijn niet standaard, maar uniek per bedrijf. Zaken die overheden moeten aanpakken, kunnen zij soms niet ontwikkelen binnen hun standaardsystemen. Deze zijn hiervoor niet adaptief genoeg.
- Ook veranderen wet- en regelgeving weleens, waarop overheden dan vlot moeten inspelen met behulp van hun software.
- Daarnaast komen ook binnen bedrijven en overheidsorganisaties interne reorganisaties voor. Andere teams moeten betrokken worden bij processen of processen gaan zelf anders lopen. Standaardsoftware is hier door het gebrek aan flexibiliteit niet tegen bestand.

Waarom no code software als aanvulling?

Om dit op te lossen kun je dure en tijdrovende software op maat (laten) ontwikkelen, maar je kunt misschien nog wel beter no code software gebruiken om dat kleine beetje maatwerk te ontwikkelen die het specifieke probleem oplost. Met no code is dit namelijk in een mum van tijd gedaan. En een stuk goedkoper!

De snelheid met no code wordt geïllustreerd door de mogelijkheid voor rapid application development (RAD), een agile manier van werken waarbij je meteen begint met prototypen en tussendoor steeds meer feedback verzamelt. Binnen een no code platform heb je namelijk al gebouwde componenten die je eenvoudig naar de juiste plek in je formulier kan slepen (drag and drop). Er is daardoor ook geen programmeerkennis meer nodig en een applicatie maken kan tot tien keer sneller (en dus goedkoper) dan bij een traditionele applicatie. Je kunt dus inderdaad in een mum van tijd (precies gezegd: binnen een week) die 20 procent maatwerk bouwen met no code!

Extra voordelen

Je kunt no code software ook werkelijk als aanvulling gebruiken op je huidige systemen doordat het goed koppelbaar is met van alles en nog wat. Interconnectiviteit via API's is een gegeven doordat het een SaaS oplossing is. Daarnaast kun je met no code niet alleen formulieren bouwen, maar door middel van regels en rollen ook volledige processen automatiseren. Zo kun je ook meteen doorpakken met de digitale transformatie van jouw organisatie!

Praktijkvoorbeeld: no code als aanvulling

Een voorbeeld: je huidige standaardsysteem werkt prima, maar deze mist bepaalde functionaliteiten, zoals een klantportaal die een brug slaat tussen je back- en frontoffice. In plaats van je hele huidige systeem te herbouwen, kun je dit klantportaal beter ontwikkelen met een no code platform en aan je huidige systeem koppelen. Herbouwen kan maanden of zelfs jaren duren, terwijl je het klantportaal in principe volgende week al klaar kunt hebben met no code software. Makkelijke keus, toch?

Waarmee kunnen wij u helpen?

Laat ons weten wat u zoekt. Wij helpen u graag.

Zoekt u meer informatie over dit onderwerp?

Bezoek dan de sectie over dit thema op de website ICTinformatiecentrum.nl en vind daar alle informatie die gratis voor u beschikbaar is.

Zoekt u meer informatie over bedrijfssoftware?

Gaat u een selectietraject van bedrijfssoftware starten, vraag dan via ICTinformatiecentrum.nl de bijbehorende box aan met alle informatie over selectie, implementatie, het aansturen van het project, oplossingen, enz.

Zoekt u een oplossing, adviseur of leverancier?

Bel, mail of chat dan even met ons. Dat is de snelste weg. Omschrijf voor welk IT vraagstuk u een oplossing zoekt en wij gaan voor u aan de slag.

Wilt u op de hoogte blijven van nieuws en ontwikkelingen?

De ICT nieuwsbrief informeert u en 14.000 andere ICT beslissers en belangstellenden wekelijks over de meest relevante ontwikkelingen. Informatie, inspiratie, eyeopeners en noodzakelijke kennis.

Zoekt u iets anders?

Onze websites spreken voor zich. Bezoek ICTinformatiecentrum.nl om uw weg te vinden in alle thema's waarover wij informatie hebben. En als we u kunnen helpen, doen we dat graag. Bel of mail ons gerust!

ICT informatiecentrum, Houten | T 030 227 21 20 | info@ictinformatiecentrum.nl



Kennispartners

De inhoud van dit e-boek is tot stand gekomen met medewerking van de onderstaande kennispartners. Heeft u een vraag aan hen of over het thema van hun bijdrage, neemt u dan gerust contact met ze op.

Appronto

Stadionstraat 15 (NAC Stadion) | 4815 NC Breda

T +31(0)85 800 00 45 | sales@appronto.nl | apronto.nl

Jump

Stationsweg 41 B | 3331 LR Zwijndrecht

T +31(0)85 060 30 01 | info@jump.nl | jump.nl

Software voor jou

Beursstraat 11 | 7551 HP Hengelo

T +31(0)85 401 76 60 | info@softwarevoorjou.nl | softwarevoorjou.nl

Quint

De Oude Molen 1 | 1184 VW Amstelveen

T +31(0)20 305 37 00 | quint@quintgroup.com | quintgroup.com

Kotuur

Meester D.U. Stikkerstraat 10-4 | 6842 CW Arnhem

T +31(0)26 365 35 60 | info@kotuur.com | kotuur.com

Conclusion Low Code Company

Herculesplein 80 | 3584 AA Utrecht

T +31(0)30 219 38 00 | info@conclusion.nl | lowcodecompany.nl
