

BEHEER

Low code biedt gemak, maar dat gemak komt ook met risico's

HOE VOORKOM JE EEN WILDGROEI AAN LOW-CODEAPPS?

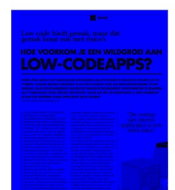
DOOR LOW CODE IS HET MAKKELIJKER GEWORDEN OM APPLICATIONS TE MAKEN EN UPDATES UIT TE VOEREN. VOORAL BINNEN BEDRIJVEN IS DIT EEN HANDIGE TOOL OM BEDRIJFSPROCESSEN TE VERSNELLEN. LOW-CODEAANBIEDERS MAKEN HET MOGELIJK BELANGRIJKE COMPONENTEN TE BEWAREN EN TE GEBRUIKEN VOOR NIEUWE APPLICATIONS. MAAR ALS HET ZO EENVOUDIG IS, HOE VOORKOM JE DAN DAT IEDEREEN MAAR APPLICATIONS GAAT MAKEN?

door Christel Verloop beeld Shutterstock

LOW CODE HEEFT ALS GROOT VOORDEEL DAT ER SNEL GEWERKT KAN WORDEN, LEGT MARTEN KRUISINGA, REGIONAL VP BENELUX AND NORDICS BIJ APPIAN, UIT. "Stel je hebt als bedrijf te maken met nieuwe wet- en regelgeving waardoor je bepaalde bedrijfsprocessen moet omgooien. Dan is het van belang dat je als bedrijf snel kunt omschakelen. Denk bijvoorbeeld aan een bank. De snelheid zit 'm in de bouwtekening van de applicatie. Als je kijkt naar de bouwtekening van applicatie A, kun je delen daarvan meenemen naar applicatie B. Hierdoor loop je als bedrijf geen vertraging op en gaat cruciale informatie niet verloren." Maar daar zit ook een risico in, denkt professor Tijs van der Storm van het Centrum Wiskunde & Informatica (CWI) en Rijksuniversiteit Groningen. "We weten vanuit softwareontwikkeling dat hergebruik heel moeilijk is. Als je heel veel componenten hebt die je kunt hergebruiken, wordt het een soort zoek-

probleem. Mensen weten dan niet meer waar ze moeten zoeken en waarop. Als je op dat punt komt, willen mensen ook helemaal geen componenten meer hergebruiken. Het wordt lastig om een component te vinden en je moet de kwaliteit van zo'n component ook maar vertrouwen. Dus dan denken ze sneller: ik doe het zelf wel." Die componenten hergebruiken kan dus enerzijds erg handig zijn en ervoor zorgen dat er veel minder uren aan besteed hoeven te worden. Daarbovenop beweegt alle belangrijke informatie netjes en makkelijk mee bij bedrijfsontwikkelingen. Toch is ook het tegenovergestelde een risico: als er nooit iets nieuws wordt gemaakt, kun je afhankelijk worden van één component, stelt Van der Storm. Raakt dat component bijvoorbeeld per ongeluk verloren, dan kan dit grote gevolgen hebben voor andere applicaties die het gebruiken. Die afhankelijkheid introduceert ook een risico: als er een probleem is met een vaak hergebruikt component, dan kan dat

'De overlap van interne applicaties is een reëel risico'



BEHEER

verregaande gevolgen hebben. Volgens Van der Storm is het dan ook belangrijk daar een middenweg in te vinden.

EEN POORTWACHTER IS CRUCIAAL

Een eventuele wildgroei kan volgens Van der Storm veroorzaakt worden door een managementprobleem. “Je moet echt een soort poortwachter hebben die dit in de gaten houdt. Zo’n poortwachter kijkt dan welke apps er zijn en als iemand een nieuwe, vergelijkbare app maakt, zegt hij dat ze te veel op elkaar lijken en dat de makers van de apps maar even met elkaar moeten gaan praten.” Die poortwachter moet wel onafhankelijk zijn, en niet bijvoorbeeld een teamlead. “Hij mag niet partijdig zijn of belang hebben bij een van de teams.” Het idee van een poortwachter wordt ook beaamd door Marten Kruisinga van Appian. Appian is een grote, internationale aanbieder van een low-codeplatform, met ook veel Nederlandse klanten. “Wat wij zien, is dat bij veel van onze klanten een soort centraal Appian-expertisecentrum bestaat dat zich ontfert over de applicaties binnen het bedrijf. Juist hier zit de kracht en snelheid in. Ik heb nog nooit een echte wildgroei gezien bij een van onze klanten, maar ik kan me voorstellen dat als iedereen over die applicaties gaat, er chaos ontstaat binnen je organisatie.”

De overlap van interne applicaties is dus een reëel risico als er geen poortwachter aanwezig is. Een andere tip die Tijs van der Storm geeft, is het instellen van bepaalde tools. “Je kunt naast zo’n poortwachter ook tools maken of gebruiken die dit soort problemen detecteren – een soort kloondetectie. Dan kun je automatisch zien of iets er al is. Dan gaat er een soort vlaggetje omhoog: deze app lijkt wel heel veel op iets wat een ander al heeft gedaan. Dan kun je dus met de ander gaan praten en samen gaan werken.” Volgens Daniël Luthra, eigenaar van appontwikkelaar Cloudmax, ligt de oplossing voor een wildgroei deels bij het doen van je inkopen. “Om te beginnen:

wees kritisch bij de inkoop van bestaande diensten, maar sta hiervoor open alvorens je besluit om voor alles zelf een oplossing te bouwen. Inventariseer wat precies nodig is en zoek naar synergie tussen applicaties. Je kunt onder andere afspreken gebruik te maken van één programma. Het is voor beheerders en gebruikers zelden productiever om met te veel applicaties te werken; low code is hierop geen uitzondering”, aldus Luthra.

LIEVER EEN WILDGROEI AAN APPS DAN AAN DATA

Kruisinga benoemt daarmee een belangrijk punt binnen deze kwestie: wat telt als te veel applicaties binnen een bedrijf? Een van Appians klanten is bijvoorbeeld een grote Nederlandse verzekeraar. “Intern hebben zij nu ongeveer 35 interne applicaties. Je kunt denken, is dat veel? Ik denk dat het goed is om niet per se te kijken naar het aantal applicaties, maar naar de functie en de inhoud. Deze verzekeraar heeft veel verschillende takken binnen de organisatie. Op het moment dat jij gebruikmaakt van tientallen applicaties binnen je bedrijf, en die ook echt nodig hebt en niet onnodige overlapping creëert, dan zie ik dat niet als wildgroei.” Mocht je uiteindelijk toch een aantal verschillende applicaties hebben die veel van dezelfde componenten bevatten, is dat dan eigenlijk zo erg? Volgens Van der Storm kan dat meevallen: “Het voordeel is dat al die apps op hetzelfde platform in dezelfde organisatie draaien, waardoor er vaak een soort interoperabiliteit is. Als je iemand hebt die dat in de gaten houdt en als een soort poortwachter werkt, dan hoeft het niet eens heel erg te zijn als er vijf verschillende apps zijn om een klant te onboarden. Als die apps maar met dezelfde data praten. Een wildgroei aan data is naar mijn mening erger dan een wildgroei aan apps.”

MOET HET NOG JAREN MEE?

Als je zo’n poortwachter inschakelt, heb je in ieder geval al veel minder kans dat



BEHEER



Wat telt als te veel applicaties binnen een bedrijf?

je het overzicht verliest binnen je bedrijf. Toch vraagt Luthra mensen na te denken voordat ze low code inzetten. “Een low-codeapplicatie is afhankelijk van het low-codeplatform waar het op gebouwd is.” Volgens hem staat de klantbehoefte centraal en is low code in dat opzicht slechts weer een nieuwe methode om in die behoefte te voorzien. “We maken bij Cloudmax beperkt gebruik van low code. We beschikken over de kennis en kunde om complete applicaties te ontwikkelen zonder de keuzebeperkingen van een low-codeplatform. Het voordeel hiervan is dat je in het

eindresultaat geen concessies hoeft te doen.” Volgens hem is low code in het bijzonder handig voor interne bedrijfsapplicaties, maar raadt Luthra aan om low code op commercieel gebied alleen te gebruiken om een nieuw idee mee te toetsen. “Met een klein budget is low code bijvoorbeeld ideaal voor het bouwen van een prototype. Na het verkrijgen van positieve respons kan men overgaan tot de ontwikkeling van een volledig product waarbij ook het intellectueel eigendom gewaarborgd is. Ik zal dus niet zeggen dat low code per definitie de voorkeur heeft.” 