

Betere dienstverlening financiële organisaties met continuous delivery

Flexibeler, efficiënter en in kort tijdsbestek software ontwikkelen

Sinds de kredietcrisis en door opkomende technologieën staan banken voor nieuwe uitdagingen. Enerzijds lopen de inkomsten terug doordat er minder spaarders zijn, persoonlijke schulden lopen op en er komen iedere dag weer honderden faillissementen bij. Door de opkomst van internet zijn er nieuwe concurrenten die specifieke diensten aanbieden en zo inkomsten wegnemen. Anderzijds zijn er steeds meer technologieën die bijvoorbeeld online en mobiel bankieren mogelijk maken, wat de banken juist kansen biedt.

Banken die zich willen onderscheiden in de markt zullen zich meer moeten richten op de dienstverlening en hierbij nieuwe technologieën optimaal benutten. Dat is precies waar continuous delivery uitkomst biedt.

De rol van IT in dienstverlening

Wat producten betreft, lijken banken veel op elkaar. Het onderscheidend vermogen zit met name in de dienstverlening, hier valt voor financiële organisaties genoeg te winnen. Klanttevredenheid wordt voor meer dan 50% bepaald door de mate van service die klanten ervaren. Deze service wordt grotendeels gedreven door proceskwaliteit en daarmee is IT een van de meest bepalende factoren in de strategie van een bank. Het is daarom belangrijk voor financiële organisaties om kritisch te kijken naar hun ontwikkelmethodes en na te gaan of de huidige situatie tot IT-oplossingen leidt met optimale functionaliteit voor de eindgebruiker.

Flexibel, efficiënter en sneller kwaliteitssoftware ontwikkelen

Continuous delivery zorgt kort gezegd voor het aanpakken van de gehele levenscyclus van software - van analyse, ontwerp en cyclisch opleveren naar test, deployment en feedback - in één end-to-end-werkwijze. De verantwoordelijkheid voor de hele levenscyclus wordt ondergebracht in gezamenlijke teams en onderscheid tussen ontwikkelen en beheer vervaagt hierbij. Alle fasen van de levenscyclus worden verregaand geautomatiseerd met bewezen, standaard tooling die in de markt beschikbaar is. De invoering van continuous delivery zal leiden tot het vergroten van de kwaliteit van IT en het reduceren van de kosten die gemoeid zijn met IT.

Een agile ontwikkelaanpak alleen volstaat niet

Organisaties kiezen steeds vaker voor flexibelere manieren om software te ontwikkelen. IT is complexer geworden en daarom volstaat de traditionele 'waterval'-methode, waarbij duidelijk afgebakende fasen stap voor stap worden doorlopen, in de meeste gevallen niet meer. Hierbij zijn de omgevingsfactoren te onzeker en is meer flexibiliteit vereist. Een logische stap voor de IT-ontwikkeling is daarom om te kiezen voor een agile en iteratieve aanpak. Een agile-aanpak zorgt ervoor dat teams de juiste zaken ontwikkelen voor de business en er snel kan worden bijgestuurd door kort cyclisch te werken. Dit heeft als grote voordeel dat het de flexibiliteit van het ontwikkelproces bevordert doordat functionaliteit in stappen wordt toegevoegd.

Het ontwikkelteam functioneert beter, omdat ze de eerste ontwikkelfasen - analyse, ontwerp, bouw - niet afzonderlijk afhandelen. Toch is hiermee de kous nog niet af. Het nadeel van deze aanpak is namelijk dat de opvolgende testfase lang kan duren en dat er tijd overheen gaat voordat de software implementatie is doorgevoerd in de productieomgeving. Beheerders hebben immers een drukke agenda en hun streven naar stabiliteit zorgt ervoor dat zij het aantal releases laag houden. De denkfout die men hierbij maakt, is dat deze aanpak zorgt voor tijdsbesparing. Aanvankelijk klopt deze redenering, maar omdat de implementatie vaak handmatig gebeurt, neemt het risico op fouten toe waardoor beheerders alsnog in de avond of het weekend doorwerken om het probleem te herstellen. De agile-ontwikkelmethode is een stap in de goede richting, maar de grotere uitdaging voor financiële organisaties is om zowel flexibeler als efficiënter en sneller software te kunnen opleveren.

Continuous delivery: een passende oplossing voor nieuwe uitdagingen

Om deze mate van flexibiliteit, efficiëntie en snelheid in softwareontwikkeling te behalen, kunnen financiële instellingen het beste overgaan op de continuous delivery-methode. Deze methode is er juist op gericht om nieuwe ideeën zo snel en efficiënt mogelijk in productie te krijgen. Ontwikkelen wordt een iteratief proces waarbij continu kleine stukken functionaliteit automatisch worden uitgerold. Hierdoor komen deploy-stappen vaker in het OTAP (Ontwikkel, Test, Acceptatie en Productie)-proces voor, waardoor software releases vaker plaatsvinden en functionaliteit in een doorlopend proces wordt getest en stap voor stap wordt verbeterd. Doordat functionaliteit vaker en sneller kan worden opgeleverd, kun je beter op de behoeften van klanten inspelen. Het automatische testen zorgt er bovendien voor dat het team eerder inzicht krijgt in de kwaliteit van applicaties. Ontwikkelteams krijgen ook sneller feedback op software van eindgebruikers. Hierdoor worden fouten in een vroeger stadium opgelost en verbeteringen eenvoudiger toegevoegd. Dit komt de dienstverlening naar de klant toe direct ten goede.

Realiseren van continuous delivery

Veel banken hebben baat bij het flexibeler, efficiënter en sneller ontwikkelen van software. Maar hoe realiseer je de continuous delivery? Het is belangrijk dat mensen, processen en tools goed zijn afgestemd. Ontwikkelaars, testers en beheerders werken intensief samen om in iteraties - de miniprojecten binnen een softwarepakket - functionaliteit van hoge kwaliteit op te leveren. Hierbij is niet alleen goede communicatie belangrijk, maar ook het volgen van de agile- en iteratieve processen. Verder moet precies duidelijk zijn hoe het uitrolproces opgezet is.

Verandering van processen noodzakelijk

IT-afdelingen die volgens de watervalmethode werken, zullen de verschillende fasen - zoals ontwikkelen, testen, acceptatie en securityafstemming - stap voor stap en afzonderlijk uitvoeren. Bij continuous delivery daarentegen is het belangrijk dat ontwikkelaars, testers en andere IT'ers nauwer samenwerken en zoveel mogelijk taken automatiseren zodat snel kan worden overgegaan naar productie. Om continuous delivery succesvol te realiseren, is het noodzakelijk om gebruik te maken van tools die

dit proces ondersteunen. Zo is het belangrijk om een deployment pipeline samen te stellen. Deze pipeline zorgt ervoor dat programmering in de ontwikkelomgeving snel overgaat naar functionaliteit en dat deze snel in productie kan worden gebracht.

Vaardigheden van ontwikkelteam

Naast het veranderen van werkprocessen vraagt de transformatie ook een andere rol van de teamleden. Enkele belangrijke vaardigheden zijn vereist. Zo is het nodig dat ontwikkelaars en beheerders meer communiceren en samenwerken. Het is essentieel dat de teamleden elkaar door en door begrijpen om samen tot het gewenste resultaat te komen. Dit kan wel eens lastig zijn, aangezien ontwikkelaars en beheerders traditioneel gezien andere belangen hebben. Een ontwikkelaar wil zo snel en zoveel mogelijk functionaliteit naar productie brengen. Een beheerder daarentegen zoekt juist naar stabiliteit en zo min mogelijk veranderingen. Vanuit de continuous delivery-aanpak streven beiden hetzelfde doel na, namelijk het opleveren van kwalitatief goede functionaliteit waar de klant direct iets aan heeft. Goede communicatie en nauwe samenwerking is nodig om focus op kwaliteit te behouden en het ontwikkelproces efficiënt te laten verlopen.

Herstructureren van IT-afdeling

Continuous delivery vraagt ook om een herinrichting van de IT-afdeling. De verschillende fasen in het ontwikkelproces moeten niet langer worden gezien als afzonderlijke onderdelen en ontwikkelaars en beheerders moeten meer als één team samenwerken. Bovendien zal met name de rol van de beheerder aanzienlijk veranderen. Deze is met continuous delivery minder tijd kwijt aan het uitrollen van de software en er is meer tijd nodig voor samenwerking met het team, de automatisering van zoveel mogelijk taken en het verder verbeteren van het deployment-proces.

Door de veranderende rollen is niet iedereen even enthousiast over continuous delivery. Beheerders kunnen denken dat ze overbodig worden. En teams die zich volledig richten op het testen van software passen niet meer in de aanpak. Met continuous delivery zijn multidisciplinaire teams nodig die deze verschillende onderdelen uit ontwikkeltrajecten in nauwe samenwerking en in een iteratief proces doorlopen. Het automatiseren van terugkerende taken speelt een belangrijke rol. Enerzijds om de kans op fouten door handmatig werken te verminderen. Anderzijds kunnen teamleden meer tijd steken in samenwerking en kwaliteitsbewaking en tegelijkertijd sneller en efficiënter werken.

Hoe kan Info Support helpen?

Hoewel de voordelen van continuous delivery aantrekkelijk zijn, vinden veel organisaties het lastig om op eigen kracht over te stappen. Toch is er geen reden om dit niet te doen. Technische ondersteuning is vereist en het is raadzaam om de transformatie goed te begeleiden met de juiste kennis en expertise. Info Support is specialist in het ontwikkelen, beheren en hosten van software en helpt banken en andere organisaties om de implementatie van continuous delivery te starten of voort te zetten. Zo kan iedere financiële organisatie gebruik maken van de voordelen van continuous delivery. Op

aanvraag inventariseren zij in hoeverre de continuous delivery-methode al past binnen de organisatie. Vervolgens geven zij advies over welke verdere stappen eventueel nodig zijn om flexibel, efficiënt en in een kort tijdsbestek software van hoge kwaliteit te kunnen ontwikkelen. Bovendien beschikt Info Support over gespecialiseerde teams die al op basis van continuous delivery werken. Deze teams kunnen ingehuurd worden om als voorbeeld te fungeren. Eigen aanwezige IT-teams kunnen zo nieuwe technieken eenvoudig bestuderen en ontwikkeling middels continuous delivery stapsgewijs overnemen.

Direct aan de slag met continuous delivery

Om continuous delivery te realiseren, is het nodig veranderingen aan te brengen in zowel processen als structuur van de afdeling. Hieronder verschillende tips die financiële organisaties kunnen helpen om te starten met continuous delivery:

- **Creëer een haalbaar, betrouwbaar proces voor het releasen van software:** Kenmerken van een goed release-proces zijn dat kleine delen nieuwe functionaliteit eenvoudig uit te rollen zijn en dat eventuele fouten direct te koppelen zijn aan een bepaalde release. Fouten zijn snel herleidbaar en nieuwe functionaliteit kan sneller worden uitgerold.
- **Automatiseer zoveel mogelijk:** Hoe meer taken geautomatiseerd zijn, hoe minder fouten optreden door menselijk handelen en hoe sneller stappen uitgevoerd kunnen worden.
- **Bewaar alles in versiebeheer:** Het is belangrijk dat software reproduceerbaar is. Door code, deploy-scripts, en liefst ook scripts om de omgeving in te richten in versiebeheer te plaatsen, is het altijd mogelijk om naar een bepaalde situatie terug te gaan. Als problemen worden geconstateerd, kan direct een oudere versie worden uitgerold. Zo kan een probleem in de nieuwe versie worden opgelost zonder dat de eindgebruiker daar last van heeft.
- **Stel releases van probleemsoftware niet uit:** Iedere organisatie heeft problemen met bepaalde software. Stel de releases van deze probleemsoftware niet uit maar release juist zo vaak mogelijk. Maak knelpunten zichtbaar en los deze snel op.
- **Bouw kwaliteit in:** Bij de traditionele waterval-methode wordt software uitvoerig getest. Bij continuous delivery gebeurt dit op continue basis. Om tijd te besparen, dienen testen zoveel mogelijk geautomatiseerd te worden. Zo is in korte tijd te zien of een applicatie correct werkt. Het is belangrijk om automatische unit-testen en functionele testen te maken en automatisch te controleren op de code kwaliteit.
- **Eindgebruiker staat centraal:** Functionaliteit wordt gebouwd voor de klanten. Het ontwikkelproces is dus pas afgerond als het op productie staat en de eindgebruiker er succesvol mee kan werken.
- **Het team is verantwoordelijk voor het opleverproces:** Beheerders, ontwikkelaars, testers en andere teamleden zijn samen verantwoordelijk om zo goed mogelijke functionaliteit op te leveren. Goede samenwerking en afstemming is daarom essentieel.
- **Continue verbetering:** Hoe goed het team ook is, er is altijd ruimte voor verbetering. Het is belangrijk dat teams alert blijven om verbeterpunten te signaleren. De retrospectie is een goed moment

hiervoor. Dit helpt om steeds meer onderdelen te automatiseren of op een andere manier te verbeteren zodat in steeds kortere tijd, efficiëntere software van toenemende kwaliteit kan worden opgeleverd.

Geschreven door Johan Janssen en Dennis Joosten, Info Support