

# Hoe je succesvol een platform ontwikkelt

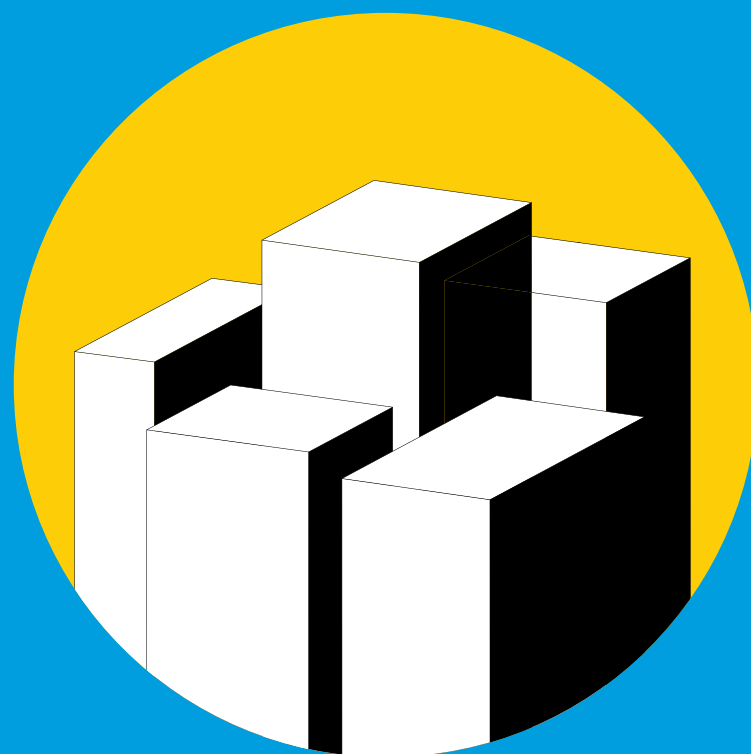
21 tips over DX & UX



# Inhoud

<b>4</b>	Platformen, Developer Experience en User Experience
<b>10</b>	Welkom in de platformrevolutie!
<b>12</b>	Platformopenheid
12	Benut openheid om met developers je platform te laten vliegen
15	Case: Integratieplatform voor betaalmethoden
18	De gebruiker centraal op het platform
<b>19</b>	Positieve netwerkeffecten
20	Het zijn gebruikers en developers die een platform groot maken
22	User centricity met 'Give it to me now!'
23	User Experience voor multi-level rollen op een platform
24	Case: Energy trading platform
<b>27</b>	Schaalbaarheid
27	Developers de gelegenheid bieden om schaalbaarheid te realiseren
28	Performance optimalisatie via User Experience design
31	Case: Nieuw UX design voor complex financieel product
<b>35</b>	De conclusie
<b>36</b>	Checklist voor een fantastische User én Developer Experience
<b>38</b>	Over de experts
<b>39</b>	Over ISAAC

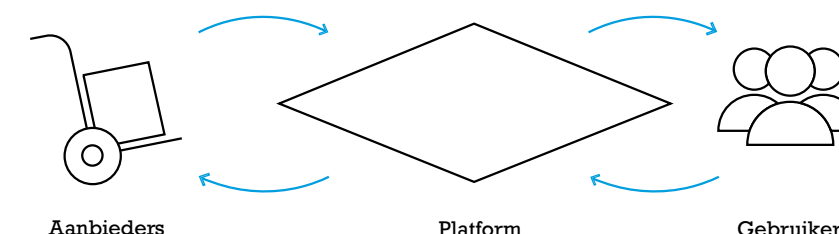
# Platformen, Developer Experience en User Experience



De begrippen platformen, User Experience en Developer Experience zijn veelvoorkomende termen, maar worden nogal eens op verschillende manier uitgelegd. Voorafgaand aan dit whitepaper vertellen we daarom wat wij onder deze begrippen verstaan.

## Platformen

Google, Facebook en Amazon zijn platformgiganten van de 21e eeuw. Platformen brengen gebruikers, vraag en aanbod, bij elkaar om diensten te gebruiken, te verlenen of transacties te doen. In een traditioneel businessmodel 'pusht' een bedrijf goederen of diensten in een pijplijn naar de eindklant. Een platform daarentegen stelt resources ter beschikking voor gebruikers en aanbieders. Een platform matcht gebruikers en aanbieders en faciliteert transacties.



De sleutel om daar een succes van te maken, zit in aantrekkelijk zijn voor de juiste gebruikers en hen verleiden tot gebruik van het platform.

## Basisprincipes van een platform

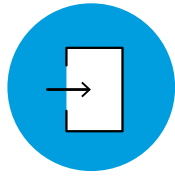
Concurrentievoordeel en onderscheidend vermogen gaan in dit digitale tijdperk over iets anders dan voorheen. Tegenwoordig gaat het veel meer over welk bedrijf in staat is om de mogelijkheden van technologie maximaal en relevant te benutten. Wij zien vier kernfactoren die invloed hebben op platformsucces:



### 1. 'Give it to me now!'

User centricity staat centraal in het businessmodel van succesvolle platformen. Een platform is on demand. Gebruikers kunnen services 'op afroep' gebruiken en centraal staat de beleving die gebruikers op het platform ervaren. Platformen die zich onderscheiden, zijn daarom volledig gefocust op de taak of het doel van gebruikers.

Het platform is als een stuk gereedschap: het moet gebruikers ondersteunen in de 'job at hand' en niet in de weg zitten of zelfs tegenwerken. Onmiddellijke tevredenheid van gebruikers in alle mogelijke user flows die je als platform wil ondersteunen, is daarom heel belangrijk. Voldoende 'feedback' genereren met een gedegen meetplan is cruciaal om die optimale gebruikerservaring mogelijk te maken, maar ook door te ontwikkelen wanneer behoeften veranderen.



## 2. 'Open' is de nieuwe standaard

Een platform is eigenlijk een digitale ontmoetingsplek die twee kanten van een markt matcht: vraag en aanbod. Dat moet zo simpel mogelijk gebeuren. Een succesvol platform is daarom onder andere beschikbaar op de preferred devices van gebruikers, heeft de stappen van onboarding tot het absolute minimum

teruggebracht en is zo open mogelijk. En dankzij deze openheid is een platform dus ook voor producenten en developers eenvoudig toegankelijk via (API-)gateways en developer portals.



## 3. Positieve netwerkeffecten

Succesvolle platformen zorgen ervoor dat een platform aantrekkelijk is voor gebruikers en voor producenten. Ze weten snel de juiste hoeveelheid massa op te bouwen en te profiteren van een positief sneeuwbaaleffect. Doordat gebruikers graag op het platform actief zijn, wordt een groeicyclus gecreëerd die ook weer nieuwe

gebruikers aantrekt. Bedrijven met een succesvol platform zijn bovendien op relevante plekken zichtbaar voor gebruikers en platformproducenten, bijvoorbeeld developers. Ze weten wat ze op het gebied van customer experience voor gebruikers en developers moeten doen om vertrouwen te krijgen. En ze weten via dit soort kritische succesfactoren uiteindelijk positieve netwerkeffecten te realiseren: een zichzelf versterkend effect naar meer gebruikers en dus ook weer meer waarde.



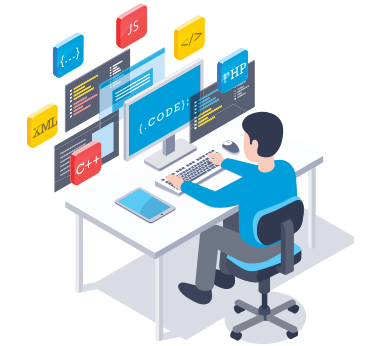
## 4. Schaalbaarheid

Platformen zijn er natuurlijk in alle maten, van enorm, voor iedereen en wereldwijd, tot specifiek en B2B voor een nichemarkt. Ze delen echter allemaal de noodzaak tot betrekkelijk soepel schalen. Dat komt zowel terug in techniek als processen: de software wordt zo opgezet dat er behoorlijk wat gebruikers toegevoegd kunnen

worden zonder drastische aanpassingen, en processen zijn zo veel mogelijk 'lean' en geautomatiseerd. De curve van waardegroei gaat harder omhoog dan die van de operationele kosten. En juist bij groei en schalen geldt: meten is weten. Datagedreven doorontwikkelen is daarom de weg naar meer succes. Schaalbare technologie en de al genoemde openheid richting betrokkenheid van developers zijn daarbij belangrijke middelen.

## Developer Experience (DX)

Bij veel bedrijven hebben developers lange tijd een rol op de achtergrond gespeeld. Vaak als de beheerders van bestaande technologieprojecten of als de medewerkers waaraan websiterevisies of nieuwe (mobiele) apps worden uitbesteed. In onze open, connected maatschappij is dat logischerwijs niet meer van deze tijd. Zeker omdat in een platformstrategie in toenemende mate juist ook externe developers key bijdragers zijn.



Developers zijn de mensen die de digitale ervaringen bouwen waarmee veel bedrijven uiteindelijk het geld verdienen. Zij zijn op een platform, naast gebruikers, een aparte doelgroep. Developers zorgen er als platformproducenten en -consumenten voor dat een platform op schaal kan opereren, en zo (nog) aantrekkelijker en succesvoller kan worden. Interesse en betrokkenheid van die groep is in een platformstrategie daarom topprioriteit. Wat trouwens niet per se eenvoudig is in een tijd waarin developers één van de meest bedrijfsbeperkende resources zijn. Een platform succesvol draaien, vraagt daarom een geweldige ervaring voor developers.

Op platformen zijn niet alleen APIs, Software Development Kits (SDKs) en plug-ins nodig voor eenvoudige integratie van services, maar worden die elementen idealiter ook echt behandeld als producten. Met een platform verschuift namelijk de focus van APIs met een puur interngerichte middlewarefunctionaliteit, naar APIs als waardevolle producten die in staat zijn businessmodellen succesvol te ontsluiten, zowel intern als extern. Zelfs als het businessmodel bijvoorbeeld gericht is op geld verdienen uit consumenteninteracties, zijn externe developers vaak ook betrokken in die strategie. Zij bouwen of dragen bij aan die digitale interacties met consumenten. Developers verzorgen, vanuit welke organisatie dan ook, de integratie.

**Developer Experience focust op gemak, toegankelijkheid, codekwaliteit, documentatie en mogelijkheden tot veilig experimenteren van open, toegankelijke software.**

Een platform biedt betrouwbare en eenvoudig te doorgronden componenten zoals APIs, SDKs en (platform specifieke) plug-ins. Vragen die je steeds moet blijven stellen bij het 'challengen' van de status quo zijn:

- 1 Hoe makkelijk is het om tegen (tools van) het platform aan te bouwen
- 2 Hoe toegankelijk is het platform qua techniek
- 3 Wat voor informatie geven we over de beschikbare tooling
- 4 Hoe meten we de totaalervaring van developers wanneer ze interacteren met een platform?

## User Experience (UX)

Het succes van een platform valt of staat met de gebruikerservaring. Is het platform waardevol? Levert het de gebruiker waarnaar hij op zoek was? Bereiken gebruikers hun doel of kunnen ze succesvol een taak afronden? Gebruikers stel je tevreden wanneer het ontwerp bijdraagt aan de taken en doelen die ze via het platform willen afhandelen. Het uitgangspunt van User Experience is dan ook de gebruiker, mensen die taken en doelen hebben, en hun gedrag.



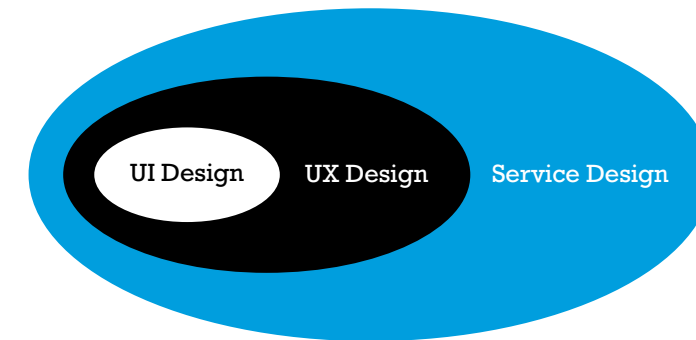
**In essentie gaat User Experience over de totaalervaring van gebruikers met het merk, dienst, product en platform.**

In het ontwerpproces voor die totaalervaring zit gelaagdheid. Aan de basis ligt de strategie, waarin doelstellingen en gebruikerskenmerken en -inzichten bij elkaar komen. Met tools en methoden uit Design Thinking ga je in de eerste fase aan de slag om gebruikers nauwkeurig in kaart te brengen. Denk daarbij aan methodieken als gebruikersonderzoek, interviews, service blue prints, customer journey mapping en concurrentieanalyse. De strategie die uit de eerste fase volgt, vormt de kapstok. Die kapstok kun je er overigens in elke iteratie ook weer bij pakken om gebruikersdata te interpreteren en richting te geven aan wijzigingen en verbeteringen. De kennis uit het onderzoek naar gebruikers kun je vervolgens gebruiken om functionaliteiten te kiezen en specifieke elementen vorm te geven.



Voordat je daadwerkelijk iets gaat bouwen, is het verstandig om een stap in te bouwen waarin je prototypes en concepten test bij gebruikers. Je hebt immers wel controle over wat je ontwerpt, maar niet over hoe gebruikers dat uiteindelijk gebruiken. Door te toetsen aan gebruikers ontstaat een gevalideerd startpunt voordat het bouwen begint. Onthoud wel: het is een startpunt en niet een eindstation. De User Experience op een platform vraagt daarom om een gedegen meetplan, waarbij je continu feedback ophaalt en inzichten uit gebruikersdata gebruikt om te verbeteren. Continu kleine wijzigingen uitrollen, meten, leren en opnieuw naar 'productie' rollen is in het ontwerp van een fantastische UX cruciaal.

Als we iets verder inzoomen op de gelaagdheid van User Experience, dan is het belangrijk om iets te weten over het verschil tussen User Experience (UX) Design en User Interaction (UI) Design. Beide typen design hebben hun eigen kenmerken en kwaliteiten, maar de één kan niet zonder de ander. Maar wat wil dat zeggen?



De UI beslaat het directe contact tussen gebruikers en het platform. Het UI Design zit aan de voorkant en is zichtbaar op het scherm van de gebruiker. Denk daarbij aan zaken als kleuren, lettertypes, grafische elementen, knoppen, (micro)animaties of lettergrootte. Dit soort kenmerken moeten dus ook wel passen bij wat jouw merk wil uitstralen. Effectiviteit van een UI Design wordt vaak redelijk eenvoudig getest met A/B-testen. Levert bijvoorbeeld de gele knop meer interactie of conversie op dan de blauwe? En helpt een animatie de gebruiker om te begrijpen wat zijn actie voor effect had?

In tegenstelling tot UI Design zit UX Design in de processen die achter de schermen plaatsvinden. UX Design speelt zich af op het niveau van functionaliteiten, structuren, customer journeys en user flows. Het gaat vooral over de vraag: hoe ontwerp ik processen en flows die ervoor zorgen dat gebruikers op het platform taken en doelen eenvoudig af kunnen handelen? Een goed UX Design valt of staat dus met een goede user flow, een digitaal wandelpad dat is gebaseerd op wensen en behoeften van gebruikers. Een goed digitaal wandelpad maakt de interactie van gebruikers met het platform praktisch en relevant.

En hoewel een doeltreffende ervaring voor gebruikers veel waarde biedt, brengt het voor bedrijven ook op verschillende manieren meerwaarde. Goede UX straalt rechtstreeks positief af op je corporate brand. Zo kan er bijvoorbeeld extra geld worden verdiend doordat gebruikers het platform zullen aanraden aan anderen, maar ook omdat gebruikers eerder bereid zijn om voor functionaliteiten op het platform te betalen. Daarentegen levert het ook besparingen op. Met nuttige gebruikersinzichten wordt geld bespaard op (door)ontwikkeling, want er wordt immers niets meer ontwikkeld waar gebruikers geen behoefte aan hebben.

Hoe dan ook, een UI zonder UX is als een winkel waar je moeite hebt om de producten en kassa te vinden, terwijl een UX zonder UI een leeg winkelpand is waar je geen product of kassa kunt vinden. In de ontwikkeling van een degelijk platform kunnen Design Thinking, UX en UI daarom niet ontbreken ●

# Welkom in de platformrevolutie!

Facebook is een mediabedrijf, maar maakt geen content. Uber is een taxibedrijf, maar heeft zelf geen chauffeurs of taxi's. Airbnb heeft de hotelindustrie op zijn kop gezet, maar bezit zelf geen enkele kamer of appartement. En SnappCar verhuurt auto's, maar bezit zelf geen wagenpark. Allemaal sprekende voorbeelden van platformen die de afgelopen jaren hard zijn gegroeid en waar gebruikers graag diensten gebruiken en transacties doen.

De gemene deler in deze voorbeelden? Het platform maakt transacties mogelijk en brengt gebruikers en producten bij elkaar. Met als belangrijkste factor voor succes: aantrekkelijk zijn voor de juiste gebruikers en hen verleiden het platform te gebruiken. Platformen als Airbnb, Uber en SnappCar, bieden toegevoegde waarde door een probleem op te lossen, gemak te bieden of nieuwe dienstverlening te introduceren. En dat doen ze met een ijzersterke User Experience en Developer Experience.

Platformen hebben een enorme potentie. En nu, in de tijd dat technologie heel maakbaar is en bijna oneindige mogelijkheden biedt, is een veelgehoorde vraag: hoe ga je concreet met een platform aan de slag? Om succes te boeken, moet je jezelf een aantal vragen stellen. Onder andere:

- 1 Wat is het probleem dat ik voor gebruikers en aanbieders oplos?
- 2 Hoe ga ik groeien? Sales op een platform betekent onder meer 'verkoop' aan twee kanten tegelijk: vraag en aanbod.

- 3 Is mijn platform experience de winnaar ten opzichte van de concurrent? Voor gebruikers maakt het immers niet zo veel uit of diensten worden aangeboden via aan platform of niet. Zij gaan voor de beste ervaring. Wat onderscheidt de ervaring op jouw platform van de concurrentie?

Hoewel de eerste twee vragen natuurlijk aan de basis van een degelijke platformstrategie liggen, staat in dit whitepaper juist de derde vraag centraal:

**Hoe maak je effectief de stap van een platformstrategie en de voornemens van een 'open' businessmodel naar een winnende User Experience en Developer Experience?**

In dit whitepaper bieden we 21 praktische handvatten voor de uitwerking van UX en DX voor een krachtig digitaal platform. Wat zijn typische kenmerken? Hoe verhouden die zich tot centrale thema's in een platformstrategie: openheid, positieve netwerkeffecten en schaalbaarheid?

De checklist is op bladzijde 36 te vinden →



# Platformopenheid

Een platform functioneert als digitale ontmoetingsplek, waar vraag en aanbod bij elkaar worden gebracht. Hoewel daarvoor verschillende verschijningsvormen denkbaar zijn, zijn platformen op hoofdlijnen in te delen in twee typen: (1) open end-to-end modellen, waarbij platformen aan twee zijden open zijn voor gebruikers en producenten en (2) eenzijdige platformen die alleen voor gebruikers open zijn.

Als we het hebben over platformmodellen die aan twee zijden open zijn, dan kennen we bekende voorbeelden als Google, Facebook en Airbnb. Veel bedrijven denken dat het een utopie is om ook met zo'n soort platform te opereren. Maar opereren op die schaal is niet nodig om succesvol een platform te kunnen runnen. Lang niet ieder bedrijf concurreert immers op een (wereldwijde) consumentenmarkt. Veel bedrijven zijn actief in een industry vertical. Ook juist in die context kun je met een krachtig digitaal, datagedreven platform succesvol zijn.

Om met een platform in een industry vertical voet aan de grond te krijgen, draait het om snel de kritieke massa in de vertical te bereiken. Om dat voor elkaar te krijgen, moet je platformgebruikers weten te overtuigen met een fantastische User Experience en Developer Experience. Maar hoe richt je dat goed in?

## Benut openheid om met developers je platform te laten vliegen

Openheid benutten om samen met developers een platform te laten vliegen, betekent niets minder dan het platform niet te veel dichttimmeren. Je wilt deze doelgroep minimaal de mogelijkheid bieden om via gateways en/of developer portals eenvoudig te interacteren met het platform. Daarbij geldt onverminderd dat gebruik van het platform natuurlijk ook waarde moet bieden, bijvoorbeeld door de

beschikbaarheid van specifieke domeinkennis en resources van jouw bedrijf.

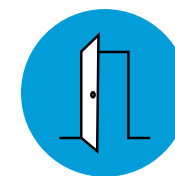
Vanuit een outside-in gedachte kan een developer portal het beste worden neergezet als one-stop shop waar developers producten zoals APIs, plug-ins en Software Development Kits (SDKs) kunnen vinden, er toegang toe hebben, erover kunnen lezen en leren, maar er ook mee kunnen experimenteren. De mate van openheid van het platform – een grens die bedrijven overigens zelf kunnen opschuiven of juist terugschroeven – beïnvloedt uiteindelijk ook de creativiteit van de gebruikers. Hoe opener het platform is, hoe meer ruimte je developers geeft om beschikbare tooling breed toepasbaar in te zetten. Maar daarmee dus ook de mogelijkheid biedt om het in te zetten voor use cases waaraan je misschien zelf niet direct had gedacht.

In deze context vijf tips voor een optimale Developer Experience:



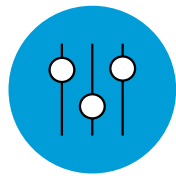
### 1. Wees explorable

Maak het makkelijk om een gevoel te krijgen bij het systeem en de resources die beschikbaar zijn via het platform. Zorg er bijvoorbeeld voor dat developers daarmee kunnen experimenteren. Zeker als zij voor het eerst gebruikmaken van het platform en ze je (nog) moeten leren kennen. Als resources dan laagdrempelig toegankelijk zijn en eenvoudig te begrijpen, zullen developers zich sneller aan jouw platform willen verbinden.



### 2. Gebruik open standaarden

Maak zo veel mogelijk gebruik van industry open standaarden. Dat maakt het voor developers makkelijk en aantrekkelijk om met tools op het platform aan de slag te gaan. Sluit bijvoorbeeld aan bij standaarden als npm, Composer, Maven, Github, Open API, GraphQL en Cocoa pods.



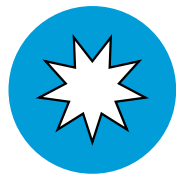
### 3. Zorg voor een goede sandbox

Maak de drempel om te experimenteren met de tooling zo laag mogelijk. Bied bijvoorbeeld een 'explore and play' sandbox aan die zich hetzelfde gedraagt als de productieomgeving van het platform, zodat developers veilig kunnen prototypen en experimenteren.



### 4. Stel goede documentatie beschikbaar

Zorg voor ruim voldoende, up-to-date documentatie voor verschillende doelgroepen, bijvoorbeeld ook voor inkopers, en maak die eenvoudig beschikbaar. Plaats documentatie bijvoorbeeld als MD-file in de Git repository of deel het via wiki's. Goede documentatie bevat, naast uitleg en informatie, ook voldoende 'cut-and-paste' voorbeelden. Ook is het belangrijk om eraan te denken informatie over zaken als versioning op te nemen. Heb je veel om uit te leggen? Investeer dan in een technical writer. Niets is namelijk zo vervelend als rammelende, slecht geschreven teksten zonder samenhang of consistentie. Kortom, zorg dat de documentatie duidelijk beschrijft wat developers wel en niet kunnen verwachten, maar bijvoorbeeld ook wat de (technische) roadmap van het platform is. Dat helpt developers anticiperen en optimaal gebruik te maken van de resources.



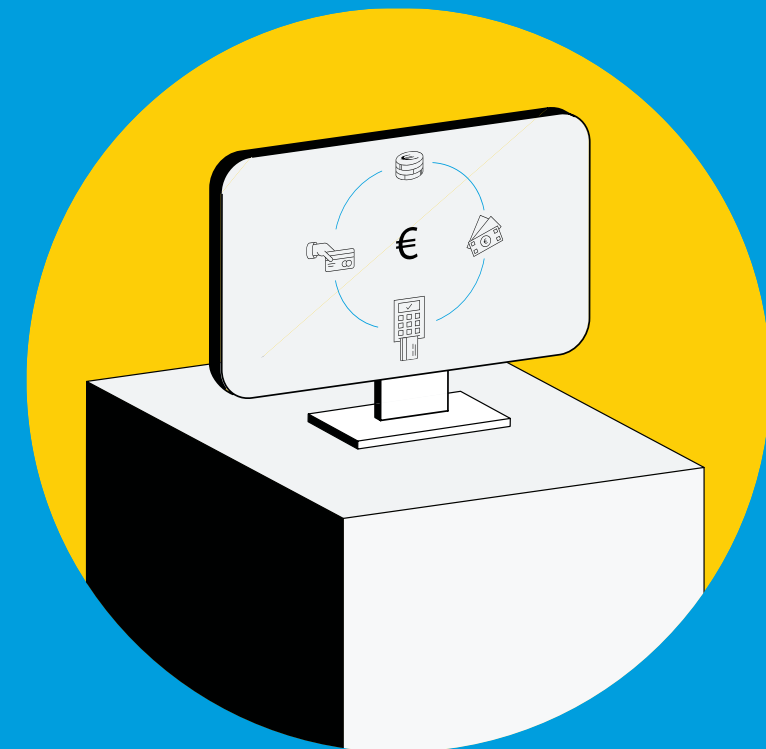
### 5. Houd rekening met nieuwe use cases

Je kan (en wilt) niet per definitie alle denkbare toepassingen van een platform van tevoren al bepalen. Laat het creatieve denkwerk over aan 'de community'. Om dat mogelijk te maken moet je logische 'extension points' aanbieden en de interne platformstatus moet zo transparant mogelijk zijn zodat developers daar ook mee aan de slag kunnen. Bied via technieken als Webhooks (voor push berichten) en GraphQL (voor query-vrijheid) flexibiliteit aan de developer-gebruikers om 'in te haken' op het platform. Met nieuwe toepassingen gaat het positieve netwerkeffect van een platform vliegen.

Case

# Integratieplatform voor betaalmethoden

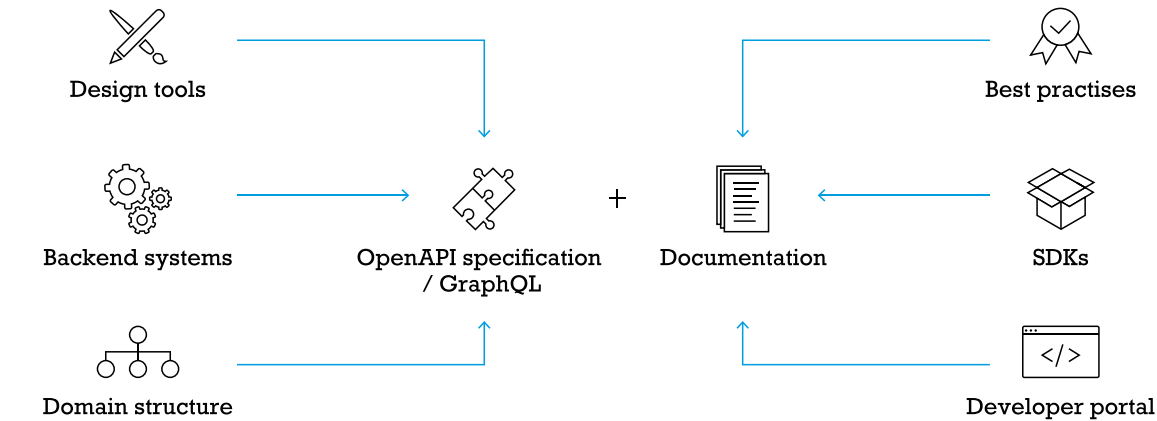
ingenico  
ePayments





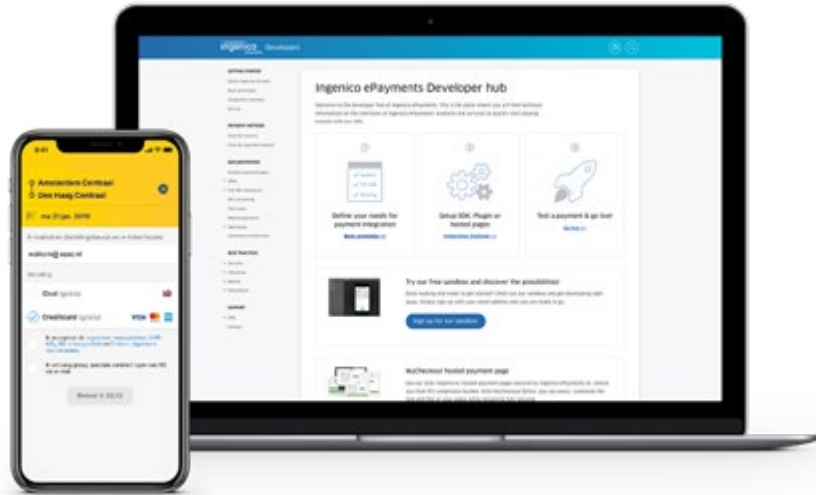
Ingenico ePayments heeft wereldwijd een rijk aanbod van betaalproducten. De systemen van Ingenico ePayments ondersteunen tokenization, fraudeonderzoek, look-up services en financiële mechanismen zoals uitbetalingen, valutaconversies en terugbetalingen voor internationale merchants. Om de uitgebreide set aan services overzichtelijk aan developers te presenteren, heeft ISAAC een volledig herontwerp gedaan van de User- en Developer Experience: Ingenico Connect.

Eén van de uitdagingen was om alle functionaliteiten op een heldere manier beschikbaar en inzichtelijk te maken voor merchants. Het landschap van Ingenico ePayments is bovendien veel uitgebreider dan dat van nieuwe Payment Service Providers (PSP's) die nog maar net de markt bestormen met een beperkt portfolio van betaalproducten. Er werd een volledig nieuwe, gestructureerde REST API ontworpen en geïmplementeerd en structureel gedocumenteerd.



In de API-structuur is extra zorg besteed aan het gebruik van de juiste naamgeving die op een heldere manier verwijst naar concepten en hiërarchie in de structuur. De API-structuur wordt aangevuld met een aantal andere, ondersteunende componenten, zoals de volledige API-referentie in de developer portal, de server SDKs (zoals Node.js, Java en PHP) en de client SDKs (Android, iOS en JavaScript). Met behulp van de formele hiërarchie en herbruikbare scripting zijn de REST API, de API-referentie en de SDKs altijd automatisch gesynchroniseerd en up-to-date voor een optimale ervaring wanneer developers integreren, testen en runnen.

Na het ontwerp van de REST APIs heeft ISAAC ook het 'exposen' van de interne betaalstatus van het platform onder handen genomen met de inzet van Webhooks. Deze push-notificatiestandaard maakt het voor merchants mogelijk om precies die 'events' op te vangen en te verwerken die ze zelf willen hebben. Denk aan het money-in-bank moment of een eventuele fraudeverdenking. Of juist het reageren op een terugboeking, terwijl je de order al aan het uitleveren bent. Webhooks maken het platform van Ingenico extra open en dragen enorm bij aan de Developer Experience.



### Businesswaarde

Voor het herontwerp van de User- en Developer Experience voor Ingenico was het aansluiten op het internationale betaalnetaamwerk voor merchants geen makkelijke opgave. Zij waren vaak weken bezig om de complexe interfaces te ontsluiten vanuit hun systemen, konden lastig testen wat een lage explorability tot gevolg had en de interne status van een betaling was alleen met 'polling' en wat onduidelijke statuscodes op te vragen. Na de introductie van Ingenico Connect, dat veel meer met een 'platform mindset' is ontworpen, sluiten developers door de sterk verbeterde Developer Experience in enkele uren (en bij meer complexe betaalproducten in enkele dagen, in verband met regelgeving) aan op het internationale betaalnetaamwerk van Ingenico.

Ook worden er nu heel nieuwe integraties bedacht door developers van merchants, die voorheen niet mogelijk zouden zijn geweest. Denk bijvoorbeeld aan de naadloze aankoopbeleving op internationale game-sites als die van Blizzard of aan de check-out van online tickets in de mobiele app van NS. De Developer Experience maakt het aantrekkelijk voor meer merchants om Ingenico in te zetten. De sterke focus op User Experience in de betaal flow zorgt voor een optimale conversie in al die nieuwe customer journeys. De schaalbaarheid van de architectuur bewijst zich door het aantal nieuwe merchants dat gebruikmaakt van Ingenico, maar ook de volumegroei per merchant. Black Friday bij Ali Baba? Via het Ingenico Connect platform kan de commercereus in China probleemloos enorme hoeveelheden betalingen afhandelen.

Bekijk de hele case →

## De gebruiker centraal op het platform

Vanuit het oogpunt van UX wordt platformopenheid goed benut wanneer in het design, op alle lagen van het platform, de gebruiker centraal staat. Op een goedwerkend platform kunnen gebruikers acties en transacties direct en eenvoudig afhandelen, maar worden ook data en interacties op een heldere, herkenbare manier overgebracht of gevisualiseerd. Om dat goed voor elkaar te krijgen, is het nuttig om met methodieken vanuit onder andere Service Design en Design Thinking alle gebruikerservaringen en -interacties te ontwerpen en ontwikkelen.

Het startpunt van UX- en UI Design op een platform zijn de gebruikers.



Vanuit een user centric benadering doe je onderzoek naar gebruikers, waardoor je exact leert begrijpen wat hun behoeften en wensen zijn. Daarbij horen bijvoorbeeld gebruikersonderzoek, observaties, interviews en workshops, maar ook onderzoek naar contextafhankelijkheden. In dat laatste type onderzoek wordt ook naar ogenschijnlijk eenvoudige dingen gekeken, zoals preferred device, wat gebruikers zoeken, wat hun taken zijn of in welke omgeving of routine zij het platform gaan gebruiken. Door een variëteit aan aspecten van gebruikers te onderzoeken, probeer je de echte vraag en behoeften van gebruikers zo exact mogelijk te achterhalen. Gebruikers bepalen namelijk uiteindelijk of het platform het überhaupt waard is om actief te gebruiken.

## Positieve netwerkeffecten

Het positieve netwerkeffect gaat in de kern over aantrekkelijkheid en groei, gedreven door toegevoegde waarde die via het platform wordt geboden. Aantrekkelijkheid van een platform begint bij 'het probleem' dat voor gebruikers wordt opgelost of de 'nieuwe service' die beschikbaar wordt gemaakt.

Zo bracht Microsoft Windows jaren geleden softwareontwikkelaars, gebruikers en hardware leveranciers bij elkaar en voegde voor hen allen veel waarde toe. Microsoft ontwikkelde immers software die op verschillende apparaten van verschillende fabrikanten prima presteerde. Een ook de appstore van Apple is een platform met netwerkeffecten. Ondanks dat er met enige regelmaat misschien kritiek is over de manier waarop, controleert Apple elke app die aan de Appstore wordt toegevoegd. Daarmee garandeert het bedrijf in sterke mate het aanbod van veilige en kwalitatieve apps.

In eerste instantie gaan positieve netwerkeffecten over toegevoegde waarde, maar in de tweede instantie gaat het over succes en daaruit volgende groei. Elke keer dat een gebruiker zich aansluit op een platform en er gebruik van gaat maken, groeit het platform en wordt het gevarieerder. Dat maakt het aantrekkelijker voor andere gebruikers om ook aan te sluiten en het creëert een cyclus van groei: een zichzelf versterkend effect naar meer gebruikers en dus ook weer meer waarde.

Vaak is een positief netwerkeffect effect tweezijdig. In ieder geval als er op een platform twee partijen nodig zijn om vraag en aanbod te matchen, bijvoorbeeld bij SnappCar of Airbnb. Deze platformen worden aantrekkelijk voor gebruikers als er veel aanbieders zijn, maar andersom geldt natuurlijk hetzelfde. Het grotere, gevarieerdere aanbod veroorzaakt meer waarde voor gebruikers, als er veel

aanbieders zijn, maar andersom geldt natuurlijk hetzelfde. Het grotere, gevarieerdere aanbod veroorzaakt meer waarde voor gebruikers, wat aantrekkelijkheid oplevert voor nieuwe gebruikers. Dat wordt ook wel een tweezijdig positief netwerkeffect genoemd. Snel de juiste hoeveelheid massa opbouwen op het platform is dus wenselijk, of het platform nou eenzijdig of open end-to-end is.

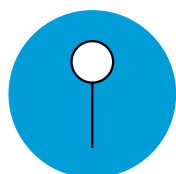
### Het zijn gebruikers en developers die een platform groot maken

De juiste hoeveelheid massa opbouwen op het platform, hoe ga je daar nou concreet mee aan de slag? Als we kijken naar developers als doelgroep, dan is een aantrekkelijk platform in sterke mate afhankelijk van (1) vertrouwen in het platform en (2) zichtbaarheid. In de Developer Experience op het platform geef je dat vorm met de volgende zaken:



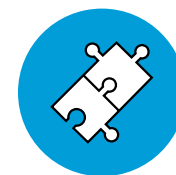
#### Wek vertrouwen door kwaliteit te leveren

Kwaliteit van code en API-ontwerp op het platform straalt rechtstreeks af op je corporate brand. De code die hoort bij het platform, zoals die van SDKs of plug-ins, wordt gebruikt op de servers van je klanten. Zorg dus voor hoge kwaliteit en clean-code, vrij van code smells. Naadloze samenwerking met andere code (voorbeeld door goede scoping) is een hygiënefactor.



#### Be where the developers are

Gaat over zichtbaarheid op plekken waar de development community aanwezig is, bijvoorbeeld op Github, Slack en Stackoverflow. Denk erover na waar developers graag vragen stellen en zorg ervoor dat ze via die kanalen antwoorden kunnen krijgen. Juist door op voordehand liggende plekken zichtbaar te zijn, is het eenvoudiger om snel(ler) te groeien in gebruikersaantallen.



#### Maak integratie makkelijk

Faciliteer dat developers laagdrempelig gebruik kunnen maken van code op het platform. Vuistregel is uiteraard dat integraties in ieder geval niet complexer mogen zijn dan bij de concurrent.

Het is dus verstandig om aan te sluiten bij standaarden en bijvoorbeeld componenten beschikbaar te stellen, waaronder APIs, Software Development Kits (SDKs) en plug-ins.



#### Gebruik geen exotische licenties

Kwaliteit van code en API-ontwerp op het platform straalt rechtstreeks af op je corporate brand. De code die hoort bij het platform, zoals die van SDKs of plug-ins, wordt gebruikt op de servers van je klanten. Zorg dus voor hoge kwaliteit en clean-code, vrij van code smells. Naadloze samenwerking met andere code (voorbeeld door goede scoping) is een hygiënefactor.



#### Communiceer over de roadmap

Helder communiceren over de roadmap van het platform helpt developers te begrijpen waar het systeem vandaan komt en naartoe groeit. Bovendien bevordert dat de creativiteit van developers om het platform in te zetten voor (nieuwe) toepassingen, maar bijvoorbeeld ook in te spelen op aankomende releases. Iets dat netwerkeffecten weer positief kan beïnvloeden.



#### Luister naar de community

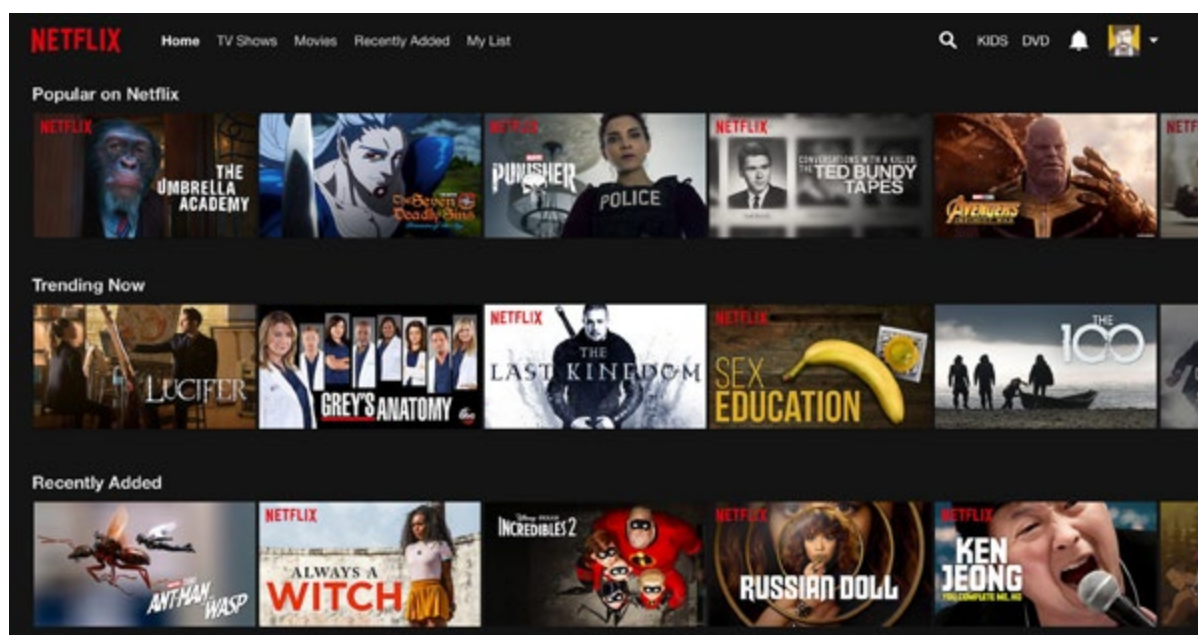
Door te luisteren naar de bevindingen van gebruikers, kun je gericht aan de slag met 'quick wins'. Positieve netwerkeffecten gaan immers over de aantrekkelijkheid van het platform. Als feedback van de community bijdraagt aan de kwaliteit, dan is het natuurlijk zinvol om daar iets mee te doen. Ook developer centricity is – zoals eerder benadrukt – een voorwaarde voor platformsucces.

## User centricity met 'Give it to me now!'

Als we kijken naar gebruikerservaring van niet-developers is een concept als 'Give it to me now!' cruciaal. Gebruikers hebben altijd gerichte doelen of taken, die ze tegenwoordig in toenemende mate vervullen via digitale middelen. Dat geldt zowel voor consumenten als op een platform voor B2B. Het ontwerp voor de platformervaring en customer journey start daarom vanuit het principe 'Give it to me now!'.

Vanuit het 'Give it to me now'-concept zijn flows 100 procent gericht op handelingen en acties die horen bij het doel of de taak van de gebruiker.

Netflix is hiervan een goed voorbeeld. Na het al dan niet inloggen, krijg je direct waarvoor je komt: film- en serie-aanbod, afgestemd op jouw kijkgedrag en anticiperend op jouw wensen en behoeften.



Geen poespas, geen gedoe, gewoon direct zien wat je als gebruiker verwacht. Stel we trekken dit voorbeeld door naar een pensioenoverzicht. Veel mensen checken zeker niet dagelijks hun pensioen, maar willen wel goed begrijpen hoe zij ervoor

staan op de dag dat zij met pensioen gaan. Vanuit het 'Give it to me now!' gedachtegoed wil je direct na inloggen daarom dus helder visualiseren wat er nu in de pensioenpot zit, maar ook wat er naar verwachting op de eerste dag van jouw pensioen in de pot zit, enzovoort. Daarbij horen doelgerichte stappen, geen onnodige vragen op welk moment dan ook en heldere flows die in elke stap bijdragen aan de doelen en taken van gebruikers. Uiteindelijk valt of staat succes van een platform, net zoals bij Developer Experience, met de mate waarin gebruikers een platform überhaupt willen gebruiken.

User Experience en Developer Experience spelen de hoofdrol.

## User Experience voor multi-level rollen op een platform

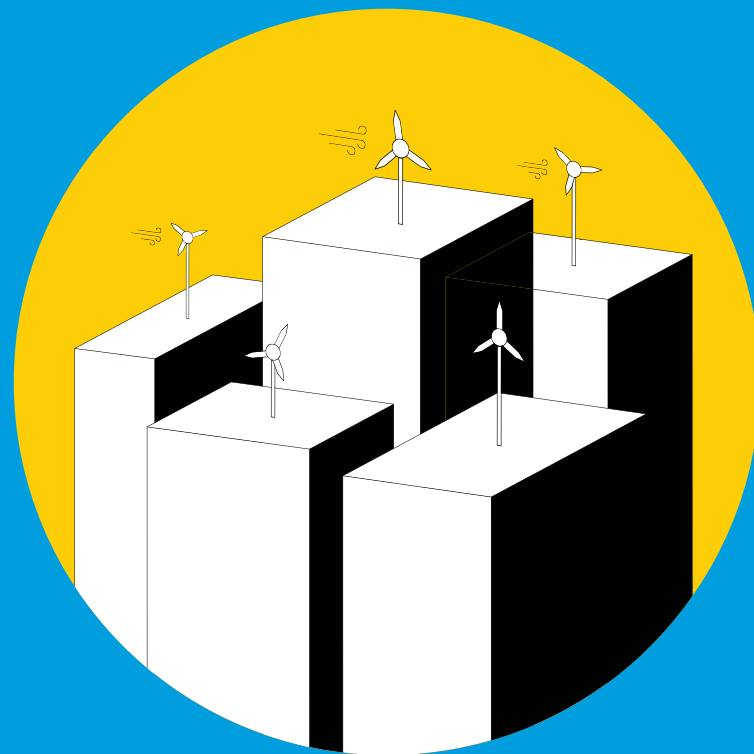
Sommige bedrijven hebben naast de User Experience en Developer Experience voor 'normale' gebruikers, nog een extra uitdaging. Iets dat geregeld voorkomt bij een platform dat gericht is op een industry vertical. Laten we bijvoorbeeld kijken naar Salesforce, een cloud based CRM-platform. Salesforce zit bij veel bedrijven verweven in dagelijkse taken van medewerkers die het CRM op wat voor manier dan ook als resource- of rapportagetool inzetten. Juist op een platform met de complexiteit en uitgebreidheid van Salesforce, zijn beheerbaarheid en efficiënt werken belangrijke thema's. Gebruikers werken er immers vaak een groot deel van de dag mee. Door specifiek rekening te houden met de verschillende type gebruikers en daarvoor ook verschillende ervaringen te creëren, zorg je ervoor dat iedereen goed met het platform kan werken. Voorbeelden daarvan zijn 'uitgeklede' user flows, 'short cuts', specifiek ontwerp voor andere devices of schermopstellingen of het gebruik van herkenbare, domeinspecifieke termen. In zo'n geval worden er in het UX- en UI Design specifiek rekening gehouden met zogenaamde power users.

Power users zijn gebruikers die andere doelen en taken hebben op een platform, en vooral veel meer van hun tijd besteden dan 'normale' gebruikers.



Case

# Energy trading platform



De energie-industrie is de afgelopen jaren enorm aan het veranderen. In de markt komt, naast de grote energiegi-ganten, steeds meer ruimte voor kleinere initiatieven die de markt effectief willen betreden. Onder meer door de ontwikkelingen op het gebied van renewable energy. Het krachtige platform van Origami stelt de energie -industrie in staat om alles wat stroom gebruikt, opslaat en genereert geheel geautomatiseerd en realtime te beheren. Dergelijke 'energy assets' zijn makkelijk aan te sluiten, te regelen en te optimaliseren. De volgende stap voor het Origami platform is traders bij energiebedrijven helpen om rendement uit energy trading te maximaliseren door effectieve tradingadviezen te geven.



Het platform van Origami brengt vraag en aanbod van energie zo effectief mogelijk bij elkaar. Om dat proces nu en in de toekomst te verbeteren, werkt een team van promovendi aan Cambridge University continu aan de optimalisatie van datamodellen. Factoren als forecasting, klimaatdata, marktdata, locatie, type energiebronnen en concurrentiedata spelen daar bijvoorbeeld een rol in. Deze voorspellingen resulteren in complexe tradingadviezen. Omdat energy traders vaak in een tijdsbestek van enkele minuten een beslissing moeten nemen over hun strategische positie in het volgende tijdvak van steeds een half uur, is het belangrijk dat zij adviezen snel en eenvoudig kunnen begrijpen. Origami heeft ISAAC gevraagd om uit te werken hoe tradingadviezen het beste verwerkt kunnen worden in een UX- en UI Design waarmee traders daadwerkelijk effectief kunnen werken. In een reeks van workshops op het kantoor van ISAAC en bij Origami in Engeland is gewerkt aan het doorgronden van beoogde gebruikers van de nieuwe Origami adviestool, begrip van het domein en de complexiteit van de tradingadviezen, maar ook aan journey mapping, persona's en system mapping. Met een mix van methoden uit Design Sprint en Service Design is toegewerkt naar concepten en prototypes van het UX- en UI Design.



# Schaalbaarheid

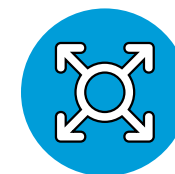
Als het gaat over schaalbaarheid wordt vaak bedoeld op de flexibiliteit die een platform technisch gezien moet bezitten, bijvoorbeeld om grote aantallen transacties tegelijk te verwerken of realtime beschikbaar te zijn voor grote aantallen gebruikers tegelijk. Vanuit het oogpunt van DX en UX is het op het gebied van schaalbaarheid belangrijk om ook oog te hebben voor andere zaken.

## Developers de gelegenheid bieden om schaalbaarheid te realiseren

Om een platform te laten schalen, moet er veel development plaatsvinden. Wanneer daar – zoals in de meeste gevallen – de resources van externe developers bij nodig zijn, dan is er eigenlijk maar één ding dat telt: toegankelijkheid en gemak in een fantastische DX.

In een tijd waarin developer resources één van de meest organisatiebeperkende factoren is, zijn happy en tevreden developers erg belangrijk.

Zij zullen dan namelijk gemotiveerd zijn om goede digitale ervaringen voor jouw platform businessmodel te creëren. In het licht van schaalbaarheid willen we hier stilstaan bij twee do's:



## Schaalbaarheid van het platform

Natuurlijk moet een platform schalen onder load, maar we hebben het hier niet over systeembelasting. Schaalbaarheid in deze context gaat over schalen naar meer developers, meer toepassingsgebieden en meer integraties. Dat kun je faciliteren via logische 'entry points', hooks en events. Via die 'deuren' kunnen externe developers op een

Het uitgangspunt van het ontwerp is de power user-rol die gebruikers van Origami hebben. Gedurende hun hele werkdag werken zij in verschillende systemen, die soms op wel 12 schermen draaien. Via die systemen verhandelen ze stroom, checken ze weersvoorspellingen, sluiten ze assets aan, regelen en optimaliseren ze die. Het belangrijkste in deze fase was om de fysieke assets en de financiële rendementskant bij elkaar te brengen in de visualisaties in tradingadviezen.



Energy traders moeten en willen in één klap in een tradingadvies kunnen zien hoe het rendement van hun energyassets zich gedurende verschillende tijdvakken zal ontwikkelen en welke fysieke acties daarbij passen: aan- of uitzetten, intensiveren of afschalen



gecontroleerde manier met het platform interacteren. Bied daarnaast ook inzicht in de interne processen van je service zolang dat interessant zou kunnen zijn. Laat bijvoorbeeld performance- en data-inzichten zien en bied die ook weer aan via APIs. Ook is het nuttig om krachtige clients optimale integraties te bieden via webhooks en query-talen als GraphQL. Door garanties te bieden over API performance neem je schaalbaarheid echt serieus. Het helpt daarbij om een doel te stellen: 'mijn platform moet binnen honderd milliseconden antwoorden onder deze voorwaarden'. Developers zien dat weer als positief signaal om jouw platform in hun unieke workflows op te nemen. Zo schaal je zonder maatwerk door gebruik te maken van de developmentkracht buiten je eigen organisatie.



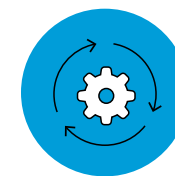
### Transparant communiceren over systeemstatus

24/7 beschikbaarheid van systemen is belangrijk om een platform succesvol te kunnen runnen. Het kan echter altijd voorkomen dat (een deel van de) functionaliteiten op een

bepaald moment niet beschikbaar zijn, om wat voor reden dan ook. Juist op die momenten is het van cruciaal belang om systeemstatus inzichtelijk te maken, maar ook een voorspelling te tonen vanaf welk moment het systeem weer volledig beschikbaar is. In de praktijk is het lastig om dat echt goed te doen, maar besef ook dat storingen en 'errors' onderdeel zijn van het 'contract' dat je platform biedt. Behandel ze met aandacht en ontwerp ook voor de uitzondering. En waarom zou je geen API of Webhook aanbieden die bij storingen pro-actief een berichtje naar buiten stuurt?

### Performance optimalisatie via User Experience design

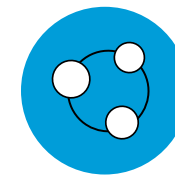
Platformschaalbaarheid vanuit UX perspectief gaat voor een belangrijk deel natuurlijk ook over happy en tevreden gebruikers. Die heb je namelijk nodig om de gebruikersaantallen op het platform effectief te laten groeien. Maar uiteindelijk is het ook belangrijk dat een platform snel en effectief kan worden doorontwikkeld. Drie aandachtspunten om dat goed voor elkaar te krijgen:



### 1. Datagedreven doorontwikkeling

Maak een gedegen meetplan, gekoppeld aan de strategie en doelen. Investeer in het echt doorgronden van je doel funnels. Door op het platform gebruikerservaringen met bijvoorbeeld

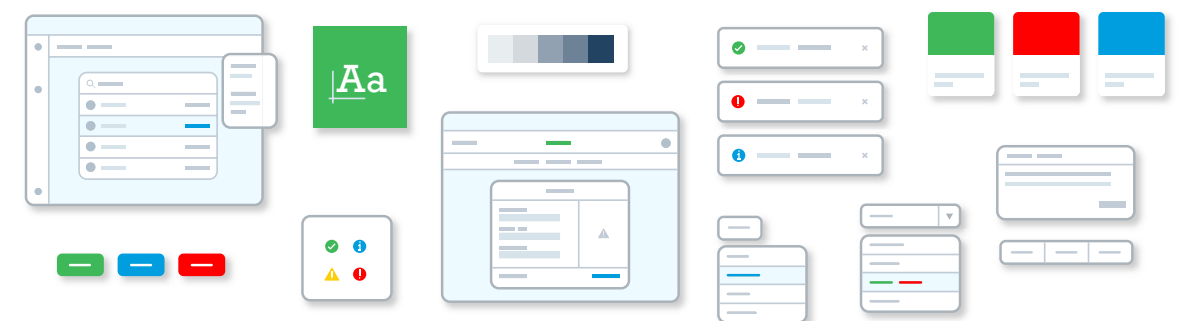
tagging en visual heat maps te loggen, is de performance van het platform niet alleen inzichtelijk voor (development) teams, maar ontstaan ook waardevolle inzichten in ervaringen van gebruikers. Die inzichten dienen weer als input voor doorontwikkeling, die resulteren in performanceverbeteringen van het platform. Wees daarnaast niet bang om te testen: alle grote platformspelers rollen continu kleine wijzigingen uit, meten het resultaat, leren en rollen weer opnieuw.



### 2. Design system

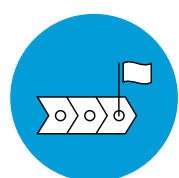
Misschien is de start van het te bouwen platform bescheiden, maar bedenk dat UX- en UI Designs die in het begin worden gemaakt van invloed zijn op toekomstige producten en services.

Herhaling en herkenbaarheid spelen namelijk een centrale rol in elke UI. Een goede basis om uiteindelijk schaalbaar en flexibel te kunnen ontwikkelen aan de vormgeving en gebruikersbeleving van een platform, is de ontwikkeling van een design system.



Hierin worden processen, style guides en componenten vastgelegd, en maak je die vervolgens als standaard beschikbaar in moderne desing tooling, zoals Sketch of Adobe XD. Het opzetten van deze ontwerpbibliotheek kost in het begin weliswaar tijd en energie, maar de investering verdient je moeiteloos terug doordat UX-teams zich kunnen focussen op de User Experience van nieuwe producten en services.

Met een design system beschikken ze immers steeds over componenten die hun kracht al hebben bewezen. Dit bespaart tijd en geld, en komt de gebruikerservaring van services ten goede. Een groot platform heeft vaak een apart team dat continu werkt aan het design system. Voor de wat meer bescheiden platformen is de doorontwikkeling van het centrale design system met UX- en UI-inzichten- en componenten een mooie afwisseling met de taakspecifieke uitbreidingen die de rest van de tijd in beslag nemen.



### 3. Een UX release train

Om continu te optimaliseren in de UX is het betrouwbaar kunnen releasen van interface-changes belangrijk. Een release zit aan het einde van je 'release train'. Geen platform kan

zonder een geoliede machine van releases. Zorg voor goede tests; het design system en de componenten daarin zijn afgedekt met unit tests en browser tests. Hierop wil je ook blijven innoveren om steeds sneller en beter te kunnen testen. De APIs waar de UX zijn data uit haalt kennen eigen integratietests en het software ontwikkelingsproces is herhaalbaar en centraal. 'If it hurts, do it more often' is een oude maar bewezen uitspraak rond releasen.

Case

## Nieuw UX design voor complex financieel product

Computershare



Zekerheid, vindingrijkheid en voordeel. Daar is de aanpak van financial service provider Computershare op gebaseerd. De wereldwijde marktleider biedt voor grote bedrijven diverse financiële producten, waaronder spaar- en aandelenprogramma's, om de betrokkenheid van medewerkers bij een organisatie te vergroten. Met deze 'employee share plans' hebben wereldwijd miljoenen medewerkers aandelen en creëren zij daarmee rendement. Om de waarde van dit redelijk complexe financiële product op een begrijpelijke manier inzichtelijk te maken, ontwikkelde ISAAC diverse user journeys en UX designs voor de share plans. Centraal stond hierbij de opbouw van een design system.

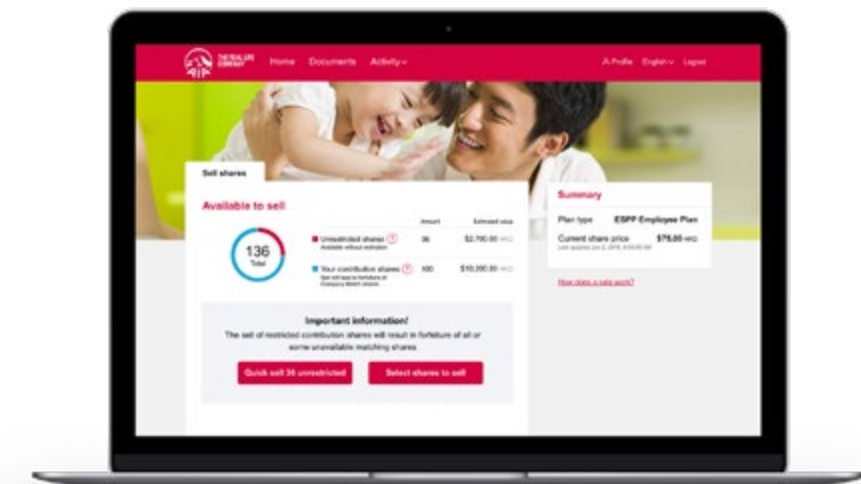


Een doordacht UX design waarmee gebruikers altijd de status van hun share plan begrijpen, heeft flink wat voordelen. Gamification van UX zet aan tot herhaalgebruik van de share plan toolbox. Het gebruik ervan is daardoor immers eenvoudig en leuk, omdat de UX begrip en zekerheid kweekt. Gebruikers met de nieuwe versie van de share plan tool op hun smartphone, bellen minder vaak naar het service center, stellen minder vragen en scoren beter in het behalen van de doelstellingen.

Zekerheid, vindingrijkheid en voordeel. Daar is de aanpak van financial service provider Computershare op gebaseerd. De wereldwijde marktleider biedt voor grote bedrijven diverse financiële producten, waaronder spaar- en aandelenprogramma's, om de betrokkenheid van medewerkers bij een organisatie te vergroten. Met deze 'employee share plans' hebben wereldwijd miljoenen medewerkers aandelen en creëren zij daarmee rendement. Om de waarde van dit redelijk complexe financiële product op een begrijpelijke manier inzichtelijk te maken, ontwikkelde ISAAC diverse user journeys en UX designs voor de share plans. Centraal stond hierbij de opbouw van een design system.

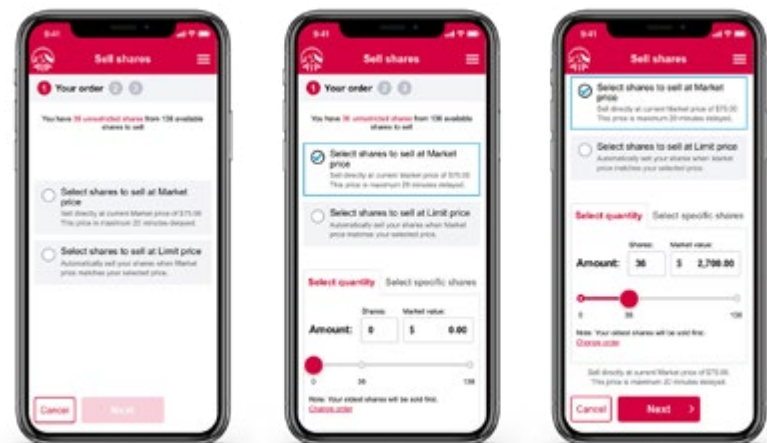


Een doordacht UX design waarmee gebruikers altijd de status van hun share plan begrijpen, heeft flink wat voordelen. Gamification van UX zet aan tot herhaalgebruik van de share plan toolbox. Het gebruik ervan is daardoor immers eenvoudig en leuk, omdat de UX begrip en zekerheid kweekt. Gebruikers met de nieuwe versie van de share plan tool op hun smartphone, bellen minder vaak naar het service center, stellen minder vragen en scoren beter in het behalen van de doelstellingen.



Het designproces werd met Design Thinking vormgegeven. Pressure cooker sessies brachten ISAAC- en Computershare-teamleden via 'empathy' – wie zijn de gebruikers en wat willen zij echt – en de daaruit volgende 'ideation' – nieuwe ideeën op basis van industry best practices – al snel bij schetsen, mock-ups

en testbare prototypes. De share plan-acties die door complexe wizards worden ondersteund, zoals acties op een optieplan en verkopen van aandelenpakketten, werden in kaart gebracht met Event Storming. De primaire processen kregen prioriteit; uitzonderingen werden ook in de uiteindelijke user interface secundair. Altijd op basis van het principe ‘Give it to me now!’.



Voor het ontwerp van een optimale UX, is productkennis van groot belang. Een dergelijk ontwerp kan ook alleen tot stand komen in een gefocuste mix van domeinexperts en ervaren UX'ers met marktkennis. Het projectteam van ISAAC volgde bij Computershare daarom een ‘crash course’ over share plans, optieregelingen en bonuswetgeving. Al snel bleek de complexiteit van het product vooral te zitten in de scala aan mogelijkheden en configuraties, de afhankelijkheden van landspecifieke wet- en regelgeving en de behoeften van de ‘big corporate’ klanten van Computershare. Belastingregels die bijna jaarlijks veranderen, maken de mix van complexiteit compleet. Op basis van jarenlange ervaring in de financiële markt slaagde ISAAC er echter toch in om het share plan voor deelnemers eenvoudig inzichtelijk te maken.

Stap voor stap rolt Computershare de nieuwe interfaces, wizards en customer journeys op basis van het nieuwe design system uit over het brede, wereldwijde portfolio. Daardoor geven ook steeds weer nieuwe gebruikersgroepen feedback over het financiële product. Het begrip van het product verbetert en de betrokkenheid van gebruikers neemt zichtbaar toe. Kortom, van lastig te begrijpen koersgrafieken en optieplannen naar engagement en ‘employee happiness’.

Bekijk de hele case →

## De conclusie

De platformrevolutie – de krachtigste ontwikkeling van het digitale tijdperk – biedt bedrijven de kans om businessmodellen te creëren die meer waarde bieden dan traditionele rechtlijnige ‘push-to-market’ modellen. En bovendien huidige markten compleet op zijn kop te zetten. Ook nieuwe realiteiten dringen zich overduidelijk aan ons op: de enorme beweeglijkheid van de digitale klant, maar ook de noodzaak voor andere salesstrategieën en de urgentie van het probleem dat met het platform wordt opgelost. Duidelijk onderscheidend vermogen met een fantastische User Experience en Developer Experience is op een platform onmisbaar.

Om een platform succesvol te maken, moet er allereerst veel development aan het platform plaatsvinden. Daarentegen zijn developer resources echter wel één van de meest organisatiebeperkende dingen van dit moment. Dat benadrukt daarom alleen maar meer het belang om als bedrijf met een platform te vechten voor die Developer Experience. En daarnaast ook heel slim om te gaan met de doorontwikkeling en optimalisaties – volledig gedreven vanuit de data die het platform daarover genereert. Hetzelfde geldt voor de gebruikerservaring. Of gebruikers überhaupt het platform het gebruiken waard vinden, hangt in sterke mate samen met de ervaring die hen wordt geboden. Niet alleen het probleem dat het platform voor ze oplost, maar ook de seamless, ‘Give it to me now!’-beleving die ze daar krijgen, zal hun oordelen positief of negatief beïnvloeden.

# Checklist

In dit whitepaper hebben we 21 praktische handvatten toegelicht voor de ontwikkeling van de User- en Developer Experience op een platform. In de checklist hieronder vatten we die beknopt samen.

## Platformopenheid

- DX: Wees explorable en laagdrempelig toegankelijk voor developers
- DX: Gebruik open standaarden, zoals npm, Composer, Maven, Github, Open API en GraphQL
- DX: Zorg voor een goede 'explore and play' sandbox
- DX: Stel goede documentatie beschikbaar voor verschillende doelgroepen op logische plaatsen
- DX: Houd rekening met nieuwe use cases en laat daarom het creatieve denken over aan de community
- UX: Stel de gebruiker centraal
- UX: Zet methodieken uit Service Design en Design Thinking in en probeer zo exact mogelijk de drivers van gebruikers te achterhalen

## Positieve netwerkeffecten

- DX: Wek vertrouwen door kwaliteit te leveren, vrij van code smells en code die naadloos samenwerkt met andere code
- DX: Gebruik geen exotische licenties, houd tooling zo toegankelijk mogelijk
- DX: 'Be where the developers are' en zorg dat je beschikbaar bent voor vragen op logische plekken als Stackoverflow, Slack en Github
- DX: Maak integratie zo makkelijk mogelijk
- DX: Communiceer over de roadmap, zodat developers begrijpen en kunnen anticiperen op waar het platform vandaan komt en naartoe groeit

- DX: Luister naar de community en neem hun input serieus voor doorontwikkeling
- UX: Pas 'Give it to me now!' toe waarmee je gebruikers doelgericht faciliteert bij hun doelen en taken op jouw platform
- UX: Houd rekening met eventuele multi-level platformrollen en power users en pas de designs daarop aan

## Schaalbaarheid

- DX: Zorg dat een platform kan schalen onder load, maar juist ook naar meer developers, toepassingsgebieden en integraties
- DX: Communiceer transparant over systeemstatus
- UX: Ontwikkel datagedreven aan het platform en maak daarom een gedegen meetplan dat is gekoppeld aan de strategie
- UX: Wees niet bang om te testen: alle grote platformspelers rollen continu kleine wijzigingen uit, meten, leren en rollen weer iets nieuws uit
- UX: Start vanaf het begin al met het opbouwen een Design System, dat altijd centraal beschikbaar is in design tooling als Sketch of Adobe XD
- UX: Zet een UX release train op, want voor continu optimaliseren is het betrouwbaar kunnen releasen van interface-changes belangrijk



# Over de experts



## Friso Geerlings – Chief Technology Officer

Met een brede technische achtergrond richt Friso het vizier van ISAAC op effectieve, veelal open source, internettechnologie om de meest complexe vraagstukken voor klanten te voorzien van een effectieve digitale strategie en oplossing. De hoofdfocus van Friso ligt daarbij op customer experience, webtechnologie, APIs, data-driven decisions, transacties, high-performance systemen en integratie. De brug tussen developers, gebruikers en systemen bouwen staat voor hem al jaren centraal.



## Michael van Leeuwen - Business Unit Manager Finance

Na 10 jaar in de betaalindustrie bij equens Worldline werkzaam te zijn geweest, maakte Michael in het voorjaar van 2019 de overstap naar ISAAC. Recentelijk deed hij in het kader van de MBA Business & IT aan Nyenrode Business University onderzoek naar strategische opties voor Open Banking Platforms. Hij richt zich op platform- en ecosysteemontwikkeling, trends in de financiële industrie en op de mogelijkheden van veranderende regelgeving zoals PSD2 en hoe bedrijven daar slim op in kunnen spelen.



## Koen Peters – Technical Director Front-end

In zijn rol van Technical Director Front-end zorgt Koen ervoor dat zijn team altijd efficiënt en up-to-date is met de nieuwste technieken en frameworks op het gebied van front-end. Hij heeft goede kennis van web development en business analyse, is pragmatisch ingesteld en heeft een echte 'getting things done' mentaliteit. Zijn focus binnen ISAAC is gericht op development efficiëntie, bedrijfsbreed, door middel van standaardisering en het gebruik en de ontwikkeling van generieke componenten, de bevordering van kennisdeling en het waarborgen van codekwaliteit.



## Joost van Traa – Art Director

Joost staat aan het roer van de afdeling Concept & Design bij ISAAC. Voor iedere markt ontwikkelt Joost een resultaatgericht, modern concept. Onder het motto 'goed begonnen is al half gewonnen' voorziet hij klanten al vanaf de tekentafel van advies. Joost is gespecialiseerd in UX op platforms en dan vooral voor de financiële industrie. Hij weet exact complexe diensten en producten doeltreffend om te zetten naar een geschikte UX.



## Johan Michels – Lead Designer

Onder het motto 'embrace the whitespace' ontwerpt Johan voor bedrijven in Commerce en Finance gebruiksvriendelijke UX en UI. Met behulp van mock-ups, prototypes en micro-animaties zorgt hij voor een eindresultaat dat gezien mag worden.

# Over ISAAC

ISAAC helpt bedrijven met strategische scoping, software development, implementatie en datagedreven groei van digitale services. Met bewezen techniek, tools en frameworks ontwikkelen we een breed scala aan oplossingen, inclusief online payments, consumptief krediet, cards en loyaliteitsprogramma's. Met **20 jaar ervaring** in de financiële industrie en commerciële dienstverlening en technische know-how van **140+ developers** in Eindhoven, werken we voor toonaangevende bedrijven in Europa. Voor onze projectteams is het business as usual om een aanvulling te zijn op in-house IT-teams en nieuwe oplossingen naadloos te integreren met bestaande systemen en processen. Door te werken met ISAAC verminder je projectrisico, realiseer je snellere time-to-market, verbeteren we klanttevredenheid en word je volledig ontzorgd van operationele lasten.

## Een greep uit ons portfolio

PHILIPS

Atradius  
Collections

DMG

payvision 

TOMMY  
HILFGER

  
EindhovenAirport

van Dale

Qander.  
CONSUMER FINANCE

CM.

DAF

 BNP PARIBAS

WINKELSTRAAT.NL  
FASHION. BOUTIQUE STYLE



+31 40 2908979  
welkom@isaac.nl  
[www.isaac.nl](http://www.isaac.nl)

Marconilaan 16  
5621AA Eindhoven