

# Inleiding

Je hebt dit boek in handen, omdat je op zijn minst vermoedt dat je betere prototypes zou moeten maken van je apps. Je creëerde een app die het probleem van de eindgebruikers niet oploste. Of je bouwde een app die nooit gedownload werd, omdat er vergelijkbare apps waren met een veel beter interactiemodel. Of je app kreeg slechte feedback in de store, omdat je een onhandige gebruikersinterface ontwierp. Al deze problemen voorkomen je door eerst prototypes te maken, die te testen op mensen uit je doelgroep en de resultaten daarvan mee te nemen in je ontwerp.

Prototypes maken is een essentiële fase in het ontwikkelen van apps voor elk platform. Er zijn veel online- en offlinetools beschikbaar waarmee je een prototype voor een app kunt maken. De basisprincipes van die tools zijn gelijk, maar er zijn verschillen in de uitwerking. De opzet van dit boek is om de overeenkomsten en de verschillen tussen die tools zichtbaar te maken, zodat je zelf kunt beslissen met welke je aan de slag gaat.

We beginnen door nog eens te benadrukken wat een prototype is en waarom je dat nodig hebt bij het ontwerpen van een mobiele applicatie. Het is waardevol om even stil te staan bij andere manieren om voor software prototypes te maken, zoals schetsen, paperprototypes, HTML-prototypes en offlinesoftware voor prototypes. De voorwaarden voor een goed prototype zetten we eerst op een rijtje. We introduceren de vijf onlinetools voordat we aan de behandeling van de overlappende eigenschappen ervan beginnen.

Een belangrijk gedeelte van dit boek is gewijd aan de generieke eigenschappen van de belangrijkste onlinetools voor prototypes. Pas daarna kunnen we gewicht geven aan specifieke eigenschappen, zodat je een bewuste en verantwoorde keuze kunt maken.

Ten slotte gaan we in op de toepassing van je prototype: hoe test je eigenlijk en hoe pas je de resultaten daarvan toe op je ontwerp?

Aandacht besteden aan een ontwerp waarvan je proefondervindelijk weet dat het werkt voor jouw doelgroep, kost meer tijd aan het begin van het proces. Door van tevoren de juiste onderbouwde keuzes te maken, vermijd je frustraties en fouten bij het bouwen van je app. Daarmee zet je flinke stappen in de richting van goede, succesvolle apps, die gebruikers downloaden, installeren, gebruiken én betalen.

# Inhoud

1	Over prototypes	1
	Waarom een prototype?	2
	Hoe bereik je succes?	3
	Verfijning van een prototype	4
2	Ander manieren voor prototyping	7
	Concept	8
	Schetsen	8
	Paperprototypes	12
	HTML-prototypes	14
3	Offlinetools voor prototypes	17
	Microsoft PowerPoint	18
	Adobe Photoshop	20
	Axure Rapid Prototyping	22
	Balsamiq Mockups	25
	Justinmind Prototyper	26
	Adobe Experience Design	30
	Flinto voor Mac	35
	Marvel voor iOS en InVision iOS app	35
4	Hoe zet je een goed prototype op?	37
	Wat is een goed prototype?	38
	Hoe ga je te werk?	40
	Wat heb je nodig voor onlinetools?	42
5	Introductie van de tools	43
	Flinto Lite	44
	Fluid.UI	45
	InVision	47
	Marvel	48
	Proto.io	50

6	Generieke eigenschappen van onlinetools	55
	Ontwerpen	56
	Het prototype installeren en delen	65
	Samenwerken in een team	68
	Afwikkeling	72
7	Specifieke eigenschappen	77
	Flinto Lite	78
	Fluid.UI	79
	InVision	82
	Marvel	90
	Proto.io	91
8	De juiste tool kiezen	97
	Wat zijn je randvoorwaarden?	98
	Wat zijn jouw criteria?	100
	Welk tool werkt voor jou?	102
9	Testen met je prototype	103
	Het prototype testen	104
	Testscenario's maken	106
	Usabilitytest afnemen	107
	Gegevens vergaren	108
10	Testresultaten toepassen	111
	Showstoppers	112
	Serieuze problemen	113
	Kleine problemen	113
	Vormgeving	114
11	Als het prototype klaar is	115
	Communiceren met stakeholders	116
	Referentiemateriaal voor programmeurs	116
	Vergelijken met gebouwde app	117
	Weggooien of houden?	118
	Index	119

# Over prototypes

*Een app moet een probleem voor een eindgebruiker oplossen. Het concept voor een app is de oplossing die je biedt. Een app met een goed concept is niet per definitie succesvol. Succes hangt af van veel factoren, waarbij prototypes een hoofdrol spelen.*

## In dit hoofdstuk:

*Waarom je een prototype bouwt.*

*Waarom je app altijd concurreert met andere.*

*Hoe succes met een app is verzekerd.*

*Over verfijning van een prototype.*

*Wat de breedte en diepte van een prototype zijn.*

*Wat het uiterlijk en de interactie over een prototype zegt.*

## Waarom een prototype?

Er zijn twee belangrijke redenen om een prototype te maken. Ten eerste zijn er waarschijnlijk talloze andere apps die iets soortgelijks doen als jouw app. Voor bijna elk probleem van een gebruiker is al eens een oplossing verzonnen. En mocht je stuiten op een nieuw of nog niet eerder opgelost probleem, dat zijn er op korte termijn andere apps die jouw lichtend voorbeeld volgen en precies hetzelfde doen. Je app moet altijd concurreren met andere apps.

Ten tweede kan jouw concept ondergesneeuwd raken in een onbruikbare userinterface. Het ongecontroleerd toevoegen van nieuwe functionaliteit ondermijnt het oorspronkelijke concept en maakt een app onoverzichtelijk en nodeloos complex. Ondanks goede bedoelingen wordt de oplossing bedolven onder omslachtige manieren van werken, overbodige userinterface-elementen, ingewikkelde interactie of overdadige esthetiek.

Concurreren kan door een hele mooie userinterface te maken. Door alle elementen van app op te poetsen, aan te scherpen en te verfraaien, maak je een betere indruk dan je concurrenten. Een betere aanpak is om de gebruikerservaring te optimaliseren. Door je app handiger, sneller en slimmer te maken; door beter aan te sluiten op het *mental model* van je eindgebruiker. Mooi is mooi, maar beter is beter.

Als je een betere app kunt maken dan je concurrenten, hebben gebruikers argumenten om positieve feedback te geven. Het besparen van tijd wordt geapprecieerd; een logische opbouw wordt geprefereerd. Bovendien blijven ze een tool gebruiken die een probleem doelmatig en doeltreffend oplost. De beste userinterface voor de taak levert de beste gebruikersreacties op. Als je er ook nog een mooie app van kan maken, is mooi ook beter. Daarmee is succes van een app verzekerd.

## Hoe bereik je succes?

Door in de ontwerpfase prototypes te itereren, met eindgebruikers te testen en de testresultaten te integreren in het ontwerp ben je op weg naar een succesvolle app. Je eerste ontwerp is een goede gok, zelfs als het gebaseerd is op onderzoek naar je doelgroep. Het tweede ontwerp komt in de richting, begint iets te betekenen voor je gebruiker. Opeenvolgende versies zullen geleidelijk de juiste oplossing met het juiste interactiemodel en de juiste userinterface opleveren. Het is een bewezen methode om kwalitatief goede software te maken. Ga niet uit van je eigen ervaring of voorkeuren. Als je je prototypes voorlegt aan testers, zal het je verbazen hoeveel aannames je hebt verwerkt in je ontwerp. Bouw geen app op basis van jouw aannames over je doelgroep.

Overgangen en animaties van de userinterface zijn belangrijke onderdelen van de interactie in een app. Statische methodes om een userinterface vast te leggen, bieden onvoldoende houvast om bewust na te denken over bewegingen op het scherm. In een prototype zijn wel mogelijkheden om overgangen en animaties te simuleren en daarmee presenteer je een veel realistischer beeld van het eindresultaat.

Een prototype is waardeloos als je het niet gebruikt om je app te testen op eindgebruikers. Dit houdt in dat je je doelgroep benadert met een simulatie van je app op het device dat ze zelf gebruiken. Dat je hun reacties registreert, hun problemen noteert en hun feedback waardeert. Aan resultaten van gebruikerstesten valt niet te tornen. Je leert wat niet werkt, dat verander je; je ziet vanzelf wat niet begrepen wordt, dat verbeter je; ten slotte ontdek je wat je testpersoon echt waardeert aan je app, dat behoud je. Uiteindelijk heb je een efficiënte én effectieve app: die doet wat die moet doen. Handig, snel en slim, bij voorkeur mooi vormgegeven. Dat is wat eindgebruikers met sterren en aankopen waarderen.



## Verfijning van een prototype

Prototypes kunnen in meer of mindere mate worden uitgewerkt. In sommige gevallen is het moeilijk om een prototype te verfijnen en in andere gevallen is te veel verfijning juist ongewenst. Prototypes zijn in te delen in 'high fidelity' en 'low fidelity'. Een high-fidelity-prototype is ver uitgewerkt en benadert het eindresultaat of is er zelfs niet van te onderscheiden. Een low-fidelity-prototype is schetsmatig en ziet er niet uit als het eindresultaat. Een paper-prototype is een low-fidelity-prototype, een prototype van een onlinetool is meestal high fidelity.

In het boek *Paper prototyping* van Carolyn Snyder gebruikt de schrijfster vier andere dimensies om prototypes te beschrijven:

- **Breedte** Het aantal functies van de definitieve app, uitgedrukt in een percentage, die in een prototype zijn uitgewerkt, wordt de breedte van een prototype genoemd. Een breed prototype is completer en lijkt meer op het eindresultaat. Het is meer werk om te maken.
- **Diepte** De mate waarin systemen onder de oppervlakte van de userinterface zijn uitgewerkt, bepaalt de diepte van een prototype. Details en betrouwbaarheid van een prototype zeggen iets over de diepte ervan. Als foutafhandeling en uitleg is aangebracht in het prototype is het diep. Het is minder diep als het de keuzes van gebruikers in bepaalde scenario's negeert of vastloopt bij bepaalde keuzes. Als een prototype onvoldoende diep is, kan een testgebruiker niet afwijken van een specifiek scenario, zonder in problemen te komen. Prototypes van onlinetools kunnen diepte hebben, maar apps zijn van zichzelf al niet heel diep.



De breedte en de diepte bepalen de mate waarin de testpersoon het prototype kan onderzoeken. Usabilityproblemen kunnen onontdekt blijven als een gebruiker niet alle functies van de app kan bereiken of niet alle functies in het prototype zijn uitgewerkt.

- **Uiterlijk** In hoeverre het prototype de beoogde vormgeving van de app weergeeft, zoals lettertype, kleur en afbeeldingen, zegt iets over het uiterlijk ervan. Een paperprototype kan een hoge score voor uiterlijk krijgen als het gemaakt is op basis van afdrucken van het grafisch ontwerp voor de app. Een klikmodel in zwart-wit met alleen lijnen en een onopvallend lettertype kan minder op het eindresultaat lijken dan het paperprototype, dus een lage score krijgen. In het begin van het project is een schetsmatig prototype belangrijk om creatieve inbreng van stakeholders te krijgen. Als veel ontwerpkeuzes zijn gemaakt, is een prototype dat niet te onderscheiden is van het eindresultaat geschikter. Prototypes van onlinetools hebben een hoge score voor uiterlijk.
- **Interactie** De interactiedimensie van een prototype wordt bepaald door de mate waarin een prototype waarheidsgetrouw de in- en uitvoermogelijkheden van het beoogde resultaat weergeeft. Reactietijden en animaties zijn onderdeel van deze dimensie, maar denk ook aan geluiden en pushberichten. Onlinetools genereren prototypes die een hoge score voor interactiviteit krijgen. Meer wat betreft structuur en navigatie dan interactie binnen de schermen. Offlinetools als Axure kunnen wel interactie binnen een scherm simuleren, bij onlinetools moet je er wat harder voor werken.

De functie van een prototype is in eerste instantie om ervan te leren. Een onlineprototype voor een app kan niet gebruikt worden als basis voor de code van een softwareproject. Je kunt moeilijk aan de resulterende code komen en als dat al lukt, is die nutteloos voor je app. Daarnaast kunt je het uiteindelijk ontwerp ermee communiceren en controleren. Het prototype kan – mede



doordat het online is – goed gebruikt worden om programmeurs te instrueren en te verifiëren bij het definitief bouwen van de app.