



Introductie - Perspectieven

Een techneut ziet een bedrijf als een product met functionaliteiten. Om alles werkend te krijgen, moet deze functionaliteit logisch en compleet zijn.

Een designer ziet een bedrijf als een product waar eindgebruikers blij van worden. Het is belangrijk dat design en esthetiek voor de juiste beleving zorgen. Dat alles gebruiksvriendelijk is en dat gebruikers een goed gevoel van krijgen.

De verkoper ziet een bedrijf als succesvol reageren op kansen in de markt. Zien wat de trends zijn en daarop inspelen. Je moet de wensen en pijnpunten van je klanten leren kennen, zodat je ze kan helpen met een oplossing.

Al deze perspectieven helpen elkaar:

Rol	Perspectief	Kracht	Valkuil
De Techneut	<i>Functionaliteit</i>	Met aandacht voor detail en volledigheid een werkend product maken	Verliest veel tijd door teveel functies te maken die niemand begrijpt of nodig heeft.
De UX-Designer	<i>User Story:</i> Beleving vanuit perspectief eindgebruiker	Distilleert essentie van wat belangrijk is voor eindgebruiker en zorgt voor fantastisch product.	Vergeet dat meerwaarde voor de eindgebruiker niet altijd meerwaarde (inkomsten) voor het bedrijf betekent.
De Verkoper	<i>USP:</i> Meerwaarde product voor klant	Heeft contact met de markt (betalende klant) en weet precies wat nodig is om inkomsten te genereren.	Verkochte product heeft hoge kosten of is niet haalbaar om te maken.

Ondanks onze verschillen hebben we wél een gemeenschappelijk doel: We willen allemaal een succesvol bedrijf met een fantastisch product.

Hoe kunnen we samenwerken om dit te realiseren?

We hebben hiervoor een gezamenlijk perspectief nodig: één taal en methode. Daarvoor introduceer ik een concrete uitwerking van het *Lean Startup* gedachtegoed.

Het "Lean Startup" perspectief in een notendop

Waar andere perspectieven zeggen:

"Hoe maak ik de beste functies?" (techneut),

"Hoe ontwerp ik fantastische gebruikerservaring?" (UX-designer) of

"Hoe schep ik meerwaarde voor de klant?" (verkoper)

Daar zegt het "Lean Startup" perspectief:

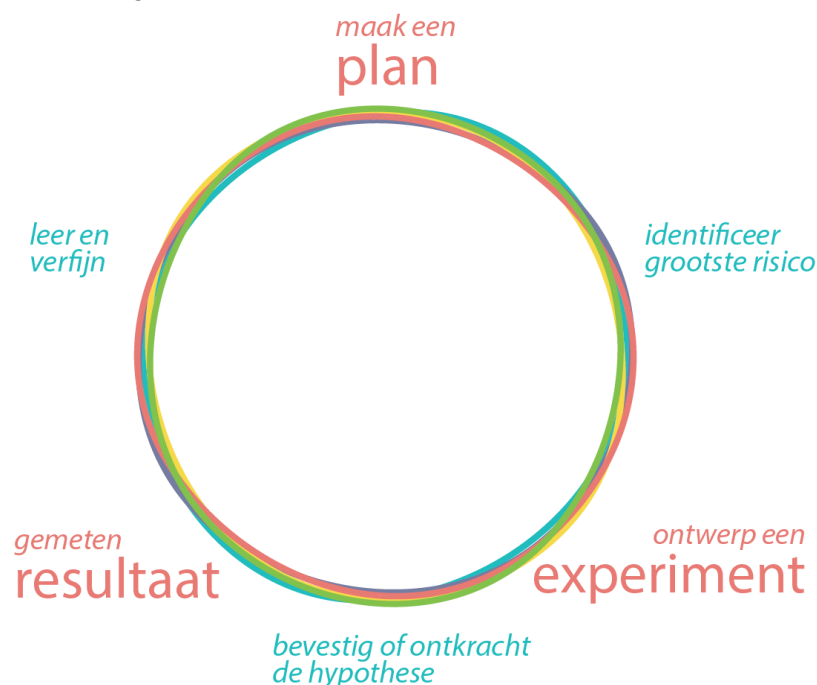
"Hoe leer ik zo snel mogelijk over de meest riskante gedeeltes van mijn bedrijf?"

Het Lean Startup zag dat het soms heel erg lang kon duren voor een bedrijf feedback krijgt op het bedrijfsplan. Hierdoor kwamen onjuiste aannames veel te laat boven water. Dit leidt tot verspilling van tijd, energie en geld. Zo kan het gebeuren dat een volledig werkend product wordt gebouwd om vervolgens te ontdekken dat er geen vraag vanuit de markt naar is.

De oplossing van de Lean Startup is het bedrijfsplan te verbeteren door zo snel mogelijk te leren over de meest riskante gedeeltes van het plan. Dit gebeurt door middel van een systematische manier van experimenteren.

Samengevat is het Lean Startup perspectief in één zin: *"Voorkom verspilde moeite door met systematische experimenten zo snel mogelijk te leren over de meest riskante gedeeltes van je bedrijfsplan"*

Dit wordt gedaan in de volgende iteratie:



1. Maak een plan

Wat zijn de doelen?

Wat zijn de doelen? Hoe meten we dat? Met welke strategie proberen we dat te bereiken?

Voorbeeld:

- Investering binnenhalen

Wat is het plan?

Gebruik het "Lean Canvas" om alle aspecten van een bedrijfsplan te inventariseren:

Probleem	Oplossing	UVP <i>High-level concept</i>	Unfair Advantage	Customer Segments
<i>Huidige oplossing</i>	Key Metrics		Customer Channels	
Kosten			Inkomsten (prijsmodel)	

Acties:

1. Kopieer het "Lean Template" project in Asana
2. Organiseer een brainstorm sessie om strategie te bepalen
 - a. Wat zijn de doelen?
 - b. Hoe meten we dat?
 - c. Met welke strategie proberen we dat te bereiken?
 - d. Vul het "Lean Canvas" in.
3. Documenteer de resultaten van de brainstorm sessie.
 - a. "Het Plan" word omschreven in de taak-omschrijving van Asana
 - b. Ga naar www.leanstack.com en vul een "Lean Canvas" in. Zet de URL in Asana.

2. Wat is het grootste risico?

Met het Lean Canvas kun je op zoek gaan naar het grootste risico. Het is handig om onderscheid te maken in drie soorten risico's:

- **Klant risico's bij de "Customer Development" van het plan**
 - Hebben we een probleem gevonden?
 - Wie zijn onze klanten? (defineer klant segmenten)
 - Hoe bereiken we ze? (kanalen, **Acquisitie**, sales-funnel)
 - Waarmee overtuigen we ze? (UVP, sales-pitch, presentatie of website)
- **Product risico's bij de "Product Development" van het plan**
 - Hebben we een oplossing voor het probleem?
 - Is de oplossing haalbaar?
 - Werkt de oplossing ook echt?
 - Lukt het om eindgebruikers te **Activeren**...?
 - ... en te behouden (**Retentie**)?
- **Markt risico bij de "Business Goals" van het plan**
 - Is er concurrentie van bestaande oplossingen?
 - Is het bedrijf levensvatbaar met het huidige prijsmodel? (**Revenue**)
 - Is de sales en het product schaalbaar?
 - Hoe kunnen onze huidige klanten voor meer klanten zorgen? (**Referral**)
 - Etc

Opmerking: De **Acquisitie-Activatie-Retentie-Revenue-Referral** staat bekend als de **AARRR** piraten metrics.

Andere risico's die kunt onderzoeken zijn:

- **Product-Solution fit:** Is het product de juiste oplossing voor het probleem?
- **Value Hypothesis:** Hoe lever je waarde aan de klant?
- **Product-Market fit:** Hoe verkoop je de oplossing aan de klanten?
- **Growth Hypothesis:** Hoe zorg je dat elke klant minstens een nieuwe klant binnenhaalt?
(Dan groei je als bedrijf)

In welke vakken van het Lean Canvas zitten het grootste risico?

Waar wil je het eerste over leren?

Acties:

1. Voeg de grootste risico's waarover je wilt leren toe in Asana (onder "Het Plan"). Beperk je tot maximaal een onderwerp per persoon.
2. Passen deze risico's in het grote plan?

3. Zijn er verborgen aannames die eerst moeten worden getest? *(Bijvoorbeeld: Voordat je de oplossing bouwt om te kijken of die werkt, kun je valideren of klanten deze oplossing ook willen hebben)*

3. Voer een experiment uit

Een experiment begint met een hypothese. Deze moet aan de volgende eisen voldoen:

- Verwachte uitkomst: Er moet een concrete, verwacht uitkomst worden genoemd.
- Falsifiable: Het moet mogelijk zijn om het tegendeel van een aanname te bewijzen.
- Repeatable: Het experiment moet gekoppeld zijn aan een actie die herhaald kan worden.
- Controle groep: Heb je een controle groep? Of een nulmeting?
- Tijd-gebonden: Er moet een einddatum op de verwachte uitkomst zitten.

Verder geldt als richtlijn: *valideer kwalitatief* en *verifieer kwantitatief*. Beslissingen maken op basis van harde cijfers klinkt ideaal, maar het is vaak erg lastig om snel en efficiënt te meten. Voor een website heb je bijvoorbeeld voldoende bezoekers nodig - en waar haal je die vandaan?

Lees ook: <https://leanstack.com/7-habits-for-running-highly-effective-experiments/>

Voorbeelden:

- Laat tijdens een interview een klant een commitment doen (inschrijven test-lijst, toezegging betaling, etc)
- Maak een sales-website en kijk of er binnen een bepaalde tijd een aantal aanmeldingen komt.
- Lanceer een product en kijk welk percentage binnen 2 weken een taak voltooid.

Acties:

1. Kopieer het voorbeeld experiment in Asana. Vul de vragen in de omschrijving in.
2. Realiseer de deliverables (prototype, mock-up, website, product, etc)
3. Voer het experiment uit (meet aanmeldingen, voer een interview, etc)
4. Documenteer de resultaten (in de omschrijving van de taak in Asana)
5. Plaats je conclusie bij "Het Plan" zodat je snel alle inzichten kunt lezen.

Hoe nu verder?

De "Lean Startup" methodiek zit op het niveau van bedrijfsstrategie. Dat zorgt dat wij gefocused kunnen blijven op wat echt belangrijk is. De Lean Startup methodiek geeft ook een concrete invulling voor de *metrics* en *klant interviews*.

De "Lean Startup" methodiek geeft echter geen invulling aan productontwikkeling. Experimenten zullen vaak een product opleveren: een software product, een (sales)website, een design, een mock-up, een prototype, etc.

We hebben te maken met de volgende processen:

- De rechterkant van het canvas - de markt:
 - Sales & Marketing
- De linkerkant van het canvas - het product:
 - Design (van concept naar design, specificaties opschrijven)
 - Techniek (van design naar werkend product, specificaties implementeren)

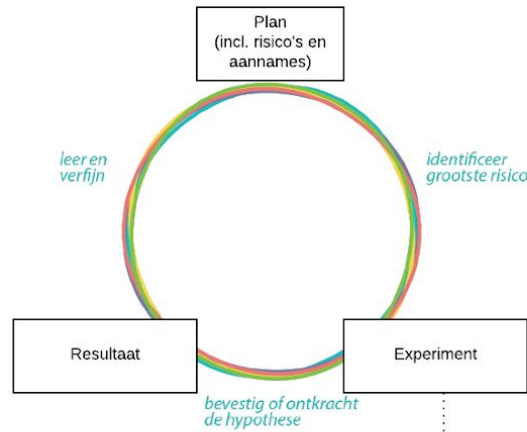
Voor Techniek stel ik de Agile/Scrum methodiek voor.

In tegenstelling tot andere methodieken ligt de nadruk op snel kunnen reageren op wensen die steeds veranderen. Dat past perfect bij experimenten die snel kunnen veranderen.

Design kan worden meegenomen in de Agile/Scrum methodiek, of er kan een aparte methodiek voor worden toegepast. (Design Sprint?)

Zie ook het plaatje op de volgende bladzijde

Lean Startup



Deliverable
Product,
Prototype,
Mock-up,
etc.

Design Sprint

Persona's

Design Briefing Meeting

User Story

ontwerpen

Wireframe /
Ontwerp

Agile/Scrum

Aanvullende SCRUM
informatie toevoegen

Product Backlog

informeert

Sprint Planning Meeting

Sprint Backlog

implementeren

Product
geïmplementeerd

release

Product Demo Meeting

test FAAL

Product
klaar voor test

test OK

Product
LIVE

release

Product
werkt

**Sprint
Retro Meeting**