UX Design

# UX Proces

Diagram

Description automatically generated

## Discover

* Briefing: info verzamelen
* Stakeholders interviews
* Gebruikersonderzoek
  + Gebruikers en gebruikersbehoeftes in kaart brengen
  + Enquêtes
* Concurrentie analyse
* Positionering

## Define

* Key question
* Scenario’s
* User stories
* Red Routes
* Functionaliteiten
* Minimum Viable Product (MVP)
  + Welke features zijn minstens nodig om het probleem dat de doelgroep heeft op te lossen
  + Eventueel aan de hand van een product matrix

Chart, treemap chart

Description automatically generated

## Design

* Informatie architectuur
* Flows
* Schetsen
* Wireframes
* Prototype
* UI-design
* Interaction design
* User testing

## Implement

* Frontend
* Backend
* Development
* (user testing)

## Launch

* Lancering
* Evaluatie
* User testing
* User reviews

# Discover

## Briefing

= Het moment waarop de klant zijn problemen, vragen en verwachtingen meedeelt.

* Aandachtspunten:
  + Actief luisteren
  + Documenteren
  + Vragen stellen:
    - Budget
    - Doelen
    - Achtergrondinfo
    - Deadlines
    - Info over het team
    - Vervolgtraject
* Info verzamelen:
  + Wat is het product
  + Product grondig doornemen
  + Verdiepen in sector/onderwerp
  + First impressions documenteren
  + Wie zijn de stakeholders
    - Een persoon of organisatie die invloed ondervindt( +/-) of zelf invloed kan uitoefenen op een nieuw product/project
  + Wij zijn bestaande/potentiële gebruikers

## Interviews

* Kwalitatief onderzoek
  + Gericht op verkrijgen van info over wat er leeft onder stakeholder groep en waarom

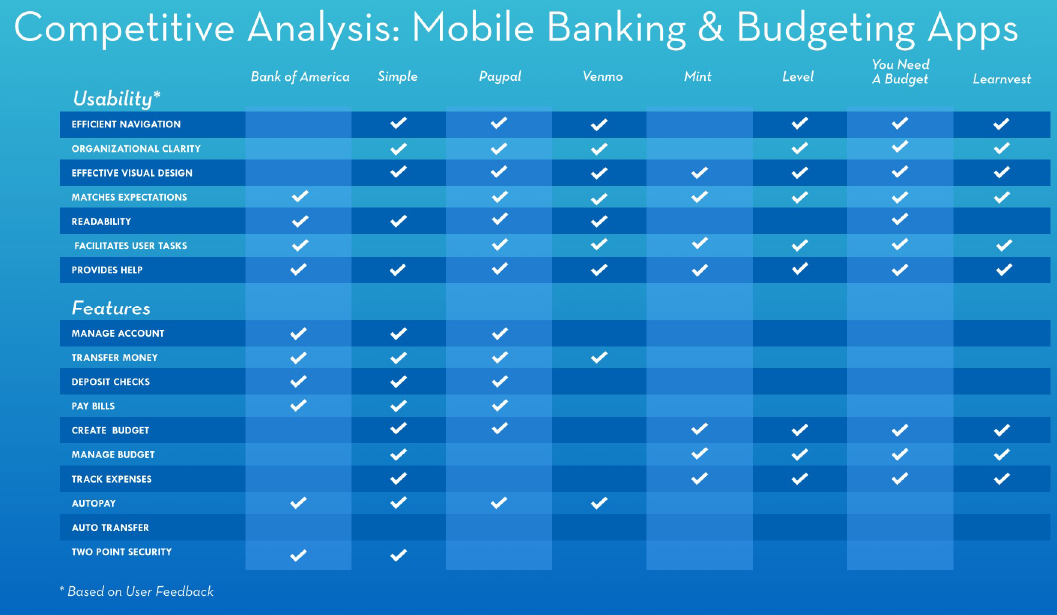
## Stakeholders interviews

* Gesprek met stakeholders in een project, zowel intern als extern, om zo inzichten over hun doelen te verzamelen.
* Prioritaire functies en Key Performance Indicators (KPI’s) opstellen
  + KPI: Variabelen om prestaties van ondernemingen te analyseren
    - Sessie lengte
    - Downloads
    - Opbrengst

## Gebruikersonderzoek

* Gebruikers & gebruikersbehoeften onderzoeken
* Wie zijn mogelijke gebruikers en wat proberen ze te doen?
* Hoe doen ze dat nu?
* Problemen en frustraties?
* Gebruikersdoelen:
  + Beschrijvingen van eindtoestanden die gebruikers willen bereiken
* Wat nodig om hun doel te bereiken?
* User Feedback verzamelen
* User Interviews
* Ratings en reviews
* Enquêtes
  + Kwantitatief onderzoek
    - Kwantitatief onderzoek biedt cijfermatig inzicht en geeft antwoorden op vragen di in termen van hoeveelheid kunnen worden uitgedrukt
  + Nadelen:
    - Grote steekproef nodig om een representatief resultaat te bekomen
    - Weinig controle
    - Geen spontane reacties
  + Voordelen:
    - Makkelijk te verwerken
    - Geen invloed op de resultaten
  + VB: DeLijn App
    - 9850 antw op 4 dagen
    - Helft is student
    - Meestgebruikt :
      * Welke route
      * Hoelaat bus

## Concurrentieanalyse

* Wie zijn concurrenten
* Hoeveel concurrenten
* Waar bevinden concurrenten zich (geografisch & positionering)
* Welke functionaliteit bieden concurrenten, zijn deze user friendly
* Wie zijn hun gebruikers
* Wat kunnen we van hen leren (wat is +/-)

## Positionering als product

* Wat is het product
* Voor wie
* Waar en wanneer wordt gebruikt (toptaken)
* Value proposition
* Wat maakt uniek (USP: Unique Selling Poroposition)

# Define

## Key question

* Kernvraage
  + Hoe kunnen we…?
  + Is er een alternatief om…?
  + Is het mogelijk om…?
  + Wat als…?
* Bv.: Hoe kunnen we ervoor zorgen dat passagiers hun halte niet meer missen?

## Doelgroep

* Het deel van de bevolking waarbij een gedragsverandering tot stand gebracht zou moeten worden.
* Heeft aantal relevante gemeenschappelijke kenmerken

### Doelgroep bepalen

* Meerder doelgroepen kunnen
  + Bv.: studenten die op kot zitten en kotbaas
* Meeste bedrijven hebben een algemene doelgroep
  + Wordt opgesplitst in subgroepen op basis van kenmerken
* Specifieker = makkelijker af te stemmen

Algemeen

* Leeftijdscategorie
* Geslacht
* Nationaliteit
* Regio/Woonplaats
* Socio-economische achetrgrond
* Culturele achtergrond
* Taal
* Educatie

Professioneel

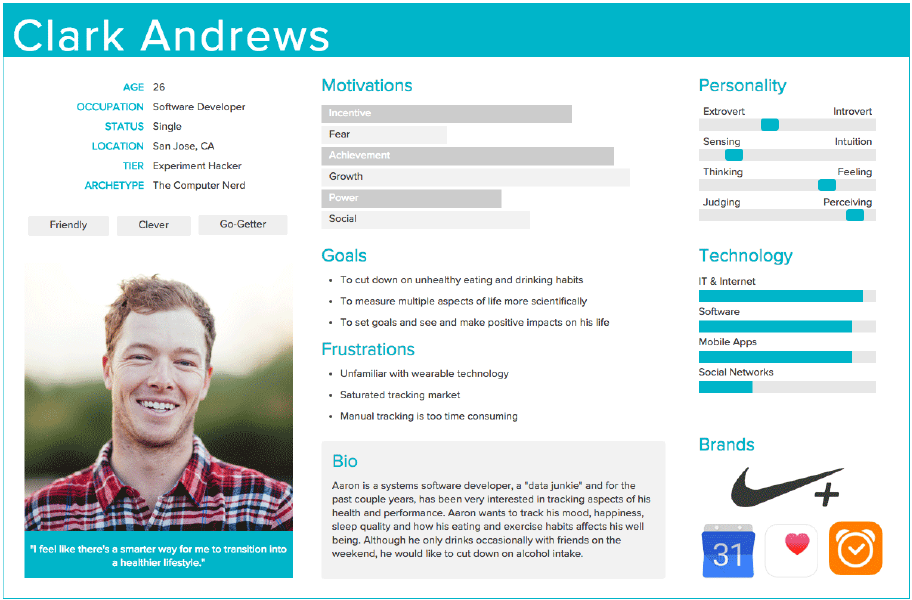
* Opleidingsniveau
* Beroep
* Functie
* Sector
* Technische kennis & mediaprofiel

Persoonlijk

* Waar ligt doelgroep wakker van
* Hoe besteden ze vrije tijd
* Wat zijn behoeftes en dagelijkse bezigheden
* Hoe ziet aankoop-/verbruiksgedrag eruit

Persona’s

= Een karakterisering van een bepaald type gebruiker. Fictief personage dat je doelgroep representeert.

* Algemeen
  + Naam
  + Foto
  + Quote: Een aanhaling of citaat, ofwel een weergave van wat iemand gezegd heeft.
* Demografisch
  + Leeftijd
  + Geslacht
  + Woonplaats
  + Regio
  + Beroep
  + …
* Karakter kenmerken
  + Karaktereigenschappen bv.:
    - Ongeduldig
    - Stipt
    - Kieskeurig
    - …
* Frustraties
* Doelen

## 

## Scenario’s

= Fictief verhaal of scenario om je in te leven wanneer een gebruiker in aarnraking komt met je product en wat hij/zij dan moet kunnen doen.

* Inlevingsvermogen
* Empathie
* Om functionaliteiten te bedenken
* Om functionaliteiten af te toetsen aan een real life scenario
* Aan alles gedacht?

## Taakanalyse

* Verzamelen functionaliteiten
  + Gebruikersonderzoek
  + Concurrentie analyse
  + Scenario’s
* Brainstormen
* Categoriseren
* « high level » taak ontleding
* Gedetailleerde taak ontleding

## Red routes

* Frequente en belangrijke taken
* Complete activieteiten
* Geen simpele taken
* Bestaan uit verschillende stappen/schermen
* Hoeveel gebruikers hebben deze functie nodig?
* Hoe vaak zullen gebruikers deze functie gebruiken
* Dit zijn de taken die je goed wil designen en testen

## Minimum Viable Product

= MVP

* Welke functionaliteiten moeten er zeker inzitten om de key question, het probleem dat de doelgroep heeft, op te lossen
* Prioriteiten

**MoSCoW**

* **M:** Must Haves:
  + Deze eisen moeten in het eindresultaat terugkomen, zonder deze eisen is het product niet bruikbaar
* **S:** Should Haves:
  + Deze eisen zijn zeer gewenst, maar zonder is het product wel bruikbaar
* **C:** Could Haves:
  + Deze eisen zullen enkel aan bod komen indien genoeg tijd
* **W:** Won’t Haves:
  + Deze eisen zullen in dit project niet aan bod komen maar kunnen in de toekomst interessant zijn

# Design

## Informatie Architectuur

* Content/informatie organiseren en structureren van websites en applicaties
* Zodat gebruikders zich gemakkelijk aanpassen aan de functionaliteit van het product
* Zodat gebruikers alles vinden zonder moeite
* Slecht georganiseerde content maakt navigatie moeilijk, gebruikers verdwalen en raken gefrustreerd
* IA is de blueprint van de design structuur
* Basis voor het plannen van een navigatie systeem

### Organization systems

* Hoe kunnen we best info categoriseren en structureren
* De groepen of categorieën waarin is verdeeld
  + Hiërarchisch (boomstructuur

Chart, box and whisker chart

Description automatically generated

* + Sequentieel

A picture containing clipart

Description automatically generated

* + Matrix

Diagram

Description automatically generated

### Labeling systemsHoe informatie weergeven

* Label: stuk tekst
* Tone of voice
* Terminology
* Vakjargon of niet?

### Navigation systems

* Hoe gebruikers browsen of bewegen door de hiërarchie of structuur van informatie
* De reeks acties en technieken die gebruikers door de app of website leiden
* Via de interface door interactieve elementen zoals buttons, links, tabs, menu’s, zoekfunctie, …

### Searching systems

* Hoe zoeken gebruikers informatie
* Scannen van tekst
* Voor sites of apps met heel veel info
  + Zoekfunctie
  + Filters

A picture containing timeline

Description automatically generated

## Flows

### User Flow

* Visualiseren een pad dat de gebruiker moet afleggen om tot een bepaald resultaat te komen
* Welke schermen heb je nodig om een taak af te werken
* Kan afsplitsingen, beslissingen en loops bevatten
* Triggert ook weer nadenken tot bepaalde functionaliteiten en schermen
* Makkelijk om te zien of een proces compleet of afgewerkt is

Diagram

Description automatically generated

Diagram

Description automatically generated

## Schetsen

* Graphical user interface, text, application

  Description automatically generatedRuw uittekenen van schermen
* Hoeft niet mooi te zijn
* Enkel belangrijke labels en acties
* Makkelijk om te brainstormen

## Diagram Description automatically generatedWireframes

* Structuur
* Inhoud
* Hiërarchie
* Functionaliteit
* Gedrag
* Bespaart tijd en kosten
* Bevordert communicatie met klant
* Visualiseert en concretiseert ideeën
* Verduidelijkt de functionaliteiten

Wireframes in MCT

* Grijswaarden
* Geen Lorem ipsum tekst
* Afbeeldingen hoeven nog niet
* Volgens Material design guidelignes voor Android

# Usability testing

= Het observeren van echte gebruikers terwijl ze een product gebruiken. Het helpt om te zien waar gebruikers moeite mee hebben en wat ze leuk vinden.

## Waarom usability testing

* Scripted test met scenario’s of taken
* Hoe gaan gebruikers effectief om met ons product
* Techniek om product te evalueren
* Afstemmen of je juist zit
* Meet bruikbaarheid of gebruiksgemak om bepaald doel te halen
* Hulpmiddel om te zien of design werkt of niet
* ≠ indicator voor falen maar hulp om bij te sturen
* Helpt uiteindelijk om gebruikers te behouden

**Zo vroeg mogelijk testen**

* Als je te lang werkt zonder feedbackloop is de kans groter dat je een belangrijke verandering moet aanbrengen nadat je het product op de markt hebt gebracht
* Als je energie kunt investeren om vroeg te leren en om te voorkomen dat er überhaupt problemen optreden, zul je later veel tijd besparen
* Wireframes en prototype testen
  + Belangrijk om die context ook mee te geven

## Doelen usability testing

* Waarom wil je je product testen?
* Wat willen we leren?
* Meest voorkomende doelen bij testing
  + Kijken of gebruikers bepaalde taken met succes kunnen voltooien (bv.: product kopen of info zoeken)
  + Kijken hoelang het duurt om specifieke taken te voltooien
  + Ontdekken of gebruikers tevreden zijn met een product
  + Identificeren van wijzigingen die nodig zijn om de tevredenheid te verbeteren
* Bereid vragen en taken voor
  + Realistische taken die een duidelijke actie zijn
    - Bv.: een checkout doorlopen
  + Red Routes krijgen voorang
  + Beschrijf taken duidelijk
  + Geef een scenario, geen instructie
  + Test de taken zelf, bugs uit je prototype halen tellen niet als test

## Test en testgroep bepalen

* Testpersonen zoeken
  + De mensen die aan de test deelnemen, moeten representatief zijn voor de doelgroep
  + Niet enkel vrienden en familie
  + Onafhankelijke en onbevooroordeelde gebruikers
  + Kwaliteit, niet kwantiteit
* Opnieuw testen
  + De eerste testresultaten tonen vooral de zwakheden in je design
  + Los deze op en test opnieuw
  + Nieuwe inzichten
* Persoonlijke tests
  + Voer de test zo uit zodat de deelnemers zich op hun gemak voelen en eerlijke feedback geven
  + Zorg ervoor dat ze niet het gevoel hebben dat **zij** worden getest
  + Beoordeel hen niet
  + Laat ze zo openhartig mogelijk zijn
  + Luisteren en observeren, niet leiden of corrigeren
  + Niet onderbreken, uitleggen of verbeteren
  + Laat hen zelf het pad kiezen, waar ze op klikken
  + Think-aloud
  + Geef context mee in het begin (wireframes of echt afgewerkt product)
  + Neem alles op

## Natuurlijke (en bijna-natuurlijke) tests

= tests waarin mensen van nature uit het product gebruiken

1. A/B testing
   1. Call to actions
   2. Headlines
   3. Formulieren
   4. Prijzen en aanbiedingen
   5. Beeldne op landingspagina’s + productpagina’s
   6. Hoeveelheid tekst per pagina
   7. Vooral bij bestaande producten
   8. Run beide varianten op hetzelfde moment
   9. Test genoeg mensen
   10. Nieuwe gebruikers
   11. Wees consistent met variaties op alle pagina’s
2. First click testing
3. Eye tracking test
4. Beta testing

# Usability Heuritics

Table

Description automatically generated with medium confidence

Graphical user interface

Description automatically generated

### 10 Usability Heuritics for User Interface Design

1. **Visibility of system status**

The design should always keep users informed about what is going on, through appropriate feedback within a reasonable amount of time.

1. **Match between system and the real world**

The design should speak the users’ language. Use words, phrases, and concepts familiar to the user, rather than internal jargon. Follow real-world conventions, making information appear in a natural and logical order.

1. **User control and freedom**

Users often perform actions by mistake. They need a clearly marked "emergency exit" to leave the unwanted action without having to go through an extended process.

1. **Consistency and standards**

Users should not have to wonder wether different words, situations or actions mean the same thing. Follow platform and industry conventions. Use UX-patterns.

1. **Error prevention**

Good error messages are important, but the best designs carefully prevent problems from occurring in the first place. Either eliminate error-prone conditions, or check for them and present users with a confirmation option before they commit to the action.

1. **Recognition rather than recall**

Minimize the user’s memory load by making elements, actions and options visible. The user should not have to remember information from one part of the interface to another. Information required to use the design (e.g. field labels or menu items) should be visible or easily retrievable when needed. ‘Suggestions’ and ‘recently viewed’ are always helpful.

1. **Flexibility and efficiency of use**

Shortcuts -hidden from novice users – may speed up the interaction for the expert user such that the design can cater both to inexperienced and experienced users. Allow users to tailor frequent actions.

1. **Aesthetic and minimalist design**

Interfaces should not contain irrelevant or rarely needed information. Every extra unit of information in an interface competes with the relevant units of information and diminishes their relative visibility.

1. **Help users recognize, diagnose and recover from errors**

Error messages should be expressed in plain language (no error codes), precisely indicate the problem and constructively suggest a solution. Humour is an easy way of diminishing worries in users when they see an error.

1. **Help and documentation**

It’s best if the system doesn’t need any additional explanation. However, it may be necessary to provide documentation to help users understand how to complete their tasks. Keep it as short and efficient as possible, focus on the users’ task and make sure the documentation is easy to find if needed.

# Design patterns

* Herkenbare interacties
* Die veel voorkomende design problemen oplossen
* Zodat gebruikers niet iedere keer opnieuw een interactie moeten analyseren
* Verminderen dus tijd en moeite
* Draagt bij tot de UX
* Bespaart de designer veel tijd
* Don’t reinvent the wheel
* Tot stand gekomen door tijd, onderzoek en ervaring

Bv.:

|  |  |
| --- | --- |
| Logo links bovenaan | Graphical user interface, text, application  Description automatically generated |
| Beschikbaarheid personen | Graphical user interface, text, application  Description automatically generated |
| Rood-geel-groen | A picture containing light, electronics  Description automatically generated |
| Pull to refresh | Graphical user interface, text, application, chat or text message  Description automatically generated |
| Numpad |  |

**Jakob Nielsen:**

**“No website is seen in isolation: users come to your site expecting things to work the same way they are already used to.”**

**A picture containing text, person, old, screenshot

Description automatically generated**

### User Interface design patterns

* UI problemen oplossen
* Input, forms
* Navigation
  + Hoe ga je van ene naar andere pagina/scherm
  + Gevolg van informatie architectuur/hiërarchie
  + 3 navigational directions:
    - **Lateral**:

Diagram

Description automatically generated

* + - * Bewegen tussen schermen/pagina’s die op hetzelfde niveau staan
      * Top level navigation
      * Dedicated navigation components
        + Website menu
        + Hamburger icon + navigation drawer (5+ items)
        + Bottom navigation (2-5 items)
        + Tabs
        + …
    - **Forward**

**Diagram

Description automatically generated**

* + - Bewegen tussen schermen op opeenvolgende niveaus
    - Downward
      * Van parent naar child
      * Content based navigation
      * Steeds dieper in de hiërarchie
    - Sequentially
      * Doorheen een flow
    - Directly
      * Van 1 scherm naar gelijk welk scherm in de app of website
      * Voorwaarts overslan van de hiërarchie
      * Mega dropdowns
      * Links op de homepage
      * Search
    - **Reverse**

**Diagram

Description automatically generated**

* + - Achterwaards bewegen tussen schermen
    - Achterwaards chronologisch
      * In de geschiedenis van recente bezochte pagina’s
      * Dus niet volgens hiërarchie
      * Back button van webbrowser
      * Back button in Android nav bar
    - Upwards
      * Niveau terugkeren
      * Wel volgens hiërarchie
      * Totdat terug op top level of homescherm komt
      * Op alle child screens die een parent hebben
      * Back button in app bar
      * Breadcrumb op websites
* Data weergeven
* Social
* Shopping
* …

### Parent-child

#### Parent screens

* Overzicht
* Lijst van items
* User moet keuze maken
* Items toevoegen

#### Child screens

* Detail
* Meer acties
  + Edit
  + Delete
  + Add to cart

### Modality

Graphical user interface, application

Description automatically generated

#### Modal screens

Graphical user interface, application

Description automatically generated A picture containing text, indoor, screenshot

Description automatically generated

* Child
* Disables the parent
* Ovrelays the parent
* Require users to complete an action
  + Save
  + Cancel
  + Ok
* Haalt users uit de general flow

### Persuasive design patterns

* Shape, circle

  Description automatically generatedOvertuigende design patronen
* Psychologisch inspelen op gebruikers
* Gedrag beïnvloeden
* Engagement vergroten
* Om bv mensen te laten terugkomen
* VB:
  + Feedback loops

Graphical user interface, application, website

Description automatically generated

* + Triggers

Graphical user interface, text, application, chat or text message

Description automatically generated

## Dark patterns

= een gebruiker snel misleiden om iets te doen + achteraf de gebruiker het moeilijk maken er terug onderuit te komen

* Roach Motels

The roach motel design is a dark pattern that makes it difficult to leave once you're in. An example of this would be having trouble unsubscribing from a service or mailing list after signing up for them the first time around, which was easy to do when they were introduced.

* Misdirection

Aandacht v/d gebruiker bewust op iets focussen zodat de focus niet op het ander aspect ligt.

Graphical user interface, website

Description automatically generated

* Privacy Zuckering

Vaktaal / UX gebruikers meer info delen over zichzelf dan ze eigenlijk willen.

Text

Description automatically generated

Bv.:

* Nieuwsbrief
* Account verwijderen Amazon