

# H04: Usability testing

## 1. Inleiding

Doorheen de jaren is het landschap van gevraagde vaardigheden van een IT'er heel hard veranderd. Waar vroeger vooral de puur technische skills de bovenhand hadden is er een verschuiving naar bijkomende skills die hun nut hebben bij het uitvoeren van een project. Wat vooral opvalt in de top 10 is dat ook testen een skill is die aan belang wint alsook het ontwerpen van de gebruikersomgeving en het software-ontwikkelingsproces.

Bij het ontwerpen van het software is het nodig om de verschillende modellen op elkaar allemaal af te stemmen. Concreet zijn er 3 verschillende modellen. We hebben een conceptueel model, mentaal model en een systeemmodel. Het conceptuele model is het model zoals de ontwerper het opstelt. Dit is dus het model van hoe de ontwerper denkt dat het programma er zou moeten uitzien. Het mentaal model is daarentegen een voorstelling van hoe de gebruiker denkt over hoe de applicatie zou moeten werken. Het laatste model is een systeemmodel. Dit model is hoe het systeem effectief werkt. Het is dus van grootste belang om er voor te zorgen dat die drie modellen zoveel mogelijk overeenkomen zodat de werking van het systeem zo optimaal mogelijk is. Zo is de bruikbaarheid van ons applicatie dan maximaal.

Om deze modellen te toetsen gaan we eerst na wat we precies we wensen te meten. Hiervoor moeten we een onderscheid maken tussen user experience en usability. De user experience of gebruikerservaring bevat alle aspecten die de gebruiker ervaart wanneer hij in interactie is met het product, service, omgeving of faciliteit. Usability of bruikbaarheid bevat de mate van effectiviteit, efficiëntie en tevredenheid die gebruikers ervaren bij het bereiken van specifieke doelstellingen in bepaalde omgevingen. Wij zullen in dit hoofdstuk focussen op hoe we de bruikbaarheid kunnen meten en dus ook testen.

Bij usability testing gaan we ons focussen op drie kernwaarden effectiviteit, efficiëntie en tevredenheid. Om de effectiviteit te meten moeten we dus achterhalen of het product doet wat het belooft dat het zou moeten kunnen. Kan de gebruiker dus effectief zijn doelen bereiken. Bij efficiëntie gaan we kijken hoeveel inspanning er nodig is van de gebruiker om die doelen te bereiken. Hoe snel en hoe vlot kan hij die verwezenlijken. Ten slotte is het ook belangrijk dat ze tevreden over de manier waarop ze met het product moeten werken. Alle drie deze zaken zijn dus belangrijk bij het testen van de bruikbaarheid van jouw product.

## 2. Meten

De gemakkelijkste manier om de usability te meten is om het gedrag van “echte gebruikers” waar te nemen terwijl ze echte taken uitvoeren met het product. Tijdens deze meting verzamelen we dus gegevens omtrent bepaalde criteria. Je kan echter niet op alles testen en dus ook niet oneindig veel criteria gaan meten en als maatstaf gebruiken. Hierdoor zullen we dus keuzes moeten maken op basis van tijd, middelen, budget, risico, ... . Om echter van een kwaliteitsvolle meting te kunnen spreken moeten we minstens volgende vier basiscriteria gaan meten:

- Succespercentage
- Aantal fouten per tijdseenheid
- De gemiddelde tijd
- Subjectieve tevredenheid van de gebruiker

De eerste drie criteria hierboven opgesomd kunnen eenduidige gemeten worden. De subjectieve tevredenheid van de gebruiker kan echter op verschillende manieren bepaald worden. Wij zullen deze meten op basis van de SUS methode, zie later voor meer info hierover.

Eens we deze metingen uitgevoerd hebben kunnen we de resultaten verwerken en gebruiken voor een aantal conclusies omtrent de status van onze product. In eerste instantie kunnen we op basis van de meetresultaten de vooruitgang tussen de verschillende versies van ons product gaan evalueren. Het stelt ons ook in staat om de marktpositie beter in te schatten. We kunnen dus nagaan of we beter (of slechter) zijn de concurrentie in het algemeen maar ook specifiek op een bepaald vlak. Daarnaast kunnen we deze meting ook gebruiken om sneller een stop/go beslissing te nemen. Het zorgt er dus voor dat we weten of ons product goed genoeg is om gereleased te worden of dat er eerst nog aanpassingen moeten uitgevoerd worden om het te verbeteren. Als laatste middel kunnen we de resultaten ook gebruiken om te bepalen of een projectleider een bonusverloning dient of niet.

De metingen worden op 2 verschillende manieren, en verschillende stadia tijdens het project, uitgevoerd. We hebben enerzijds formatief testen, anderzijds summatief testen.

### 3. Formatief testen

Tijdens het formatief testen zal een testdeelnemer gebruik maken van een prototype om vooraf gedefinieerde taken uit te voeren. Tijdens het uitvoeren van die taken denkt de deelnemer hardop (think aloud) en vertelt zo hoe hij denkt en welke acties hij uitvoert. De testdeelnemer werkt dus niet met een afgewerkte versie maar voert de testen uit op een product in een vroeg stadium. Dit is dus ook niet altijd op effectieve code maar kan ook aan de hand van schermafdrucken of papieren mock-ups zijn.

Om een goeie meting te kunnen uitvoeren worden de formatieve testen uitgevoerd door minstens 5 representatieve gebruikers die elk individueel getest worden om geen onderlinge invloed te hebben. Tijdens de test zal een testmoderator de deelnemers begeleiden door hen instructies te geven welke functionaliteit ze moeten uitvoeren en ook door vragen te stellen indien hun gedrag niet volledig duidelijk is of afwijkt van het te verwachten gedrag. Er is ook steeds een observator/notulist aanwezig om dat gedrag vast te leggen. Deze persoon kan eventueel ook dezelfde zijn als de testmoderator.

Een typische “think aloud” sessie ziet er als volgt uit:

1. Moderator **verwelkomt** de deelnemer en geeft uitleg over de test sessie en vraagt eventuele pre-test of demografische vragen.
2. Moderator verklaart “**hardop te denken**” en vraagt of de deelnemer bijkomende vragen heeft.
3. Moderator / Deelnemer leest het **takenscenario hardop** en begint te werken aan het scenario terwijl ze hardop denken.
4. Moderator / Waarnemers nemen **notities** van het **gedrag** van de deelnemer, opmerkingen, fouten en afronding (succes of falen) op elke taak.
5. Moderator vraagt aan het einde van de sessie **subjectieve vragen** of stuurt de deelnemer door naar een online enquête
6. Moderator **bedankt de deelnemer**, geeft de deelnemer de overeengekomen vergoeding, en begeleidt ze uit de testomgeving

Deze manier van testen is er vooral op gericht op het vinden en verbeteren van specifiek ontwerpproblemen. We willen dus onderliggende problemen ontdekken en aanpakken.

## 4. Summatief testen

Een summatieve test wordt steeds uitgevoerd op een volledig werkend product. Hierop voeren de gebruikers representatieve taken uit die overeenstemmen met de productbehoeften. Het resultaat van een meting noemen we een benchmark. Deze benchmark bevat minstens de vier basiscriteria zoals eerder besproken en stellen ons in staat om een vergelijking te maken tussen verschillende benchmarks van verschillende versies van ons product. Om de subjectieve tevredenheid, wat een van de basiscriteria is, te meten maken we gebruik van de "System Usability Scale" (SUS). Deze methode geeft ons een gemiddelde score voor die tevredenheid op basis van een vragenlijst. De vragenlijst bestaat uit twee soorten, positief en negatief, stellingen over het product. Op het einde weten we dan ook hoeveel % van de deelnemers het product zouden aanbevelen bij een vriend, % van de deelnemers die vindt dat usability echt moet aangepakt worden, ...

Om die score te berekenen geeft een deelnemer op elke stelling een waarde tussen 1 (= helemaal oneens) en 5 (= helemaal eens). Nadien wordt de gegeven waarde per stelling als volgt aangepast:

- Bij negatieve stelling:  $5 - \text{de waarde}$
- Bij positieve stelling:  $\text{de waarde} - 1$

Nadien tellen we deze nieuwe verkregen waarden allemaal op en vermenigvulden we de som met 2,5. Zo komen we een waarde uit tussen 0 en 100 die we kunnen evalueren:

- $\leq 51 = \mathbf{F}$ : usability moet topprioriteit worden
- $68 = \mathbf{C}$ : gemiddelde SUS waarde
- $> 74 = \mathbf{B}$ : waargenomen bruikbaarheid van 70% van alle geteste producten.
- $> 80,3 = \mathbf{A}$ : Dit is het punt waar gebruikers het product aanbevelen aan een vriend

Summatief testen is niet zoals formatief testen op gericht om ontwerpproblemen te vinden en verbeteren maar dient vooral het volledige product te evalueren aan de hand van vastgelegde criteria.

## 5. In de praktijk

In de realiteit zal een usability test steeds door minstens 5 testers uitgevoerd worden. Deze vinden ongeveer 80% van de problemen en we kunnen er vanuit gaan dat we dan ook zeker de belangrijkste problemen gevonden zullen hebben. Het is echter niet noodzakelijk of nuttig om meer testers te gebruiken want usability testing is gedrag gedreven en het gedrag tussen deelnemers op zicht verandert niet zo heel veel. Dit betekent met andere woorden dat elke bijkomende tester geen significant extra aantal problemen zal vinden. Het kantelpunt, zie grafiek, is duidelijk waarneembaar bij 5 testers. Voor het kantelpunt is het gebruiken van een extra tester wel zinnig omdat het veel meer opbrengt dan de kost die er tegenover staat.

Aangezien een product nooit volledig in 1 keer ontwikkeld wordt maar op basis van verschillende versies iteratief en incrementeel wordt opgeleverd zal ook het proces van usability testing herhaald worden. Per iteratie worden dus usability tests uitgevoerd:

- 1<sup>ste</sup> iteratie: meest significante problemen  
→ Fix deze problemen ...
- 2<sup>de</sup> iteratie: nieuwe problemen ontdekt  
→ Fix deze problemen ...
- 3<sup>de</sup> iteratie : ...  
→ Fix deze problemen ...

- ...

Als we dan uitgaan van 4 tot 6 iteraties voor ons product klaar is en we elke iteratie 5 testers gebruiken dan zullen op het einde 20 tot 30 deelnemers ons product getest hebben. We merken echter dat in realiteit deze manier van testen niet altijd perfect uitgevoerd wordt. De meest voorkomende fouten zijn:

- Men weet niet wat men wil testen
- Het team is niet betrokken
- De verkeerde groep testers (bv.: mensen met te veel voorkennis)
- Men test het verkeerde
- Inefficiënt organiseren van de test
- Geen planning hoe de testresultaten verspreid worden
- Geen iteraties om oplossingen te testen

## 6. Voorbereiding

Voor we een usability test afnemen zijn er een heel wat zaken die we moeten voorbereiden:

- Usability Test Plan
- Persona's
- Key Tasks
- Scenario's
- Mock-up

### 6.1. Usability Test Plan

De eerste stap uit de voorbereiding bestaat er uit om alle andere stappen in kaart te brengen. Er wordt dus een document gecreëerd waarop een overzicht te zien is hoe de test zal verlopen. Op het plan zien we onder andere informatie over volgende zaken:

- Doelstellingen voor de test
- Omschrijving van de personen die zullen deelnemen aan de test
- Taken die getest worden
- Locatie en tijdstip waarop test doorgaat
- Gedetailleerd tijdschema van een sessie

### 6.2. Persona's

Een persona is een gedetailleerde beschrijving van een bepaald type gebruiker van de uiteindelijke applicatie. Per type gebruiker stellen we een persona op zodat we ons kunnen inleven in die specifieke gebruiker, zijn doelen en verwachtingen van het product. Dit helpt het team om een gemeenschappelijk begrip van de gebruiker vast te leggen en beslissingen te nemen tijdens het ontwerpproces. Zo kan het ontwikkelingsteam te focussen op de behoeften en verwachtingen van de gebruiker. Ten slotte worden persona's ook gebruikt om veronderstellingen over gebruikers weg te werken.

Om een persona op te stellen baseren we ons op eigen gebruikersonderzoek. Deze informatie is verkregen via observaties, klanteninformatie, ... . Het team werkt daarna, op basis van die verkregen informatie, persona's uit vanuit verschillende perspectieven. Zo zijn we zeker dat alle type gebruikers aan bod komen en hun specifieke noden. Deze opgestelde persona's worden expliciet gebruikt tijdens het ontwikkelproject en zijn zichtbaar voor het hele team. Aangezien persona's gecreëerd werden voor het project op basis van informatie

voor dat project zijn ze dus niet herbruikbaar maar heel specifiek. Als laatste stap worden persona's ook geïntegreerd in de testscenario's van de gebruikers.

### 6.3. Key tasks

Om gericht te kunnen testen gaan we op zoek naar de kerntaken van onze applicatie. De absolute kerntaken moeten zo snel mogelijk getest worden en zijn veel belangrijker dan de niet kritische functionaliteiten. Een kerntaak is een functionaliteit die een gebruiker absoluut met het product wil uitvoeren om zijn doel te bereiken. Om deze te achterhalen proberen we de frequente taken te identificeren, alsook de kritische of problematische taken. Per taak zoeken we dus een antwoord op 2 vragen: "Hoe vaak wordt het gebruikt?", "Door hoeveel gebruikers wordt het gebruikt?".

Brengen we deze resultaten in kaart dan krijgen we een Key Task Map. Dit is een kaart met 16 zones waaruit we dan kunnen afleiden hoe vaak een taak wordt uitgevoerd en door hoeveel mensen dat is. Bij Google Maps zal bijvoorbeeld de functionaliteit om een route te plannen stevast door iedereen uitgevoerd worden. Maar slechts weinig mensen zullen sporadisch de toepassing bedienen op basis van hun stem. Beide functionaliteiten hebben dus ook een heel andere prioriteit binnen de applicatie. In eerste instantie is het vooral belangrijk om de functionaliteiten in de "rode zone" (Altijd + iedereen, Altijd + de meerderheid, Meestal + iedereen) grondig te testen op hun usability en deze op punt te stellen.

### 6.4. Scenario's

In de verschillende testscenario's beschrijven we hoe de personages hun doelen bereiken en welke gevoelens ze daarbij hebben.

### 6.5. Mock-ups

Een mock-up is een uitgewerkte versie van de gebruikersinterface waarin duidelijk weergegeven wordt welke UI-elementen er aanwezig zullen zijn, met inhoud en opmaak, en hoe deze geplaatst worden. Als een mock-up ook echt werk spreken we van een prototype