



Skal det være enten eller?

- Et tankeeksperiment.

Når man skal til at ”stille krav” til et it-system kommer det første store valg hurtigt: hvordan skal de dokumenteres, så en udvikler eller leverandør kan forstå, hvilket system man er ude eft er? Er user stories vejen frem? Eller er use cases bedre? Eller et af de mange andre bud på metoder til formidling af krav?

Andrew Stellman beskriver i sin artikel *Requirements 101: User Stories vs. Use Cases*¹ på elegant vis forskellen på **use cases** og **user stories**: han mener kort opsummeret, at user stories er bedst til at beskrive behov og use cases er mere løsningsorienterede – dvs. de beskriver både behov og (dele af) en løsning.

Hvorfor ses de så som hinandens alternativer?

At skelne mellem kravspecifikation og løsningsbeskrivelse

Min erfaring fra mange kravspecifikationer siger mig, at problemet først og fremmest er udfordringen med at skelne mellem kravspecifikation og løsningsbeskrivelse. Mange vil mene, det ligger i navnet, men det er desværre ikke altid så simpelt i virkeligheden. I mange projekter er man ikke villig til ”kun” at beskrive den ønskede funktionalitet (og de tekniske rammer systemet skal fungere i). Man vil også gerne have lidt hånd i hanke med, hvilken løsning man ender med. **Krav kan derfor stilles på forskellige niveauer: forretningsmål, domænespecifikke krav, løsningspecifikke krav, designkrav og proceskrav.** Jo flere løsningspecifikke krav vi laver, jo mere binder vi leverandøren/udvikleren. I det hele taget er vi nok mere gearet til at tænke i løsninger frem for behov. Det er svært at skille de to ad, når man står i situationen.

¹<http://www.stellman-greene.com/2009/05/03/requirements-101-user-stories-vs-use-cases/>

Der ud over er problemet nok, at de to metoder traditionelt tænkes ind i forskellige projektudviklingsmodeller. Use cases er forbundet med den traditionelle vandfaldsmodel og user stories med agile modeller. Og begge dele anvendes i øvrigt ofte forkert, fordi de fleste kun har et overfladisk kendskab til metoderne.



User Stories

User stories er bedragerisk nemme at læse. Selv slutbrugeren kan forholde sig til dem. Men mange “glemmer” at user stories ikke kan stå alene - de skal opfattes som en påmindelse/en overskrift, der så skal sikre, at man sætter sig ned og taler om de reelle behov og de mulige løsninger. Ofte bryder man faktisk en user story med en bred beskrivelse ned i stories med mere detaljerede beskrivelser, der så igen brydes ned til noget den enkelte udvikler kan forholde sig til. Om løsningen så skitseres (bemærk: ikke ”dokumenteres på forhånd”, for det behøves ikke, når relevante parter har været del af samtalen) i form af wireframes, beslutningsmodeller, proces-tegninger, henvisninger til standarder, tasks e.a. er ikke så vigtigt. Det vigtige er, at user storiene ikke opfattes som det endelige krav og ikke skal skrives/læses som sådan - hverken af kravstiller eller leverandør.

Use Cases

På den anden side opfattes use cases som den mere sikre vej for en kravstiller, men alt for restriktive af leverandøren, da man jo kan være fristet til at opfatte det (ofte) detaljerede forløb som værende hugget i sten. De er svære at lave og svære at læse - især for slutbrugerne. Men de tvinger forfatteren/forfatterne til at tænke mange forløb igennem og måske identificere ellers glemte behov. Som regel er de jo egentlig bare udtryk for forfatterens opfattelse af en mulig løsning - og ikke nødvendigvis den bedste løsning. Detaljeringsgraden og muligheden for at angive variantflows giver en falsk sikkerhed for at alle detaljer er på plads og indholdet “bare” skal kodes ... koste hvad det vil...

Brug det, der giver mening i den enkelte projektgruppe

Metoderne tilpasses i øvrigt ofte det enkelte projekt. Jeg er ikke den store regelrytter – faktisk er jeg meget tilhænger af en pragmatisk tilgang til metodeanvendelse: Brug det, der giver mening i den enkelte projektgruppe. Men det basale skal være på plads, før man kan begynde at lave variation over kendt tema. Og alle projektdeltagere skal være klar over at man nu varierer fra normen, så man ikke læser kravene med et forkert sæt briller. Det kan give mening at putte flere detaljer i en user story, men man skal vide, at det begrænser (eller ligefrem fordyrer) den endelige løsning.



Langt hen ad vejen vil det altid være en fordel at bruge det korrekte værktøj til den enkelte opgave. Hvis du vil beskrive behov, brug user stories (eller andre relevante metoder). Hvis du vil beskrive løsningen, brug use cases (eller andre relevante metoder). Det store spørgsmål er derfor:

**Kunne man forestille sig en proces, hvor kunden udarbejder user stories og så “svarer” eller viderebearbejder udviklerne/leverandøren i form af de mere detaljerede use cases (verificeret af kunden)?
– Uanset om man er til vandfald eller agile metoder.**



Skrevet af:
Elvi R. Nissen
Senior Usability Specialist