

Agile/Scrum workshop June 4th 2012

Page to put suggestions and have preliminary discussions for the Agile/Scrum workshop on June 4th 2012.

Intro

This email was sent on May 28th:

Dag lui,

aangezien hier wat vragen over kwamen heb ik bijgesloten het 1e voorstel dat we besproken hebben met de coach/trainer.

Wat we maandag a.s. gaan doen valt onder de genoemde secties Scrum workshop/Agile requirements workshop als genoemd onder aanpak.

We hebben gekozen voor Agilix omdat zij met een programma kwamen dat enerzijds flexibel is qua inhoud en anderszijds een aantal momenten bevat van ''coaching on the job'', dwz het bij en met ons meedoen in de meetings die we hebben, zodat het meer een totaalprogramma is en geen ''cursus'' (dat is wat de andere contacten ons aanboden) waarna we het zelf maar moeten uitzoeken.

Als gezegd: dit is GEEN voorgekauwde theorieles, noch een ''SCRUM adoptie cursus''; we gaan maandag beginnen met inventariseren welke onderwerpen JULLIE vinden dat we moeten bespreken, en Cesario (de coach) zal dat dan verder oppakken. We hebben duidelijk aangegeven dat het doel van de workshop is een manier van werken te bespreken en vinden die goed past bij ons team. We vinden het wel belangrijk dat een aantal Agile componenten daarin terugkomen. Na maandag zijn er nog een aantal sessie waarbij Cesario bij onze meetings aanwezig zal zijn en ''hands on'' discussie kan leiden en verdere adviezen en tips kan geven om te zorgen dat de afgesproken manier van werken ook in de praktijk kan worden geborgd.

Als jullie vooraf alvast ideeën willen uitwisselen is dat natuurlijk welkom; ik hoor het graag, en de rest van het team waarschijnlijk ook.

---Door Nico Vermaas:---

Software ontwikkeling werkt normaal gesproken met een bepaalde lifecycle, bepaalde fasen.

In vogelvlucht zijn dat:

Analyse Fase (levert gebruikers specificaties en een functioneel ontwerp) =>
Ontwerp Fase (levert een technisch ontwerp op),
Implementatie (levert functionaliteit & documentatie op).

Soms wordt het iets anders genoemd, of soms is de indeling net iets anders, maar het komt er altijd op neer dat er verschillende perspectieven zijn, waar verschillende betrokkenen zijn, waarmee verschillend wordt gecommuniceerd.

Ik heb het nu niet over de lifecycle van het hele project, maar over deelsystemen van de software.

Een groot probleem is dat dat bij ons allemaal dwars door elkaar loopt, dat er daarom geen duidelijke producten worden opgeleverd. Daarom verloopt de communicatie chaotisch en ad-hoc, veranderen de specificaties gaandeweg en verliezen we ontzettend veel tijd met het oplossen van problemen die hierdoor veroorzaakt worden.

Wat voorbeelden van waar dit bij ons fout gaat:

- gebruikers vertellen ontwikkelaars hoe ze dingen technisch geïmplementeerd willen zien,
maar hebben daarbij niet het overzicht om te weten wat voor problemen dat in andere delen van het systeem veroorzaken.
- ontwikkelaars leggen geen eenduidige specificaties van gebruikers duidelijk en centraal vast,
waardoor gebruikers/ontwikkelaars/management regelmatig de vrijheid nemen om ze gaandeweg te veranderen.
(veranderen moet wel mogelijk zijn, maar niet zonder dat dat duidelijk vastgelegd wordt, zodat we het overzicht houden).
- specificaties en technische ontwerp worden niet centraal besproken, zodat problemen in verschillende delen van het systeem vooraf geïdentificeerd kunnen worden.
Dit deden we voorheen beter (in de OMG), maar met Agile is dit slechter geworden.
- sommige ontwikkelaars praten op eigen houtje met gebruikers, beslissen dan zelfstandig over wijzigingen in de specificaties
- de interfaces tussen de deelsystemen liggen niet goed vast en veranderen regelmatig,
waardoor de software aan de andere kant omvalt.
- Doordat het overzicht mist, is het voor de meeste ontwikkelaars ook niet duidelijk in welk deel van het systeem nieuwe functionaliteit dient te worden opgenomen.
- Gemaakte afspraken worden niet onthouden.
En zelfs als ze wel worden vastgelegd, op de wiki, dan wordt er nog steeds

niet nageleefd

(het veranderende componentenverhaal is daar een voorbeeld van).

Advies:

Het opleggen van een strakke ontwikkelmethodiek lijkt me niet handig, omdat we ook flexibel moeten blijven,

Maar ik zou wel graag zien dat we wat basale technieken en methodieken zouden gaan gebruiken, die eigenlijk

voor elke IT-er gesneden koek zouden moeten zijn.

Antwoord Arno:

Een deel van de problemen die je noemt is terecht, maar zou eigenlijk niet mogen optreden als iedereen meer moeite doet te communiceren en overleggen. Hier spelen een aantal problemen:

- Bijna iedereen heeft zijn eigen taakgebied (specialisme), en heeft sterk de neiging

- dat tegen elke inmenging van buitenaf af te schermen. Daardoor is er wellicht te snel een reflex reactie: niet aankomen, dat is van mij!

- alles in het werk te stellen om zijn ding werkend te krijgen, ook als dat niet matcht met hoe anderen het eromheen designed hebben.

- individueel op te treden; dat zijn we immers gewend in ons eigen taakgebied.

- De OMG was een periodieke meeting en als zodanig niet altijd even zinvol; professioneel gezien moet het team zelf organiseren dat als het nodig is, de mensen die nodig zijn voor een probleem bij elkaar komen (in principe niet periodiek, maar wanneer het juiste moment daar is; het TEAM kan besluiten dat als men toch periodiek wil overleggen, dat natuurlijk mag!); die verantwoordelijkheid en bevoegdheid heeft ELK teamlid!

- Veel teamleden hebben een ingebouwde allergie tegen meetings; maar zolang een meeting een duidelijk doel heeft en dient om 1

- of meer teamleden verder te helpen bij het oplossen van een probleem, is het altijd zinvol.

- Wegens gebrekkig teamoverleg is er veel communicatie ''op de wandelgang'' en weinig borging van gemaakte afspraken.

Wellicht is er te weinig focus op het goed oplossen van 1 probleem, maar proberen we altijd 3-4 problemen tegelijk op te lossen en lukt dat vaak maar ten dele?

Wat kan je hier nog aan toevoegen?

From:

<https://www.astron.nl/lofarwiki/> - **LOFAR Wiki**

Permanent link:

https://www.astron.nl/lofarwiki/doku.php?id=public:astron_pwt_agile_workshop&rev=1338295424

Last update: **2012-05-29 12:43**

