

Software engineering 2013 - Evaluatie

Dexter Drupsteen
5887224

3 februari, 2013

1 Evaluatie

Evaluatie van de prestaties van ons team.

1.1 Prestaties

De grootste prestatie van ons team is samenwerking. We hebben op een open en eerlijke manier samengewerkt.

Naast de samenwerking, ben ik persoonlijk trots op de motivatie binnen het team. Het team was zeer gemotiveerd om met elkaar een mooi product neer te zetten, besteedde veel tijd aan het project en nam zijn verantwoordelijkheid voor de door het team (of teamlid) gemaakte deel van het project.

Tenslotte vind ik het resultaat wat we als team hebben neergezet erg goed. De code was wellicht niet zo netjes als had gekund, maar we hebben veel voor elkaar weten te boxen, veel features gemaakt en andere teams geholpen met integratie in ons systeem/framework.

1.2 Problemen

Als team hebben we niet veel problemen gehad, maar er is natuurlijk altijd ruimte voor verbetering. Hier de top 3 problemen die we zijn tegengekomen.

Het grootste probleem was de communicatie met andere teams. Sommige teams hebben niet aan verwachtingen kunnen voldoen, en door gebrek van communicatie (vanuit dat team, maar wellicht ook vragend vanuit ons team) werd pas op latere momenten duidelijk dat sommige delen van het project zwaar achter liepen.

Ook een probleem was een afwezige product owner zo een drie weken lang, en wat zieke dagen van andere teamleden. Verder, en dan doel ik zeker op mijzelf, structureel later aanwezig.

Het laatste probleem, wat snel verholpen was maar in het begin wat afremde, was de onbekendheid van bepaalde mensen met Git. Mergen ging in het begin moeizaam en personen hadden problemen met het opzetten van de git, het pushen en pullen. Verder was onduidelijk in wat voor structuur we git gingen gebruiken (ieder team zijn eigen repo of allemaal zelfde repo en per team een eigen branch).

2 Scrum

In dit deel wordt besproken hoe naar mijn mening Scrum binnen ons team gebruikt werd.

2.1 Rol Scrum

Door de verschillende meetings van Scrum had ieder teamlid ten alle tijden door waar de anderen mee bezig waren. De daily standup heeft heel erg geholpen om elkaar te informeren. Ook kwamen tijdens de daily standup problemen naar boven die mensen hadden. Door deze problemen te bespreken met z'n allen, waren ze al snel opgelost. Het releasen, of iets productie klaar hebben, op een vast moment vond ik ook wat toevoegen. Daardoor werd zichtbaar wat ieder team had gedaan de hele week en konden we alles samenvoegen zonder dat alles teveel uit elkaar was gelopen (verschillende richtingen op was gegaan binnen het project). Naast de daily standup en sprint ritme vind ik niet dat de Scrum methode wat heeft toegevoegd, mede doordat Scrum pas na een tijdje werkt. Het inschatten van backlog items en wat een schatting precies inhoud wordt pas duidelijk als je een sprint of zes a zeven hebt gehad (volgens de meeste Scrum guru's). Het project is veel te kort om Scrum goed in praktijk te brengen.

2.2 Methode

Zoals in de vorige paragraaf besproken hebben we eigenlijk alleen de daily standup gehouden en ons gehouden aan productie klaar maken op vaste momenten. Ook hebben we sprint planning geprobeerd (inclusief poker planning etc) alleen het inschatten had vrij weinig zin in onze ogen gezien onze velocity totaal niet bekend was. Iedereen had het gevoel dat dit wel handig is, maar wanneer je scrum voor langere tijd doet.

2.3 Evaluatie Scrum

Ik doe al minstens een jaar aan scrum en ben redelijk sceptisch geweest tegenover scrum. Maar de presentaties over scrum en hoe het echt zou moeten heeft me wel weer gemotiveerd om het op m'n werk meer te gaan sturen naar een betere en correctere manier. Het is een goede methode om met een team dat een tijd samenwerkt, een goede planning te maken en beloftes kan maken die het na komt. Verder is het hele agile principe natuurlijk heel erg handig voor de klant en verlaagt de kosten voor software engineering.

Wat ik minder vind aan scrum is dat de product owner prioriteit bepaalt, terwijl developers soms hele belangrijke dingen hebben die eerst gedaan moeten worden. Het opruimen van legacy is bijvoorbeeld niet iets wat een product owner snel bovenaan een lijst zal zetten (want het levert niets op), maar is ontzettend belangrijk voor stabiliteit van het systeem in het algemeen.

2.4 Teamprestatie en Scrum

Ik denk dat een deel van het succes van ons team te danken is aan de daily standups en de druk van release momenten. Ook het identificeren van features en wat daarvoor moet worden gedaan (hoewel dat zonder scrum natuurlijk ook

gebeurd zou zijn) heeft de groep in het algemeen een overzicht gegeven van welk werk er allemaal verzet moest worden.

3 Samenwerking

Hier volgt een evaluatie van de samenwerking binnen het team.

3.1 Goede punten samenwerking

Binnen het team was een hele open communicatie, iedereen zei wat hij vond en discussie over hoe we problemen op moeten lossen gaf inzicht in de problemen en oplossingsruimte. Gezamenlijk kwamen we tot mooie oplossingen. Mensen die vast zaten vroegen om hulp en ieder lid van het team was bereid om te helpen. Teamleden namen tijd voor elkaar en tijd om uit te leggen hoe zij het zagen, en de oplossing die ze voorogen hadden. om kwaliteit van de code hoog proberen te houden.

3.2 Punten voor verbetering

Uiteindelijk hadden we als team meer binnen de groep naar voren moeten stappen om te zeggen wat wij er van vinden en andere teams op hun verantwoordelijkheden moeten wijzen. Dit is iets wat we als collectief team niet hebben gedaan. Qua samenwerking denk ik niet dat er nog veel aanmerkingen zijn.

3.3 Rol in prestatie

De goede samenwerking en de openheid in ons team (zoals eerder genoemd) hebben gezorgd dat we als team goed presteerde. Nogmaals, openheid en informele communicatie en samenwerking werkte erg goed in ons team. Iedereen kon zijn werk doen en informeerde zowel formeel (daily standup) als informeel (door te laten zien als iets af was) leden of het gehele team over de voortgang of over problemen. Het gevoel dat we het met z'n allen deden was in ons team heel sterk en iedereen wilde er voor gaan en was bereid voor elkaar te werken.

3.4 Colocation

Het feit dat de informele berichtgeving en communicatie een groot deel was van ons succes maakt colocatie praktisch onmisbaar. Op het moment dat we niet met elkaar in gesprek hadden kunnen gaan vanwege aparte locaties, hadden problemen langer blijven bestaan (vanwege missende inzichten van andere teamleden) en was er waarschijnlijk meer onduidelijkheid geweest over werkzaamheden en verwachtingen van andere team leden. dit zou natuurlijk opgelost kunnen worden door services als Skype, colocatie is echter (weet ik uit ervaring) een stuk effectiever.

4 Persoonlijkheid

4.1 Myers-Brigs

Het Myers-Brigs type dat het meest bij mij past is *INTJ*. De score van de test op blackboard was niet geheel eenduidig over het laatste persoonlijkheids aspect, echter na bestudeering van de types in de hoofdstukken van het boek, ligt mijn voorkeur toch meer richting de J dan de P.

4.2 Vergelijking met andere types

Het grootste verschil tussen types met Extroversion en Introversion is dat mensen met een neiging naar Introversion minder snel gemotiveerd zijn door de mening van anderen. Iemand met introversion moet intrinsiek gemotiveerd zijn terwijl iemand met extroversion gemotiveerd kan worden door andere mensen. Dat zie ik heel erg bij mijzelf terug. Als mensen mij proberen te motiveren voor iets (door bijvoorbeeld te zeggen dat ik ergens goed werk voor heb geleverd) waar ik het punt niet van inzie (of het er dus niet mee eens ben dat ik goed werk heb geleverd) dan is het moeilijk om mij te motiveren. Dit komt ook door de neiging naar Judgement ten opzichte van Perception. Een mening die ik eenmaal heb gevormd over iets of iemand is lastig te veranderen. Mensen zullen mij dan erg moeten verrassen voor ik mijn mening verander terwijl iemand met een neiging naar perception eerder flexibiliteit zullen tonen in zo een situatie. De laatste persoonlijkheids preferentie die te bespreken valt is die van Thinking tegenover Feeling. Wanneer ik een idee of een mening over iemand of iets heb is dat lastig te veranderen (zoals gezegd), echter ben ik zeker beïnvloedbaar door een logische redenering iets wat de neiging naar Thinking karakteriseert. Ook zie ik dat zelf terug in mijn houding ten opzichte van (praktisch alles) wat iemand zegt of heeft gedaan. In eerste instantie ben ik erg kritisch en zal degene dus met een goed argument moeten komen waarom dat dan zou zijn (of waarom het een goede actie is). Ook zag ik terug in het project dat ik erg neigde naar een long term view. Dat wil zeggen, dat ik erg bezig was met nadenken over hoe de community die het verder zou moeten ontwikkelen het makkelijk op zou kunnen pakken of makkelijk zou kunnen veranderen.

Nu alle aspecten zijn besproken wilde ik de vergelijking met een totaal tegenovergesteld type maken. Het tegenovergestelde van INTJ is ESFP. Waar iemand met type ESFP erg flexibel is, is het bij iemand met INTJ allang gepland. Veranderingen zijn daardoor moeilijk te omvatten voor iemand met INTJ. In het boek van Myer-Brigs staat dat mensen met ESFP graag troubleshooters zijn. Echter, zelf erg fan zijnde van troubleshooten, vind ik dat iemand met INTJ dit ook kan zijn. Iemand met INTJ combineert met troubleshooten zijn lange termijnsvisie (T) (lost iets goed op), logische redenering (T), concentratie (I), overzicht van de oplossingsmogelijkheden (N) en beslissings gerichtheid (J) om tot een goede oplossing te komen. Toch ben ik het met het boek eens (ook uit eigen ervaring) dat de mensen met ESFP een stuk liever met mensen omgaan dan mensen met een voorkeur tot INTJ.

4.3 Motivatie in Software engineering

Mijn motivatie in software engineering rust op de mening die ik heb over het project dat gemaakt moet worden. Ik moet achter het project zelf staan, weten dat het iets gaat toevoegen. Het werken in kleinere kwalitatieve teams, motiveert mij meer dan werken met veel mensen (waarvan de kwaliteit af te vragen is). Verder motiveert een project mij heel erg als het intellectueel uitdagend is. Wanneer de uitdaging ontbreekt zal mijn motivatie en dus productiviteit een stuk dalen. Ik ben niet iemand die productieband werk erg interessant of voldoende vind. Creativiteit en slimmigheden in het werk vind ik mooi om te zien.

Wat mij het meeste moeite kost in software engineering projecten is afhankelijkheid van teamleden (of collega's). Ik vind het niet prettig afhankelijk te zijn van mensen waarvan ik niet weet of ze dezelfde kant op willen als ik. Het werken in een richting die me niet aanstaat maar bijvoorbeeld door meerderheid in een groep is besloten is iets wat me heel erg kan demotiveren. Ook een van de dingen die me demotiveren is het constant in discussie gaan over de richting waarin wordt opgegaan. Wanneer er een richting is besloten en een groot deel van het werk is gedaan, vind ik het (tenzij er een goede reden voor is) zonde om van richting te veranderen. Als laatste, wat me demotiveert, wil ik aanhalen dat er weinig uit mijn handen komt wanneer ik werk voor iemand die ik niet mag of iemand die niet meewerkt.