

Prenova spletne strani v igralniški industriji na agilni način - ali ko se poslovni analitik sooči z agilnim projektom

(Primer iz prakse)

Povzetek:

Agilne metodologije so postale v software-ski industriji zelo priljubljene, čeprav primarno izhajajo iz proizvodnega procesa in ne iz razvojnega procesa produkta. Priročne so postale zaradi svoje poenostavitve razvojnih procesov, ki so postali izredno kompleksni in togi.

Želja po fleksibilnosti in takojšnji povratni zanki glede narejenega je prinesla dobršno mero pozitivnih izkušenj, a hkrati tudi negativnih, v kolikor se pristopov ne lotimo z vsaj malo zdrave pameti. Spodnji primer govori o takšnem poskusu, ki se je začel nekoliko neredno, ampak smo ga uspeli spraviti v uspešen delovni proces, ki je na koncu prinesel rezultate.

Ključne besede: agilno, scrum, kanban, kaos, red

Vloga: BA, AG, PM, BPM, razvojni inženir, produktni lastnik

Prvo soočenje

Prvo soočenje s projektom, ki se je izvajal na agilni način, je bilo polno pričakovanj in v znamenju izziva, kakšno vlogo zavzeti v ekipi, če si profesionalno in po duši poslovni analitik.

Projekt je bil zasnovan po scrum metodologiji in prva stvar soočenja s projektom je bila, kakšna bo moja vloga v njem. Praksa na teh manjših projektih z visoko usposobljenim kadrom je po navadi takšna, da se ne sprašuje kaj dosti, kdo si, kaj si, ampak so vsi zadovoljni, če pride nov par rok na pomoč. Sam nisem programer, tako da sem se moral »najti« drugje. Pač znani princip metanja v vodo in instantnega plavanja. Ekipa je štela do 10 ljudi.

Jira

Pa so me porinili pred Jiro – orodje, ki je relativno zelo priljubljeno pri agilno vodenih projektih. Jira je pokazala stvari, ki so mi bile znane iz definicije agilnih

metodologij, in sicer – šprinti, backlog, uporabniške zgodbe, vendar bolj, ko sem gledal, bolj je stanje odstopalo od tega, kar govori teorija. Šprinti se kopičijo, ker nekatere stvari ostajajo odprte ali so zelo obsežne ali pa so odvisne od ljudi, ki zdaj nimajo časa, itd.. Uporabniške zgodbe se ponekod ponavljajo, ponekod so uporabljene kot taski za razvoj in ne več kot uporabniške zgodbe, epske zgodbe so bile, vendar nepovezane med seboj in pomanjkljive, poslovna pravila oz. kriteriji za prevzem (acceptance criteria) so bili nepopolni ali preveč na splošno opisani, ni se videlo konstrukta rešitve (glave in repa), itd.

Nič takšnega, kar ne bi že videl na klasičnih projektih, torej - kaos. Prvi vtis – treba je narediti red. Ok, se pogovorim z ekipo – vsi bi radi red, seveda, ampak mi za vsako stvar argumentirajo, zakaj se ne da narediti/prestaviti, ali karkoli.

Ok. Spet na začetek. Kaj se sploh izvaja? Lahko dobim pregled nad tem? Dobil. Kako se določajo prioritete, kaj se izvaja? V dogovoru z naročnikom, oz. lastnikom produkta. Kdo pa je? Aha, več jih je? Ok. Kdo pa določa prioritete med njimi? Vsi. Lepo. Spet – zelo podobno klasiki, samo, da ima zdaj ta način dela ime, ki mu pristoji in nič ni narobe, ker je itak vse agilno.

Od kaosa do reda

Kot poslovni analitik se stvari lotevam sistematično in tako sem potem postopal tudi v tem primeru. Identificiral sem:

- zahteve, ki so zaradi pomanjkljivih specifikacij ali manjka resursov trenutno neizvedljive;
- zahteve, ki so izvedljive, imajo resurse in sogovornika na strani produktnega lastnika, ki ve, kako mora izgledati končna rešitev;
- zahteve, ki so izvedljive, imajo resurse in sogovornika na strani produktnega lastnika, vendar ta še nima predstave, kako bi izgledala končna rešitev, potrebujejo jo pa;
- velikost zahtev: posamezne manjše stvari, manjši sklopi, večji sklopi;
- relativna pomembnost zahtev – relativna zaradi več produktnih lastnikov in pričakovanih rokov;
- odvisnost medsebojnih komponent rešitve oz. v kakšnem vrstnem redu se lahko stvari sploh lotimo;



Po teoriji bi seveda večina teh stvari morala biti jasna znotraj posameznega šprinta in ne bi smelo več biti prostora za dvome ali pomanjkljive informacije, vendar to zaradi takšnih in drugačnih razlogov ni bilo mogoče.

Po nekaj opravljenih intervjujih – tako s produktnimi lastniki kot s člani ekipe in drugih ekip, ki so že delali podobne projekte, smo skupaj identificirali idealni vrstni red zahtev za razvoj in to je bila dejansko samo smernica s katero smo lahko identificirali pot proti cilju ter se potem produktnimi lastniki okoli nje tudi uskladili.

To seveda ne pomeni, da ni bilo naknadnih sprememb. Te so prihajale in ni jih bilo tako malo. Dogajale pa so se v agilnem duhu: po izvedbi in testu je produktni lastnik dobil nove ideje, kako zapeljati stvar in je dal nove predloge za isto rešitev. Seveda z visoko prioriteto in pogojem, da brez teh novih sprememb ne more v živo.

Poleg teh ad-hoc idej pa so se pojavile tudi zahteve po funkcionalnostih, ki še niso bile nikjer omenjene in zavedene, so pa prav tako bile pogoj za štart v živo.

Agilno znotraj agilnega

To je samo nekaj situacij, ki jih srečujemo redno na raznoraznih projektih, klasičnih ali agilnih, vendar ne glede na metodologijo, je še vedno potrebna velika mera prilagodljivosti in improvizacije, da se ta navidezna neurejenost lahko spravi vsaj v neko obvladljivo obliko. V nasprotnem primeru je odgovor na vprašanje – kje smo in kdaj gremo lahko v živo, lahko vedno samo skomig z rameni.

Bili smo agilni znotraj agilnega. Od scrum metodologije smo tako hitro poslovali in se prestavili v kanban način, kjer je edina smernica, da delo ne stoji in da nihče ne gleda v zrak oz. je brez dela.

Moja naloga je bila, da pravočasno zagotovim specifikacije uporabniških zgodb in prevzemnih kriterijev za posamezno funkcionalnost, preden je kdo od razvojnikov prost za delo. Fokus je bil na tem, da so bili vsi resursi zasedeni in tudi, da sem se z njimi pravočasno posvetoval glede dela naprej zaradi sorodnosti tematike posamezne nove funkcionalnosti ali spremembe funkcionalnosti njihovemu preteklemu delu – vse z namenom, da se posamezna funkcionalnost ali sprememba lahko dogradi čim hitreje.

Moje delo poslovnega analitika je prišlo do izraza pri kristaliziranju zahtev posameznih produktnih lastnikov, kjer le-te še niso bile povsem jasne, so bile pomanjkljive ali pa jih je bilo potrebno spraviti v kontekst celotne rešitve (npr. to smo že rešili drugje s tem in tem – ali res potrebujete to rešitev ločeno od že obstoječe?).

To je bilo klasično delo poslovnega analitika. Druga polovica dela pa je bila delegacija dela – se pravi klasično projektno vodenje.