

Tehtävä 11

Referaatti Kalle Ilveksen kandidaatintutkielmasta Scrumban-menetelmän käyttö ketterässä ohjelmistokehityksessä

Ilves aloittaa tutkielmansa taustoittamalla lyhyesti ketterien ohjelmistokehitysmenetelmien syntyä perinteisen vesiputousmallin kohtaamaan kritiikin kautta. Ketterät menetelmät painottivat joustavuutta, valmiutta jatkuvaan muutokseen ja yksilöiden merkitystä jäykän hierarkkisen ja lineaarisen yksisuuntaisen mallin sijaan. Ilves esittelee Scrumin peruseriaatteet, joihin kuuluu ohjelmiston kehitys lyhyissä sprinteissä inkrementaalisesti eli osa kerrallaan toimivaksi saattaen. Työ tehdään kehittäjätiimin, Scrum masterin ja Product ownerin yhteistyönä ja Scrumin määrittämää sprintteihin liittyvää tapaamisten sykliä noudattaen. Toisena esimerkkinä ketteristä menetelmistä esitellään Toyotan alunperin kehittämien tuotantomenetelmien pohjalta syntynyt Kanban. Kanbanissa omaperäistä on työn etenemisen visualisointi kanban-aulun avulla, mihin liittyy eri kehitysvaiheissa oleville tehtäville asetetut määrälliset rajoitukset. Periaatteina on lean-ajattelun hengessä työn laadun ja tehokkuuden maksimointi ja samalla turhan työn ja ”jätteen” minimointi.

Scrumin ja Kanbanin peruseriaatteissa on paljon yhteistä, ja niiden eri elementtien yhdistämisen pohjalta on syntynyt Scrumbaniksi kutsuttu ohjelmistokehityksen työkalu. Scrumbanissa käytetään kanban-aulua työmäärän kontrollin keskeisenä välineenä, ja määrätyn kestoisten iteraatioiden kaltaiset Scrumin asettamat rajoitteet korvataan joustavammilla, jatkuvaan suunnitteluun pohjaavilla käytänteillä. Ilves esittelee tutkielmansa lopuksi kaksi esimerkkitapausta, joissa arvioidaan kehittäjätiimien Scrumista Scrumbanin käyttöön siirtymiseen liittyneitä etuja ja haasteita. Tapauksissa Scrumin ja Kanbanin piirteitä yhdisteltiin hieman vaihtelevasti, mutta molemmissa luovuttiin aikarajoitteisista iteraatioista ja otettiin käyttöön kanban-aulu tehtävien visualisoinnin ja samanaikaisen työn rajoittamisen välineeksi. Tavoiteltua balanssia joustavuuden ja rajoitteiden välillä ei esimerkkien perusteella ole aina helppo saavuttaa, eikä Scrumbankaan tarjoa siihen mitään ihmeratkaisua. Kuten Ilves toteaa, sopiva kombinaatio on muodostettava yrittämisen ja erehdyksen kautta jokaisessa projektissa ja tiimissä erikseen.