

TURUN YLIOPISTO  
Informaatioteknologian laitos

VIHRIÄ, TOMI: Ketterä ohjelmistonkehitys Scrum-menetelmää käyttäen –  
kohti todellista ketteryyttä: tutkimustapaus osittain ketteristä  
projekteista

Diplomityö, 99 s., 2 liites.  
Ohjelmistotekniikka  
Toukokuu 2010

---

Nykyaikaisen ohjelmistonkehityksen on pystyttävä sopeutumaan jatkuvaan muutokseen. Ketterät ohjelmistonkehitysmenetelmät vastaavat tähän haasteeseen edistämällä yksinkertaisia, keveitä, kommunikointia ja itseohjautuvia tiimejä suosivia sekä ohjelmakoodin tärkeyttä painottavia toimintatapoja ja periaatteita. Ketterässä Scrum-menetelmässä monimutkaisuutta hallitaan näkyvyyden, katselmointien ja toiminnan sopeuttamisen avulla.

Tässä tutkielmassa selvitetään ensin teoreettisesti, miten ketterät menetelmät, ja erityisesti Scrum-menetelmä, pystyvät vastaamaan nykyaikaisen ohjelmistonkehityksen haasteisiin. Teoreettisen osan jälkeen esitellään kaksi Scrum-menetelmän sovellusta, joita analysoidaan ketterien sekä Scrum-menetelmän periaatteiden ja käytäntöjen avulla. Sovelluksista yritetään tunnistaa ei-ketteriä piirteitä, analysoida minkä takia niitä löytyy, ja onko niistä haittaa sovellukselle. Lisäksi pohditaan, miten sovelluksista saisi puhtaasti ketteriä, ja mitä hyötyä tehtävistä muutoksista saataisiin.

Tutkielmassa havaittiin ketterien ja Scrum-menetelmän muodostavan päteville ihmisille toimintaympäristön, jossa voidaan kehittää iteratiivisesti ja inkrementaalisesti käytännöllisiä ratkaisuja kohdattuihin ongelmiin. Toiminta tässä ympäristössä tapahtuu itseorganisoituvasti ja tiiviissä yhteistyössä asiakkaiden kanssa.

Ensimmäisessä sovelluksessa ei-ketterät piirteet ovat vielä hallitsevassa osassa sen pitkän perinteisten menetelmien taustan takia. Ei-ketterät piirteet vaikeuttavat asiakkaiden vaatimukset täyttävän kasvavan ohjelmiston kehittämistä. Toinen sovellus noudattaa olennaisimpia ketteriä toimintatapoja. Jotta molemmat sovellukset saataisiin ketterimmiksi, niin Scrum-menetelmän avainrooleissa toimivien henkilöiden on omaksuttava ketterät periaatteet ja otettava ne käyttöön päivittäisessä työssään. Lisäksi organisaation muiden osien on oltava aktiivisia ja tuettava ketterien periaatteiden noudattamista. Tällöin kehitettävät ohjelmistot täyttävät asiakkaiden vaatimukset paremmin sekä projektia ja prosessia pystytään hallitsemaan empiirisesti.

Avainsanat: ketterä ohjelmistonkehitys, Scrum-menetelmä, empiirinen projektinhallinta

UNIVERSITY OF TURKU  
Department of Information Technology

VIHRIÄ, TOMI: Agile Software Development with Scrum – Towards True  
Agility: Case About Partly Agile Projects

Master of Science in Technology Thesis, 99 p., 2 app. p.  
Software Engineering  
May 2010

---

Modern software development has to be able to adapt to constant change. Agile software development responds to this challenge by promoting practices and principles that reflect an agile sensibility of simplicity, lightness, communication, self-directed teams and importance of working software. Scrum is an agile method that manages complexity with the help of visibility, inspection and adaptation.

The purpose of this thesis is first to clarify theoretically how agile methods, and especially Scrum method, can respond to the challenges of the modern software development. After that, two Scrum projects are demonstrated and analyzed with means provided by principles and practices of the both agile and Scrum methods. The purpose of the analysis is to recognize non-agile features from the projects, analyze why they exist and do they cause any harm to the project. It is also considered how projects could become truly agile and what would be the advantage gained from the changes.

It was discovered in this thesis that agile and Scrum methods provide an environment for the capable people, where practical solutions can be developed in an iterative and incremental way to the encountered problems. Working in the environment happens in self-organizing way and in tight collaboration with the customers.

The first project still has non-agile features as dominant part of it, because of its long roots in the traditional software development. Non-agile features complicate development of the incremental software that fulfills customer's requirements. The second project follows most of the fundamental agile practices. In order to make both projects more agile, the people acting in the key roles of the Scrum method must adapt agile principles and take them into use in their daily tasks. In addition to that the other parts of the organization have to be active and to support compliance of the agile principles. Then developed software will more specifically fulfill customer's requirements and both project and process can be managed empirically.

Keywords: Agile Software Development, Scrum, Empirical Project Management