

Petri Marttinen

Scrum ja .Net ohjelmistokehityksessä

Tietojärjestelmätieteen
kandidaatintutkielma
13.5.2009

Jyväskylän yliopisto
Tietojenkäsittelytieteiden laitos
Jyväskylä

SISÄLLYSLUETTELO

TIIVISTELMÄ	3
1 JOHDANTO	5
2 SCRUM.....	6
2.1 Scrum -prosessi	7
2.1.1 Scrum Team.....	7
2.1.2 Product Backlog.....	8
2.1.3 Sprint.	9
2.1.4 Sprint Backlog	9
2.1.5 Sprint Review ja Sprint Retrospective	10
3 MICROSOFT .NET	12
3.1 Visual Studio Team System.....	12
3.2 Microsoft Solutions Framework	13
4 VISUAL STUDIO JA SCRUM.....	14
4.1 Visual Studio Team Foundation.....	14
4.2 Visual Studion Team System ja projektinhallinta	14
4.2.1 Scrum For Team System.....	15
4.2.2 Visual Studio Team System ja Scrum	17
5 YHTEENVETO	18
LÄHTEITÄ	20

TIIVISTELMÄ

Marttinen, Petri Juhani

Tietojärjestelmätieteen kandidaatintutkielma / Petri Marttinen

Jyväskylä: Jyväskylän yliopisto, 2009.

21 s.

Kandidaatintutkielma

Ohjelmistokehitys on monimutkainen liiketoiminta-alue. Nykyään ohjelmistot mallintavat aina vain tarkemmin todellista ympäristöämme, joten monimutkaisuus ei ole enää yllätys kenellekään. Scrum-menetelmä on kehitetty tuottamaan käytettävyydeltään ja laadultaan hyviä ohjelmistoja monimutkaisiin tarpeisiin ja ympäristöihin. (Schwaber K., 2004) Scrum on suoraviivainen ketterä menetelmä, joka painottaa mm. itseorganisoituvien projektiryhmien ja prosessien yksinkertaisuuden merkitystä.

Microsoftin kehittämä .Net Framework ja sen mukanaan tuomat ohjelmistokehitystyökalut ja ohjelmistokehitysprosessia tukevat välineet tarjoavat työkalukokonaisuuden nykyaikaiseen ohjelmistotuotantoon. Tämän kehyksen ohjelmistokehityksen kannalta tärkein komponentti on 'Visual Studio Team System'-tuoteperhe (VSTS). Olennainen osa VSTS:ä on Visual Studio Team Foundation (VSTF). VSTF on palvelinohjelmisto, jonka osia ovat erilaiset prosessinhallintaan ja raportointiin kehitetyt työkalut. VSTF tukee Microsoft Solutions Foundationia (MSF), joka on monipuolinen ja dynaaminen perusta erilaisten prosessien soveltamiseen ohjelmistoprojekteissa. MSF tarjoaa tuen Scrum-mallin mukaiselle ohjelmistokehitykselle mm. erilaisten prosessimallipohjien avulla.

Mitä tämä VSTS:n tarjoama tuki Scrumille tarkoittaa käytännössä? Mitä ominaisuuksia tarjotaan Scrumin mukaiseen ohjelmistokehitykseen? Voiko

ideologialtaan prosesseja yksinkertaistavia Scrum-projekteja edes tukea prosesseja tukemalla?

Tässä tutkimuksessa käymme läpi tärkeimmät Scrum-menetelmän osa-alueet ja Visual Studio Team Systemin oleelliset osat Scrumin kannalta. Tämän jälkeen tarkastelemme kuinka Scrum-menetelmää käytännössä tuetaan 'Visual Studio Team System'-kokonaisuudessa ja ulkopuolisten sovelluslisäosien puolesta. Tutkimus on toteutettu kirjallisuuskatsauksena ja tutkimalla käytännössä ohjelmistojen tarjoamia ominaisuuksia.

Tutkimuksen tuloksina tiedostettiin Scrumin olevan menetelmänä suoraviivainen ja yksinkertainen, jolloin raskaita prosessinhallinta- ja raportointityökaluja ei välttämättä tarvita. VSTS:n tarjoamat työkalut nähdään kuitenkin hyödyllisiksi, vaikkeivat ne suoranaisesti tuekaan Scrumin ideologian mukaista ohjelmistokehitystä.

Aiheeseen liittyvä sanasto on suurelta osin englanninkielistä ja näitä englanninkielen termejä käytetään yleisesti myös suomalaisessa yritysmaailmassa yrittämättä suomentaa niitä. Tässä tutkimuksessakaan en suomenna esim. Scrum-menetelmän termejä.

AVAINSANAT: Scrum, .Net, Visual Studio Team System

1 JOHDANTO

Ketterät ohjelmistokehityksen menetelmän ovat saaneet jo tukevan jalansijan ohjelmistokehityksiprojekteissa. Scrum on tällä hetkellä eräs tunnetuimmista ketteristä ohjelmistokehityksen menetelmistä. Scrum painottaa itseohjaituvia projektiryhmiä ja nopeaa suoraviivaista ohjelmistonkehitystä.

Microsoftin kehittämä .Net Framework on Javan rinnalla tunnetuimpia ohjelmistokehitysalustoja. .Net Framework käsittää ohjelmistokomponenttikirjaston sekä ajoympäristön ja sen käytetyin ohjelmistokehitystyökalu on Visual Studio. Visual Studio Team System on kokonaisuus, joka tarjoaa ohjelmistokehitykseen työkaluja ja tukee myös ketteriä menetelmiä ja CMMI:tä.

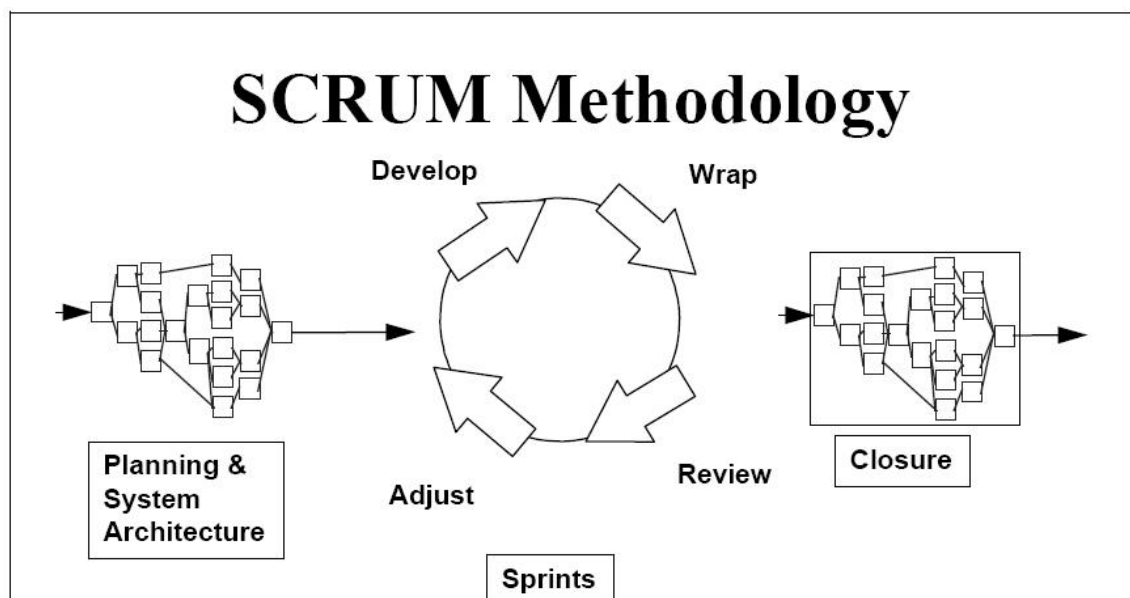
Kuinka Scrum-menetelmän mukainen ohjelmistokehitys soveltuu .Net framework -pohjaisiin ohjelmistoprojekteihin? Visual Studio Team System tarjoaa jo useita projektinhallintaan liittyviä työkaluja ja sen kehittäjä Microsoft mainostaa sen tukevia ketteriä menetelmiä. Mitä nämä ominaisuudet ovat ja tukevatko nämä ominaisuudet tyypillistä Scrum projektia? Mitä lisätyökaluja .Net -ohjelmistokehitykseen Scrumia käyttäen voisi tarvita? Voiko Scrum-projektia ylipäänsä tukea prosessinhallintatyökaluilla?

Tässä dokumentissa selvitetään Scrumin tärkeimmät käsitteet, käytänteet ja periaatteet. Tämän jälkeen tarkastellaan Microsoft .Net Frameworkin mukaista ohjelmistonkehitystä ja sen tarjoamia teknologioita ohjelmistokehityksen tukemiseen. Lopuksi selvitetään löytyykö Microsoftin tarjoamasta kehitysympäristöstä (Visual Studio Team System) työkaluja Scrumin mukaiseen ohjelmistonkehitykseen.

2 SCRUM

Scrum on ketterä ohjelmistokehitysmenetelmä. Kirjassaan *The New New Product Development Game* Ikujiro Nonaka sekä Hirotaka Takeuchi kertovat tuottavuudeltaan korkeasta järjestelmänkehityksestä. Kertoessaan menetelmästä he viittaavat Scrumilla rugby-peliin. Scrum –nimitystä käytettiin lopulta kuvaamaan koko menetelmää pelin ja menetelmän yhtäläisyyksien takia. Näitä yhtäläisyyksiä ovat mm. reagoivuus, nopeus, itseorganisoituvuus ja jatkuvuus. (Takeuchi H., Nonaka I., 1986)

Kuvassa 1 esitellään korkealla tasolla Scrumin vaiheet. Scrum perustuu ajatukseen, jonka mukaan projektin ensimmäiset ja viimeiset vaiheet ovat yleensä hyvin määriteltä (Kuvassa 1 vaiheet 'Planning & System Architecture' ja Closure). Ero esim. vesiputousmalliin ja iteratiiviseen spiraalimalliin, on tavassa tarkastella tätä ns. Sprint–vaihetta joustavana, epälineaarisenä ja usein hallitsemattomana kokonaisuutena. (Schwaber K., SCRUM Development Process)



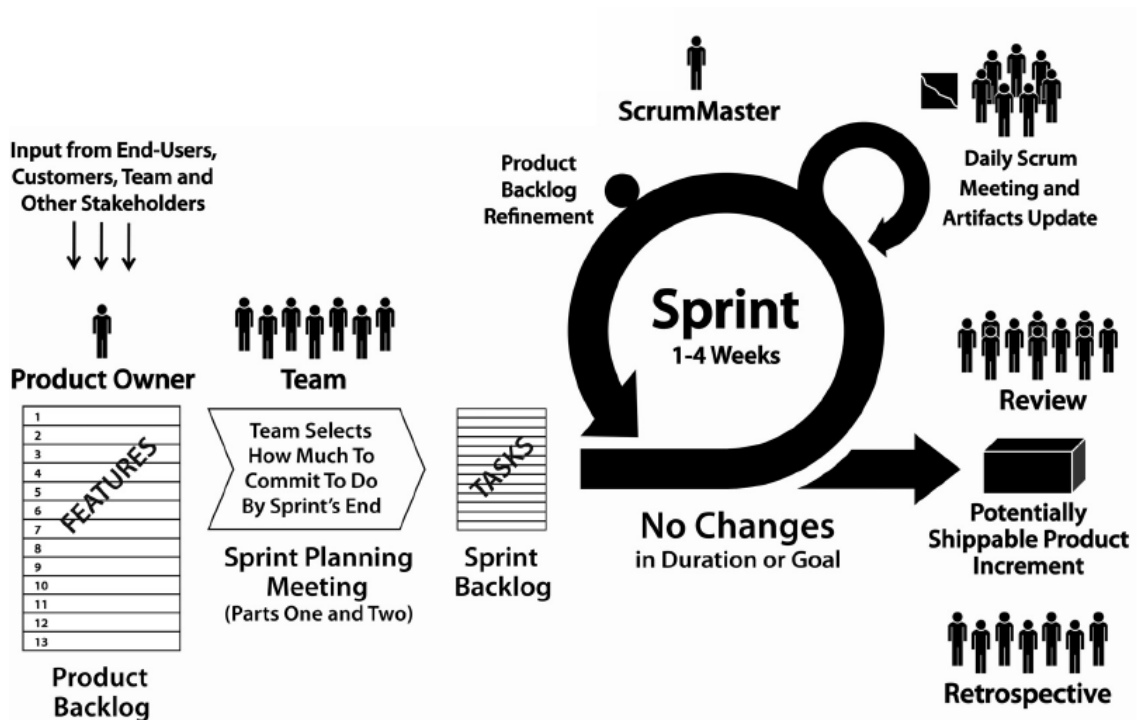
Kuva 1 Scrum Menetelmä (Schwaber K., SCRUM Development Process)

Seuraavaksi esitellään Scrum–menetelmän pääkohdat tarkemmin.

2.1 Scrum -prosessi

Tässä kappaleessa käydään läpi Scrum -prosessin vaiheet keskeisten käsitteiden avulla. Tämän työn kannalta tärkeimmät käsitteet ovat: Product Backlog, Sprint ja Sprint Backlog.

Kuvassa 2 esitellään Scrum-Prosessin tärkeimmät osat Deemerin ym mukaan. Seuraavissa kappaleissa käydään läpi jokainen kohta prosessista tarkemmin.



Kuva 2 Scrum prosessi

2.1.1 Scrum Team

Scrum Team on pieni (yleensä 5-10 jäsentä) itseorganisoituva ryhmä. Ryhmää ohjaa Scrum Master, jonka tehtävänä on mm. pitää huolta ryhmän työskentelyn sujuvuudesta ja resursseista, sekä siitä ettei ryhmän työskentelyä häiritä ulkopuolisen tahon toimesta. (Schwaber K., Beedle M., 2004)

2.1.2 Product Backlog

Ideaalitapauksessa Scrum–menetelmää tulisi alkaa soveltaa jo projektin käynnistymisvaiheessa. Menetelmän mukaan, aluksi luetellaan kaikki ominaisuudet ja toiminnot, jotka rakennettavaan järjestelmään liittyvät. Luetellut asiat kootaan listaan, jota kutsutaan Product Backlogiksi. Listaa päivitetään usein ja vaatimukset listalle voivat tulla useasta lähteestä. Vaatimuksille annetaan prioriteetit Product Ownerin (yleensä käytännössä asiakkaan edustaja) toimesta. Kukaan muu kuin Product Owner ei saa tehdä priorisointiin muutoksia. Käytännössä Product Owner valitsee järjestelmään toteutettavat ominaisuudet. (Schwaber K., Beedle M., 2004)

Kuvassa 3 näkyy Deemerin ym. antama esimerkki Product Backlogista. Product Backlogin rivit kuvastavat rakennettavan järjestelmän ominaisuuksia, korjattavia vikoja tai jopa parannuksia kehitysprosessiin. Jokainen näistä työtehtävistä voidaan arvottaa Product Ownerin liiketoiminnan kannalta ja priorisoida. Lisäksi voidaan arvioida tehtäviin liittyvät työmäärät. Tehtävien työmäärät mitataan tässä vaiheessa usein suhteessa toisiin tehtäviin. Jokaisen Sprintin jälkeen arvioidaan jäljellä olevaan työmäärää.

Item	Details (wiki URL)	Priority	Estimate of Value	Initial Estimate of Effort	New Estimates of Effort Remaining as of Sprint...					
					1	2	3	4	5	6
As a buyer, I want to place a book in a shopping cart (see UI sketches on wiki page)	...	1	7	5						
As a buyer, I want to remove a book in a shopping cart	...	2	6	2						
Improve transaction processing performance (see target performance metrics on wiki)	...	3	6	13						
Investigate solutions for speeding up credit card validation (see target performance metrics on wiki)	...	4	6	20						
Upgrade all servers to Apache 2.2.3	...	5	5	13						
Diagnose and fix the order processing script errors (bugzilla ID 14823)	...	6	2	3						
As a shopper, I want to create and save a wish list	...	7	7	40						
As a shopper, I want to to add or delete items on my wish list	...	8	4	20						

Kuva 3 Esimerkki Product Backlogista

2.1.3 Sprint

Sprint on 1-4 viikon mittainen toistuva ajanjakso, jonka aikana ryhmän tavoitteena on toteuttaa toimiva järjestelmän osa (prototyyppejä). Ennen jokaista Sprintiä ryhmä kokoontuu keskustelemaan Product Ownerin kanssa Product Backlogin sisältämistä vaatimuksista. Tämän jälkeen ryhmän jäsenet päättävät keskenään mitä ominaisuuksia Product Backlogista aiotaan toteuttaa seuraavan Sprintin aikana, jotta Sprintin tuloksena olisi valmis järjestelmän inkrementti. Näitä peräkkäisiä kokoontumisia kutsutaan nimellä Sprint Planning Meeting. (Schwaber K., Beedle M., 2004)

On hyvä huomata, ettei Scrum-prosessin mukaisten ensimmäisten Sprintien aikana ole tarkoitus keskittyä järjestelmän arkkitehtuuriin ja loppuunvietyyn suunnitteluun, kuten esim. vesiputousmallin mukaisessa toteutustavassa voisi olla tarkoituksenmukaista. Sen sijaan arkkitehtuurin odotetaan muotoutuvan itsestään Scrum-prosessin mukaisten Sprintien tuloksena. (Schwaber K., Beedle M., 2004)

Sprintin aikana Scrum Team kokoontuu myös päivittäisiin palavereihin (Daily Scrum), jolloin ryhmän jäsenet vastaavat kolmeen kysymykseen: Mitä teit edellisen Daily Scrumin jälkeen? Mitä aiot tehdä ennen seuraavaa Daily Scrumia? Onko tielläsi esteitä, jotka hankaloittavat työtäsi? (Scrum.dk)

2.1.4 Sprint Backlog

Ennen jokaista Sprintiä pidettävän Sprint Planning Meetingin tuloksena syntyy Sprint Backlog, joka pitää sisällään kyseiseen Sprintiin valitut toteutettavat ominaisuudet. Ryhmä valitsee tehtävät Sprint Backlogiin Product Backlogista niiden prioriteettien perusteella, ottaen huomioon, että tavoitteena on saada aikaan toimiva järjestelmän inkrementti (Schwaber K., Beedle M., 2004). Tehtäväkokonaisuuksia tarkennetaan ja ne pilkotaan pienempiin tehtäviin, jotta

ryhmä voi helpommin suunnitella kuinka paljon kukin osatehtävä vie resursseja.

Scrum on periaatteessa skaalautuva menetelmä. Usein Scrumin skaalaamista ei kuitenkaan suositella. Useat Scrum Teamit voivat toteuttaa samaa suurempaa järjestelmäkokonaisuutta, valiten kukin omat erilliset ominaisuudet omiin Sprint Backlogeihin. Ryhmät ovat itseohjautuvia, joten ryhmät päättävät autonomisesti, kuinka aikovat saavuttaa Sprintin tavoitteet. (Schwaber K., Beedle M., 2004)

Kuva 4 antaa esimerkin siitä, millaiselta Sprint Backlog voisi näyttää (Deemer ym.).

Product Backlog Item	Sprint Task	Volunteer	Initial Estimate of Effort	New Estimates of Effort Remaining at end of Day...					
				1	2	3	4	5	6
As a buyer, I want to place a book in a shopping cart	modify database	Sanjay	5	4	3	0	0	0	
	create webpage (UI)	Jing	3	3	3	2	0	0	
	create webpage (Javascript logic)	Tracy & Sam	2	2	2	2	1	0	
	write automated acceptance tests	Sarah	5	5	5	5	5	0	
	update buyer help webpage	Sanjay & Jing	3	3	3	3	3	0	
Improve transaction processing performance	merge DCP code and complete layer-level tests		5	5	5	5	5	5	
	complete machine order for pRank		3	3	8	8	8	8	
	change DCP and reader to use pRank http API		5	5	5	5	5	5	
						
		Total (person hours)	50	49	48	44	43	34	

Kuva 4 Esimerkki Sprint Backlogista

2.1.5 Sprint Review ja Sprint Retrospective

Sprint Review on jokaisen Sprintin päätteeksi pidettävä noin neljän tunnin mittainen palaveri, jossa Scrum Team esittelee Sprintin aikana valmistuneen tuotoksensa Product Ownerille ja sidosryhmille. Sprint Retrospective-palaveri pidetään Sprint Review'n jälkeen. Tässä palaverissa tarkastellaan mennyttä Sprintiä ja reflektoidaan Sprintin aikana opittuja asioita. Sprint Retrospective keskittyy siis prosessin tarkasteluun, siinä missä Sprint Review tarkasteli

tuotosta. Näiden palaverien jälkeen osapuolet valmistautuvat uuteen Sprintiin. (Schwaber K., Beedle M., 2004)

3 MICROSOFT .NET

Microsoftin kehittämä .Net framework käsittää ohjelmistokomponenttikirjaston sekä ajoympäristön, jossa .Net ohjelmistoja ajetaan. Kehitysympäristönä käytetään useimmiten Microsoftin kehittämää Visual Studiota. Framework tukee n. 20 ohjelmointikieltä. Useimmiten käytetään kuitenkin ainoastaan C# tai Visual Basic .Net -kieliä. (.NET Framework Conceptual Overview)

Seuraavaksi esitellään tämän työn kannalta oleelliset osat Microsoftin .Net Frameworkistä.

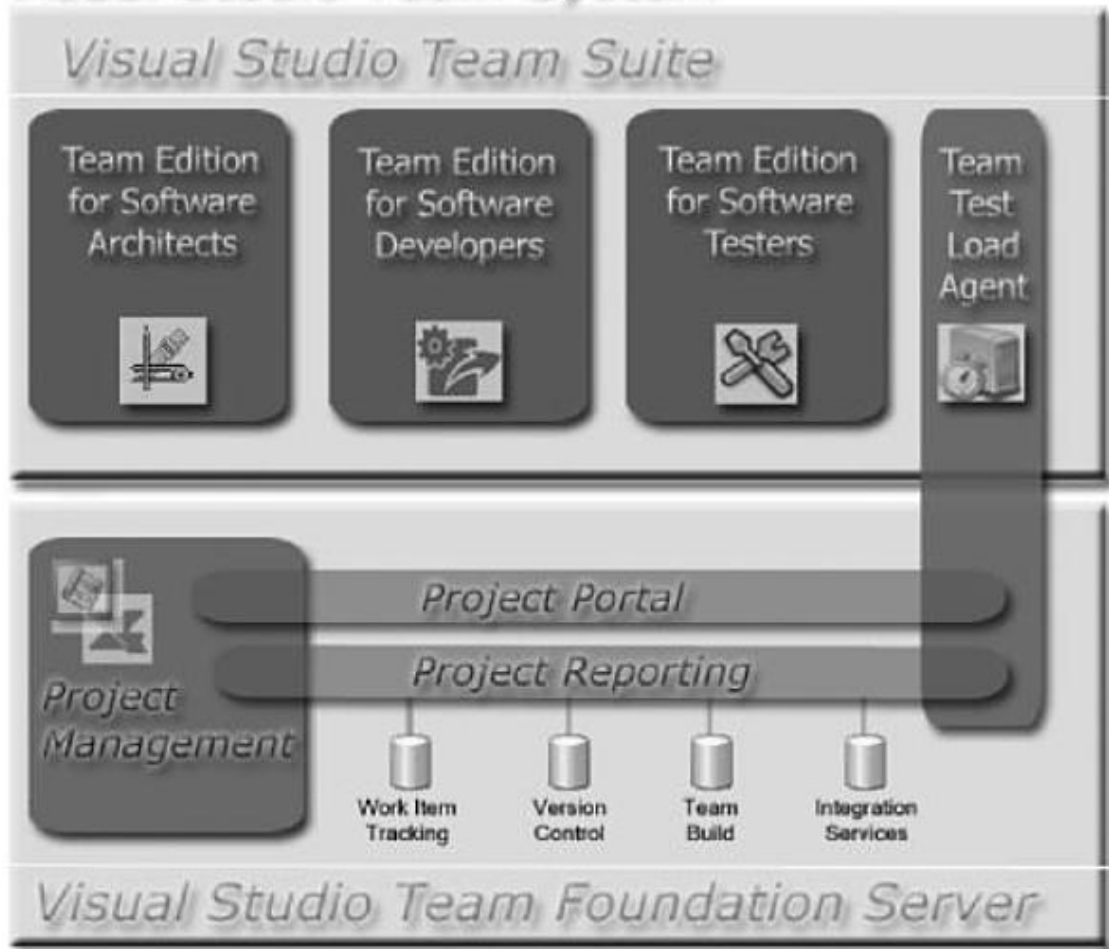
3.1 Visual Studio Team System

Visual Studio on graafinen kehitysympäristö, jonka avulla .Net -ohjelmistoja kehitetään. Visual Studio Team System (VSTS) on kokoelma työkaluja, joiden avulla ohjelmistojen kehittäjät, kehittäjäryhmät ja muut ohjelmistonkehitykseen liittyvät ryhmät (stakeholders) voivat kommunikoida keskenään reaaliaikaisesti. (Nelson D., 2006)

Visual Studio Team System koostuu client-ohjelmistoista (Visual Studio Team Suite) ja palvelinohjelmistoista (Visual Studio Team Foundation). Kaiken pohjalla toimii Team Foundation Server. VSTS on suunniteltu tukemaan erilaisia ohjelmistokehitysmenetelmiä, kuten Microsoft Solutions Framework (MSF), Rational Unified Process (RUP), Extreme Programming (XP) ja Scrum. (Nelson D., 2006)

Kuvassa 6 näkyy Visual Studio Team Systemin osat korkealla tasolla. (Nelson D., 2006)

Visual Studio Team System



Kuva 5 Visual Studio Team System ja sen osat

3.2 Microsoft Solutions Framework

Microsoft Solutions Framework (MSF) tarjoaa skaalautuvan työkalun erilaisten ohjelmistokehitysprosessien tukemiseen. MSF tukee sekä ketteriä menetelmiä, että formaaleja lähestymistapoja. MSF:n periaatteen mukaan ei ole olemassa yleiskäyttöisiä prosesseja tai menetelmiä, jotka olisivat optimaalisia jokaiseen ohjelmistokehitysprojektiin. MSF määrittelee mallien (template) avulla, kuinka mikäkin yksittäinen projekti käyttäytyy prosessiensa osalta. (Bodea C-N., 2005)

4 VISUAL STUDIO JA SCRUM

4.1 Visual Studio Team Foundation

Visual Studio Team Foundation on palvelinmella ajettava Visual Studio Team Systemin komponentti. Team Foundation yhdistää kaikki muut Team Systemin komponentit yhdeksi kokonaisuudeksi, jonka avulla ohjelmistokehitysorganisaatiot voivat organisoida prosesseja kehittäjien ja muiden organisaatioon kuuluvien välillä. (Nelson D., 2006)

Visual Studio Team Foundation tukee Microsoft Solutions Foundationia, joka on CMMI:n tasolla 3. (Nelson D., 2006)

4.2 Visual Studion Team System ja projektinhallinta

Ketterissä menetelmissä painotetaan usein ihmisten, kommunikoinnin ja muutoksen merkitystä jäykkien menetelmien ja prosessien sijaan. Prosessityön tukeminen on kuitenkin usein nähty hyödylliseksi. (Ketterät käytännöt.fi)

Visual Studio Team System tarjoaa mahdollisuuden käyttää erilaisia malleja projektinhallintaan. Näitä malleja voivat olla CMMI:n mukaiset mallit, kolmannen osapuolen tarjoamat mallit tai mallin voi kehittää itse. Scrum-malleja Team Foundationille ovat mm.

- eScrum: Web-pohjainen projektinhallintatyökalu Team Foundation Server Platformille. Tarjoaa tuen mm. product backlogin, sprint backlogien, tehtävien, retrospective, ja raporttien hallintaan. (eScrum for Microsoft Visual Studio Team System)
- Scrumptious: Scrumptious on projektimalli (template) Microsoft Solutions Foundationille. Se tarjoaa mallin kehittää ohjelmistoja Scrumin

mukaisesti, mutta keskittyy lähinnä ohjelmistonkehitykseen metodologian sijaan. (Scrumptious)

Tutkimistani työkaluista Visual Studio Team Foundationille Conchango Ltd:n kehittämä Scrum For Team System vaikuttaisi Kuitenkin kaikkein kattavimmin tukevan Scrumia. Käsittelemme tätä työkalua seuraavaksi tarkemmin.

4.2.1 Scrum For Team System

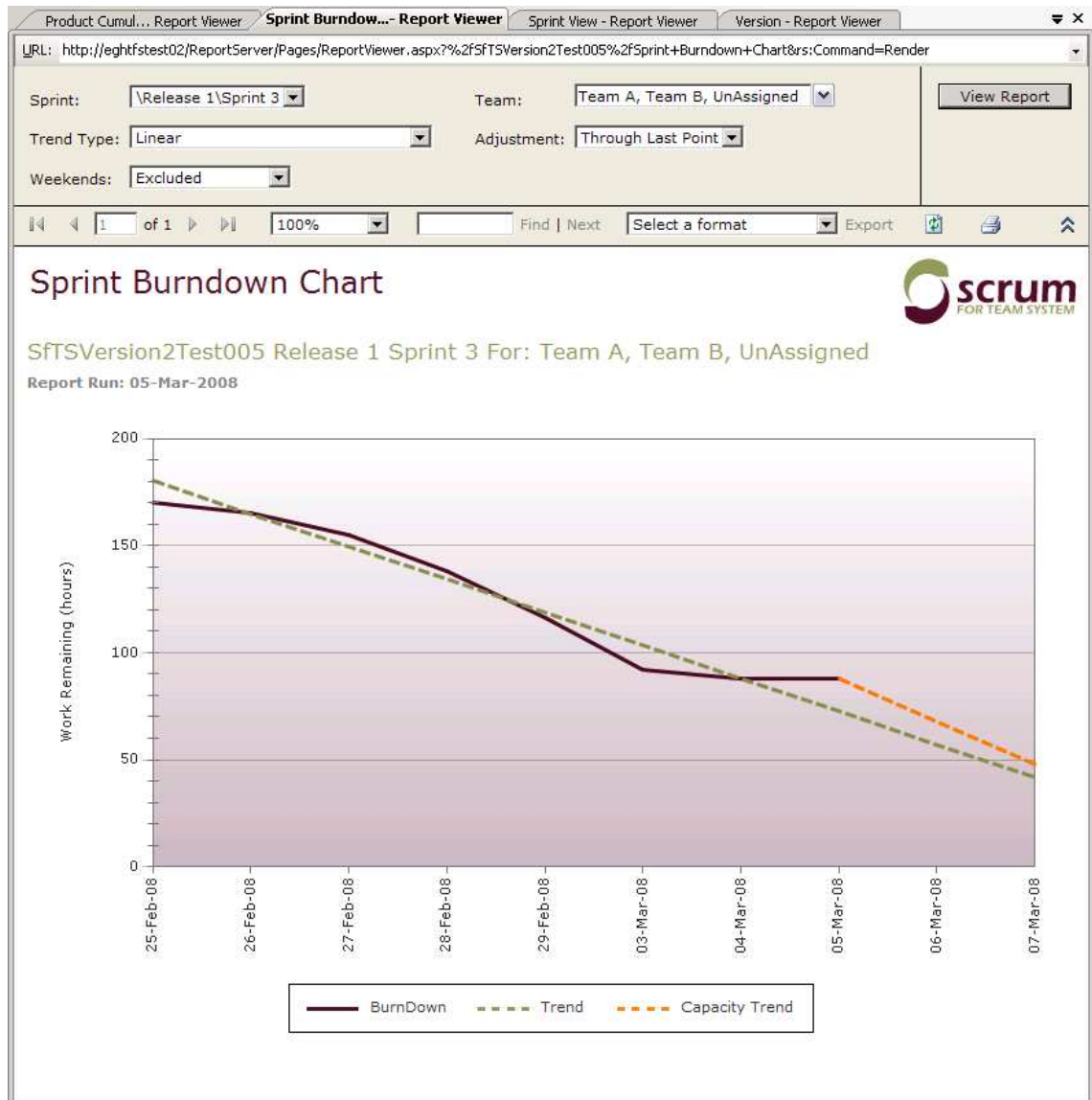
Heti nopean tutkiskelun jälkeen voisi huomata, että Scrum For Team System (SCTS) on tarjolla olevista Scrum-malleista Visual Studio Team Systemille se käytetyin, ja varmasti monelle se suoraviivaisin tapa.

Työkalun avulla voi suorittaa kaikki Scrum-menetelmää tukevat toimenpiteet, joihin Scrum Master tai projektin alullepanija voi törmätä: Backlogien luonti ja hallinta (Scrum artefacts), roolien hallinta ja ehkä tärkeimpänä itse prosessin hallinta. Työkalu tarjoaa myös graafiset näkymät raportteihin, kuten Sprint burndown chart, joka on arvio jäljellä olevasta työmäärästä.

Seuraavaksi käymme läpi tärkeimmät Scrumia tukevat ominaisuudet, joita Scrum For Team System tukee.

- **Product Backlog:** SCTS:n avulla voidaan luoda Product Backlog ja hallita sitä. Listalla olevat tehtäväkokonaisuudet voi priorisoida ja niille voi antaa työmääräarvot.
- **Sprint Backlog:** SCTS:n avulla voidaan luoda Sprint Backlog ja sen sisältämät tehtävät ja niitä voidaan hallinnoida. Scrum Teamin jäsenet voivat merkata tehtävät omille työlistoilleen ja merkata tehtäviä suoritetuiksi.

- Backlog Burndown: SCTS tarjoaa tavan seurata jäljellä olevia työmääriä Sprint Backlogeissa ja Product Backlogeissa Scrum-mallin mukaisesti. Kuvassa 8 on esimerkki Scrumin Sprint Burndown-kaaviosta. Pystyakseli kuvaa jäljellä olevaa työmäärää tunteina ja vaaka-akselilla on Sprintin aikajana.



Kuva 6 Sprint Burndown Chart

Lisäksi Scrum For Team System tarjoaa muita raportteja, joita ei kuvata Scrumissa, mutta jotka voivat olla hyödyllisiä projektinhallinnan kannalta. Näitä ovat mm. Bug Count, Bug Found and Fixed, Bug History, Development to Test Cycle Time, Product Backlog Cumulative Flow, Sprint Backlog Cumulative Flow ja Sprint Overview Chart. (Scrum For Team System)

4.2.2 Visual Studio Team System ja Scrum

Tarkastelun lopputuloksena voidaan huomata, että Visual Studio Team System tukee Scrum-prosessia lähinnä Scrumin Backlogien avulla. Muita hyötyjä voidaan saavuttaa projektinhallinnan osalta. Projektinhallinnan tehostamisesta voi olla hyötyä esim. tilanteissa, joissa Scrum-prosessia joudutaan skaalaamaan.

Tarkastelun jälkeen voidaan kysyä, onko .Net Frameworkin mukaisessa ohjelmistokehityksessä tukea Scrumille. Mielestäni parempi lähestymistapa voisi olla sellainen, jossa mietitään miten Scrum auttaa .Net Frameworkin mukaista ohjelmistokehitystä. Schwaberin mukaan Scrum paljastaa ohjelmistokehitysprosessin vikoja yrityksen sisällä. Tämä voi johtaa parempien kehitysmenetelmien, -teknologioiden ja käytänteiden käyttöönottoon ja omaksumiseen yrityksissä, ja näitä ominaisuuksia Microsoftin Visual Studio Team System voisi tarjota.

5 YHTEENVETO

Scrum on ketterä menetelmä, jonka avulla saavutetaan hyötyjä ohjelmistokehitysprojekteissa. Ketterät menetelmät usein painottavat ihmisten, kommunikoinnin ja muutoksen merkitystä jäykkien menetelmien ja prosessien sijaan.

Scrumin keskeiset piirteet voi jaotella esim. rooleihin, artefakteihin ja prosessiin. Rooleja ovat: Scrum Master, Scrum Team ja Product Owner. Artefakteja voivat olla esim. Product Backlog, Sprint Backlog, Impediment List ja Burndown kaaviot. Prosessin voi nähdä käsittävän Sprintit, päivittäiset palaverit, Sprint Planningin ja Sprint Retrospectiven.

Microsoftin kehittämä .Net framework on laajalti käytetty ohjelmistojen kehitysalusta. Se käsittää ohjelmistokomponenttikirjaston, kehitysympäristön ja ajoympäristön. Kehitysympäristön tärkein kokonaisuus on Visual Studio tuoteperhe. 'Visual Studio Team System' – kokonaisuuden Visual Studio Team Suite on client-ohjelmistotuotevalikoima, jonka avulla kehitetään ohjelmistoja. Prosessinhallintaa varten on työkaluina Visual Studio Team Foundation Server-alustalle kehitetyt työkalut.

Visual Studio Team Foundation pitää sisällään mm. Microsoft Solutions Foundation menetelmän mukaiset prosessinhallintatyökalut. Nämä skaalautuvat työkalut tukevat myös Scrumin mukaista ohjelmistojen kehittämistä erilaisten mallipohjien avulla.

Selkein mallipohja on Scrum For Team System, jonka taustahenkilönä on toiminut mm. Ken Schwaber, joka on yksi Scrum menetelmän kehittäjistä. Scrum For Team System tarjoaa kaikki Scrumin kannalta oleelliset työkalut prosessinhallintaan.

Vaikka Scrumin tukeminen jäykkien prosessityökalujen avulla ei suoranaisesti parantaisikaan puhdasta Scrumin mukaista ohjelmistokehitysprojektia,

prosessityön tukemisesta on kuitenkin usein nähty saatavan muita hyötyjä ohjelmistojen kehitystyössä.

LÄHTEITÄ

Schwaber K., Beedle M., 2004. Agile Project Management With Scrum, Prentice Hall

Schwaber K., 2004, Agile Project Management With Scrum, Microsoft Press

Takeuchi H., Nonaka I., 1986, The New New Product Development Game, Harvard Business Review

Newkirk J., Scott W., 2007. Visual studio team system: better software development for agile teams, First edition

Schwaber K., SCRUM Development Process [Viitattu 14.4.2009]. Saatavilla internetistä: <<http://www.jeffsutherland.com/oopsla/schwapub.pdf>>.

Scrum.dk [Viitattu 14.4.2009]. Saatavilla internetistä: <http://scrum.dk/about_scrum>.

Deemer p., Benefield G., Larman C., Vodde B., 2009. The Scrum Primer. Viitattu [14.4.2009]. Saatavilla internetistä: <http://scrumtraininginstitute.com/home/stream_download/scrumprimer>.

.NET Framework Conceptual Overview [Viitattu 14.4.2009]. Saatavilla internetistä: <<http://msdn.microsoft.com/en-us/library/zw4w595w.aspx>>.

Bodea C-N., 2005, Agile Software Project Management Methodologies, Economy Informatics, 1-4 2005

Nelson D., 2006, Pro Visual Studio 2005 Team System, Apress 2006

Scrum For Team System [Viitattu 14.4.2009]. Saatavilla internetistä: <<http://www.scrumforteamssystem.com/en/default.aspx>>.

Ketterät käytännöt.fi [Viitattu 14.4.2009]. Saatavilla internetistä:

<<http://www.ketteratkaytannot.fi/fi-FI/Ketteryys/ScrumForTeamSystem/>>.

eScrum for Microsoft Visual Studio Team System [Viitattu 14.4.2009]. Saatavilla internetistä:

<<http://www.microsoft.com/downloads/details.aspx?FamilyID=55A4BDE6-10A7-4C41-9938-F388C1ED15E9&displaylang=en#Overview>>.

Scrumptious [Viitattu 14.4.2009]. Saatavilla internetistä:

<<http://www.codeplex.com/scrumptious>>