

UPGRADE OF AGILE DEVELOPMENT SUPPORT TOOL

Petr Trávník

Master Degree Programme (2), FIT BUT

E-mail: xtravn20@stud.fit.vutbr.cz

Supervised by: Jitka Kreslíková

E-mail: kreslika@fit.vutbr.cz

Abstract: The goal of this article is to identify and implement upgrades for agile development support tool used by Corporate Technology department of Siemens, s.r.o. in Brno. To achieve the goal, article focuses on Scrum agile methodology used by the department and then analyzes the most used professional tools for an inspiration. The goal of the improvements is to further optimize the software development process and to reduce the administrative over-head.

Keywords: Agile development, Scrum, MVC, Team Foundation Server, MS Outlook, Software Support Development

1 ÚVOD

Cílem práce je nalézt potenciální možnosti pro zlepšení nástroje pro podporu agilního vývoje, který používá oddělení Corporate Technology společnosti Siemens v Brně. Oddělení aplikuje agilní vývoj pomocí metodiky Scrum, která je popsána v první části článku. Následuje analýza profesionálních nástrojů pro podporu agilního vývoje, která slouží jako základ při hledání dalších možných rozšíření nástroje oddělení Corporate Technology. Výstupem článku je pak popis aktuálního stavu nástroje a možných rozšíření, která jsou následně implementována a integrována do firemního prostředí.

2 AGILNÍ VÝVOJOVÁ METODIKA SCRUM

Agilní vývoj pomocí Scrum přináší výsledky ve dvou až čtyřtýdenních iteracích označovaných jako *sprinty*. Samostatně řízené týmy se skládají ze tří až devíti členů. Metodika zahrnuje speciální role *mistra Scrumu*, který zajišťuje dodržování zásad Scrumu a dále *vlastníka produktu*, který zajišťuje komunikaci se zákazníkem a správu požadavků na produkt. Scrum dále sestává z několika artefaktů, jako je *produktový katalog* obsahující požadavky na produkt nebo *katalog sprintu* obsahující požadavky, které se budou řešit v aktuálním *sprintu* [2].

Jednou z částí Scrum, která bývá mnohdy opomíjena, je *revize kódu*. Skupinová revize spočívá ve schůzce, na které členové vývojového týmu postupně prezentují a diskutují kód. Kromě snížení chybivosti kódu přináší skupinová revize mnoho dalších výhod. Členové týmu si díky ní předávají zkušenosti mezi sebou a zároveň vede k lepší čitelnosti a struktuře kódu.

Po skončení každého *sprintu* probíhá *retrospektiva*, kdy se vývojový tým sejde a hodnotí průběh právě skončeného *sprintu*. Cílem je identifikovat a zamezit opakování nastalých problémů, ale i ocenit příležitosti, které tým využil. *Retrospektiva* je hlavní součástí Scrumu pro neustálé zefektivňování vývoje.

3 ANALÝZA PROFESIONÁLNÍCH NÁSTROJŮ

Součástí práce je také analýza v současnosti nejpoužívanějších profesionálních nástrojů pro podporu agilního vývoje (dle průzkumu [3]). Srovnání zahrnovalo 7 různých nástrojů a cílem bylo především

nalézt funkce, které mohou vývojovému týmu nejvíce usnadnit práci. Nejširší podporu pro metodiku Scrum nabízely nástroje TFS 2012, JIRA a VersionOne.

Nástroj	Produktový katalog	Retrospektiva	Revize sprintu	Graf výkonnosti	Revize kódu
JIRA	✓	✓	✓	✓	externě
TFS	✓	✓	✓	✓	✓
ScrumWorks	✓			✓	
Rally	✓		✓	✓	
Mingle	✓	✓		✓	
VersionOne	✓	✓	✓	✓	externě
Assembla	✓			✓	✓

Tabulka 1: Podpora artefaktů Scrum

4 ANALÝZA SOUČASNÉHO STAVU

Současný nástroj poskytuje čtyři hlavní funkce, které pomáhají týmu optimalizovat vývojový proces. Graf výkonnosti týmu dodává přehled o aktuálně odpracovaných a zbývajících hodinách *sprintu* spolu s informací o ideálním trendu. Modul Plánování zpřístupňuje přehled naplánovaných položek a časové dispozice jednotlivých členů týmu. *Revize sprintu* poskytuje přehled naplánovaných a dokončených položek a statistiky o plánovaných a nespálených hodinách. Poslední modul pak zobrazuje dlouhodobé trendy výkonu vývojového týmu. Bylo identifikováno několik potenciálních vylepšení současného nástroje, která by přinesla další úsporu času optimalizací administrativy a ještě více zefektivnila vývojové procesy metodiky Scrum.

4.1 REVIZE KÓDU

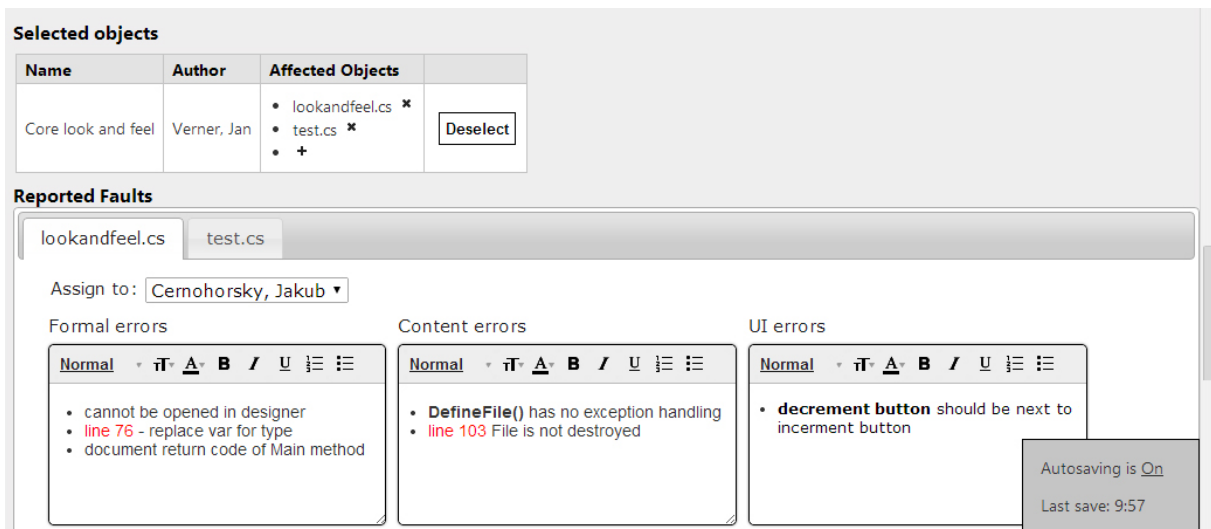
V současnosti tým využívá skupinové *revize kódu*, jejichž výsledky se zapisují formou dokumentu na týmové wiki. Zaslání pozvánek, kontrola docházky a správa jednotlivých úkolů pro opravy nalezených chyb jsou prováděny ručně pomocí poštovního klienta Outlook. Outlook nabízí API, které by umožnilo automatizovat správu setkání i úkolů pomocí funkcí pro odesílání zpráv a filtrování příchozích pošty (včetně aktualizací stavu úkolů a docházky). Správa objektů navrhovaných pro revidování je součástí wiki dokumentu, a proto se nabízí zefektivnění vytvořením databáze, která by požadavky uchovávala globálně. Výstupem každé revize jsou také obecná ustanovení, která definují dlouhodobé zásady k dodržování. Ta jsou momentálně spravována v rámci jednotlivých dokumentů a jejich souhrnné zobrazení na viditelném místě by poskytlo zvýšení přínosu *revize kódu* pro tým.

4.2 RETROSPEKTIVA

Dle zásad metodiky Scrum provádí vývojový tým *retrospektivu* po skončení každého *sprintu*. *Retrospektiva* se provádí na skupinové schůzce a výstupem je dokument na týmové wiki. Opět se zde tak nachází prostor pro automatizaci rozeslání pozvánek a docházky. Integrace do současného nástroje by pak pomohla zpřehlednit výsledky. V rámci *retrospektiv* se dále přidělují úkoly, u kterých můžeme opět automatizovat správu.

4.3 TÝDENNÍ SETKÁNÍ

Týdenní setkání je artefakt, v rámci kterého vývojový tým diskutuje s *vlastníkem produktu* o aktuálním průběhu *sprintu*, dokončených a aktuálně probíhajících úkolech a případných problémech. K setkání se předem připravuje zpráva o stavu jednotlivých úkolů. Ta je momentálně vytvářena ručně



Obrázek 1: Ukázka obrazovky již implementovaného modulu pro Revizi kódu.

a přináší velkou časovou náročnost. Automatizování vytváření zprávy pomocí spojení s Team Foundation servery oddělení by přineslo časovou úlevu.

4.4 GRAFICKÁ NÁSTĚNKA

Pro zobrazení aktuálního stavu úkolů ve *sprintu* si tým momentálně vede nástěnku na tabuli umístěné na pracovišti. Pro reprezentaci jednotlivých úkolů používá „lístečky“ a k jejich řazení využívá metodu Kanban. Převedení nástěnky do elektronické podoby by přineslo zefektivnění celého procesu.

5 ZÁVĚR

Na základě konzultace se zástupcem oddělení Corporate Technology byly pro implementaci zvoleny moduly pro *revizi kódu* a *retrospektivu*. Důraz byl kladen na všestrannost aplikace tak, aby bylo možné nasazení pro všechny týmy oddělení. Implementované moduly integrují tři různé nástroje v jeden a nabízí kvalitní grafické rozhraní díky využití technologie AJAX především pro průběžné ukládání dat. Dle kvalifikovaného odhadu vedoucího oddělení přinesly moduly v souhrnu 15% úsporu času u ovlivněných procesů. Jejich kvalita se navíc zvýšila zdůrazněním pozitivních dopadů na tým, čehož bylo dosaženo díky globální správě revidovaných objektů, závěrů *revizí* i závěrů *retrospektiv*.

PODĚKOVÁNÍ

Tento příspěvek vznikl za podpory grantu VUT FIT, FIT-S-14-2299, "Výzkum pokročilých metod ICT a jejich aplikace".

REFERENCE

- [1] Kolektiv autorů: Scrum Guide [online]. <http://www.scrum.org/Scrum-Guide>, [cit. 2013-12-28].
- [2] Schwaber, M., Beedle, K.: Agile software development with Scrum. Upper Saddle River: Prentice Hall, 2002, iSBN 0-13-067634-9.
- [3] Kolektiv autorů: Survey of Agile Tool Usage and Needs. Agile Conference (AGILE), 2011, iSBN 978-1-61284-426-8.