

Скрам – људска страна развоја софтвера

Валентино Вранић

Slovak University of Technology in Bratislava, Slovakia

Пословни факултет у Ваљеву, Универзитет Сингидунум

`vranic@fiit.stuba.sk`
`http://fiit.stuba.sk/~vranic/`

29. март 2013.

Увод

Технологија и
софтвер

Пројекат

Људи

Scrum

Сумаризација

- ▶ Технологија као појам има материјалне конотације – као да подразумева одговарујуће алате, уређаје, апарате, машине и сл.
- ▶ Постоје и софтверске технологије
- ▶ Гр. *τέχνη* – уметност, способност, вештина, занат
- ▶ Софтверске технологије су пре свега управо то

Технологија и производња

- ▶ Масовна производња
- ▶ Специјализација рада
- ▶ Покретна трака



Ford assembly line, 1913.¹

¹http://en.wikipedia.org/wiki/File:Ford_assembly_line_-_1913.jpg

- ▶ Технологије омогућавају ефективнији рад уз помоћ аутоматизације
- ▶ Али технологије треба да служе људима, не обрнуто
- ▶ Људи живе и кроз рад
- ▶ Не треба заборављати на природне људске особине
- ▶ Непрестана потражња за новим и бољим производима захтева кретивност, а кретивност је немогуће аутоматизовати
- ▶ Организовано настојање у сврху испуњења замисли – пројекат

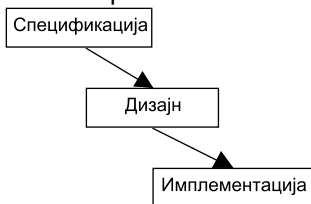
- ▶ Шта треба реализовати: *спецификација* (захтева)
- ▶ Како реализовати: *дизајн*
- ▶ Сама реализација: *имплементација* (лат. *implere* – испуњење)
- ▶ Између спецификације и дизајна се често издваја *анализа*
- ▶ Типичне фазе – боље речено активности – реализације софтверског пројекта (али важе и шире)

Друге активности у склопу реализације пројекта

- ▶ После имплементације се као посебне активности издвајају и тестирање и интеграција
- ▶ Као последња активност се у ширем схватању реализације пројекта издваја и одржавање
- ▶ Овде ћемо се задржати на поједностављеном схватању реализације пројекта:

спецификација – дизајн – имплементација

- ▶ Идеализована представа: пројекат се „прелије“ глатко кроз све активности

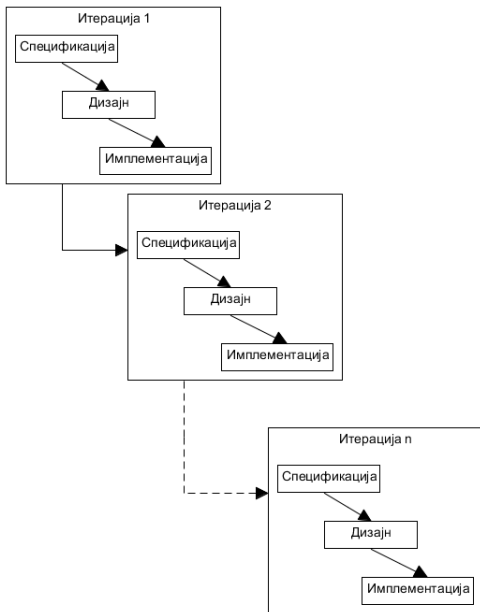


- ▶ Ако процес и протекне глатко, клијент најчешће може да констатује да није добио то што је хтео
- ▶ У пракси се приликом анализе спецификације наилази на проблеме као што су:
 - ▶ непотпуност или подспецификација (енгл. underspecification)
 - ▶ неједнозначност
 - ▶ неконзистенција или противречност
 - ▶ преспецификација (енгл. overspecification)

Непотпуност спецификације

- ▶ Непотпуност је често последица недоречености – могуће је исправити је у сарадњи с клијентом
- ▶ Спецификација бива непотпуна и с обзиром на то што *ће* клијент хтети да добије на крају
- ▶ То је нешто друго у односу на то што је клијент хтео на почетку – представе клијента се мењају, а на то утиче и сам развој софтвера
- ▶ Каже се да клијент не зна шта хоће
- ▶ Ово је унутрашњи проблем спецификације као такве: није могуће направити потпуну спецификацију пре реализације

Итерациони и инкрементални процес



Скрам – људска
страна развоја
софтвера

Валентино
Вранић

Увод

Технологија и
софтвер

Пројекат

Људи

Scrum

Сумаризација

Итерација и инкремент

- ▶ Итерација (лат. iterum – поново) представља један прелаз приликом реализације пројекта
- ▶ Резултат итерације је инкремент (нараст; lat. *increscere* – нарасти)
- ▶ Пре сваке итерације је прилика за промену захтева
- ▶ После сваке итерације настаје одговарајући производ – с тенденцијом да буде функционалан
- ▶ Клијент по свакој итерацији види и евентуално може и да и испроба производ – променом захтева управља његовим развојем
- ▶ Квалитет реализације пројекта зависи највише од – људи

- ▶ Праве грешке и уче на њима
- ▶ Тешко подносе монотони и једноличан рад
- ▶ Радо раде на занимљивим стварима
- ▶ Схватају само термине који су довољно близу
- ▶ Прилагођавају се механизмима вредновања свог рада
- ▶ Треба им лични контакт и непосредна комуникација с другим људима
- ▶ Потребно им је да виде резултате свог рада
- ▶ Не воле управљање споља

- ▶ Класични приступ реализацији пројеката или организованој производњи не узима у обзир природне људске особине
- ▶ Или можда их ипак узима у обзир – покушавајући да их потисне и победи над њима
- ▶ Некад то и успе, али то је Пирова победа: резултат је група послушних – али демотивисаних – радника који су свој рад прилагодили начину вредновања
- ▶ То се зове менаџмент
- ▶ Да ли је могуће ово избећи и из ових особина извући корист?

- ▶ Скрам (Scrum) је оквир (енгл. framework) за реализацију пројеката
- ▶ Скрам није (употпуњена) метода
- ▶ Гарантује ипак да ће се рад концентрисати на најбитније ствари
- ▶ Произилази из природних људских особина

Улоге у Скраму

- ▶ Team – тим
- ▶ ScrumMaster – СкрамМастер
- ▶ Product owner – власник производа

Власник производа

- ▶ Власник производа заступа клијенте и друге заинтересоване (енгл. stakeholders)
- ▶ Може да има свој сопствени тим
- ▶ Одговара за производ као такав и за повраћај инвестиција (енгл. return on investment, ROI)
- ▶ Интерпретира и објашњава захтеве
- ▶ Одржава уређен списак неодраженог посла на производу – product backlog

- ▶ Тим ствара производ
- ▶ Има 3–7 чланова
- ▶ Чланови немају стриктно диференциране обавезе
- ▶ Тим сам управља собом

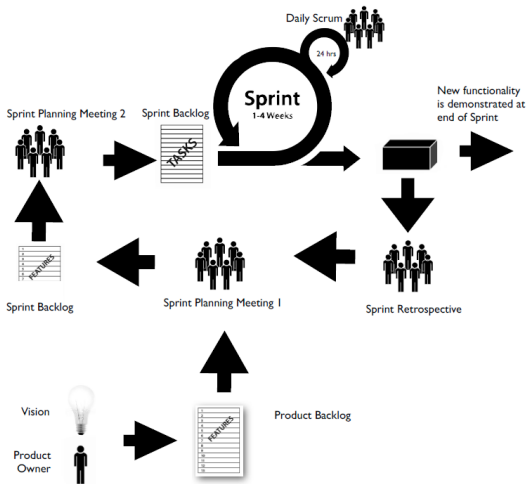
- ▶ Тим одређује колико посла ће узети у једном *спринту*, који представља итерацију
- ▶ Запис о овоме садржи списак неодраженог посла у спринту – *sprint backlog*: списак ставки одобраних са врха списка неодраженог посла на производу расписаних у задатке (енгл. *task*)
- ▶ Овај посао тим треба да заврши – резултат је увек потенцијално испоручив производ (обогаћен о инкремент у односу на претходни спринт)
- ▶ Спринт је временски ограничен – обично на 1–4 недеље – треба га одрадити одређеном брзином (преспоро трчање престаје да буде трчање)
- ▶ Али тим ради у нормално радно време
- ▶ Прати се колико посла остаје у спринту и како се обим посла смањује – *burndown chart*

- ▶ СкрамМастер помаже у одстрањивању препрека (енгл. impediments)
- ▶ СкрамМастер брани тим од захтева ван пројекта – често потичу од власника пројекта
- ▶ СкрамМастер никад не интервенише у раду тима
- ▶ Рад тима не контролише нико – основа за изградњу поверења
- ▶ Тим може да отпусти СкрамМастера

Схема Скрама

Скрам – људска
страна развоја
софтвера

Валентино
Вранић



Увод

Технологија и
софтвер

Пројекат

Људи

Scrum

Сумаризација

Jeff Sutherland's Scrum Handbook, Scrum Training Institute Press, 2010.²

²<http://jeffsutherland.com/scrumhandbook.pdf>

- ▶ Откривање грешке се у Скраму узима као нешто веома позитивно – прилика за усавршавање
- ▶ Одстрањивање грешака се неодлаже – производ се не испоручује знајући да у њему постоје грешке
- ▶ Тим се труди да се из грешке извуче наук – и да на одговарајући начин коригује процес (повратна спрега)

- ▶ Могуће је комуницирати и путем документације
- ▶ Људи имају тенденцију да стварају опширну документацију, али нико је не чита (write only documentation) – тиме се производи отпад
- ▶ Скрам је заснован на непосредној комуникацији, брзој и ефикасној
- ▶ Документација се користи за запис тога што треба памтити
- ▶ Непосредна комуникација омогућава флексибилну и координисану реакцију тима

- ▶ Приоритет ставки у списку неодраженог посла на производу одређује власник производа
- ▶ Процена времена потребног за појединачне ставке је од кључног значаја за планирање
- ▶ Тешко је планирати у апсолутном времену – процена у Скраму је зато релативна: свакој ставки се додели одговарајући број бодова
- ▶ Planning poker³
- ▶ Тим с врха списка неодраженог посла на производу узима толико ставки колико сматра да ће успети да савлада у једном српину
- ▶ После неколико итерација се вредност посла (збир броја бодова изабраних ставки) устали – означава се као брзина (енгл. velocity)

³ <http://www.crisp.se/bocker-och-produkter/planning-poker>

- ▶ Daily Scrum
- ▶ Кратак састанак на почетку сваког дана – обично се стоји
- ▶ Говоре само чланови тима
- ▶ Сваки члан каже:
 - ▶ Шта је радио јуче
 - ▶ Шта ће радити данас
 - ▶ Какве препреке га у томе спречавају
- ▶ На састанку се не дискутује
- ▶ Ништа не спречава чланове тима да размотре све што је потребно током дана, а то и са СкрамМастером

- ▶ Скрам се опире производњи непотребних ствари – отпада
- ▶ Континуално се одстрањују препреке
- ▶ Врши се ретроспектива и оцена спринта
- ▶ Не користе се метрике за оцену продуктивности – све је видљиво
- ▶ Коришћење ових метрика би водило ка прилагођењу рада начину његовог оцењивања

Порекло и исходишта Скрама

- ▶ Jeff Sutherland⁴
- ▶ Назив Скрам потиче из рагби аналогije⁵
- ▶ Скрам произилази из приступа неких јапанских компанија – узор је Тојота
- ▶ Излагање философије Скрама се често ослања о јапанску терминологију: muri – muda – mura, kaizen, sensei. . .
- ▶ Скрам ипак није условљен јапанском културом – заснован је на општим људским особинама
- ▶ Агилни приступ (angl. agile)
- ▶ Scrum Alliance⁶
- ▶ Организациони узори – семинални рад Коплина и Харисона⁷

⁴<http://jeffsutherland.com/scrumhandbook.pdf>

⁵H. Takeuchi and I. Nonaka. The New New Product Development Game. Harvard Business Review, 1986.
<http://www.sao.corvallis.or.us/drupal/files/TheNewNewProductDevelopmentGame.pdf>

⁶<http://www.scrumalliance.org/>

⁷J. O. Coplien and N. B. Harrison. Organizational Patterns of Agile Software Development. Prentice Hall, 2004.

- ▶ Напредак води ка све већој аутоматизацији
- ▶ Људска креативност је ипак кључна и незаменљива
- ▶ Технологије треба да служе људима, а не обрнуто
- ▶ Реализација пројеката може да буде организована тако да узима у обзир природне људске особине у прилог резултата
- ▶ На овој основи је постављен Скрам као оквир реализације пројеката

fiit.sk/~vranic

Увод

Технологија и
софтвер

Пројекат

Људи

Scrum

Сумаризација