

■ BPMN-modeli procesov za strukturiran zajem uporabniških zgodb

Marina Trkman, Marjan Krisper

Univerza v Ljubljani, Fakulteta za računalništvo in informatiko, Tržaška cesta 25, 1000 Ljubljana
marina.trkman@gmail.com; marjan.krisper@fri.uni-lj.si

Izvleček

Nekatere agilne metodologije, kot sta Scrum in XP, za zajem zahtev naročnika uporabljajo uporabniške zgodbe. Plan razvojnega projekta sloni na definiranih uporabniških zgodbah na začetku projekta. Značilno za agilni razvoj po Scrumu oz. XP-ju je, da se seznam uporabniških zgodb med potekom projekta (pre)pogosto spreminja. Ta značilnost je večkrat neupravičeno izkoriščena kot posledica slabše opravljenega globalnega pregleda nad zahtevami uporabnika že na začetku projekta. Obstoječe tehnike zbiranja uporabniških zgodb na začetku projekta so namreč premalo strukturirane, da bi bil verodostojen plan projekta, ki sloni na njih. Za zmanjšanje te negotovosti smo v članku predstavili tehniko, ki sloni na predhodnem (lahko samostojnem) projektu modeliranja poslovnih procesov organizacije. Na primeru smo prikazali postopek definiranja uporabniških zgodb iz BPMN-modelov procesov.

Ključne besede: uporabniške zgodbe, agilne metodologije, notacija BPMN, modeliranje poslovnih procesov.

Abstract

BPMN-process Models as a Tool for Structured Gathering of User Stories

Agile methods like Scrum and XP use user stories for gathering customer's requirements. Planning of such development project is based on user stories defined at the beginning of the project. It is typical for agile development with Scrum or XP that user stories too often get modified during the project. The reason for that is an insufficient global overview of user demands at the beginning of the project. The existing techniques for gathering user stories are not structured enough for the project plan to be credible. In order to reduce this weakness we introduced more structured technique for gathering user stories. The technique is based on pre-project of business process modeling of the organization. We demonstrated the procedure of using BPMN models of business processes to gather user stories for agile development.

Key words: user stories, agile methods, BPMN notation, business process modeling.

1 UVOD

Tradicionalno so zahteve uporabnikov glede programske opreme zapisane v specifikacijah. Zmotno je prepričanje, da zaradi definiranih zahtev ni prostora za nesoglasja med deležniki: da bodo razvojniki natančno vedeli, kaj razviti, in testerji, kaj testirati, ter da bo stranka dobila prav takšno programsko opremo, kot jo je naročila. Avtorji članka smo tako kot Cohn (2004) mnenja, da bo stranka dobila programsko opremo, ki odraža razvojnikovo interpretacijo specifikacije. Če želimo zmanjšati neskladje med zahtevanim in razvitim, moramo preiti od pisanja zahtev na pogovarjanje o njih (Cohn, 2004). Tudi pri verbalni komunikaciji je lahko težava z medsebojnim razumevanjem, vendar je tu priložnost za hitro povratno informacijo. Ravno zato so uporabniške zgodbe agilnih razvojnih metod, kot sta Scrum in XP, danes vse bolj popularne. Temeljna razlika med njima je, da je Scrum bolj vodilo menedžmentu projekta, XP pa vodilo razvojnemu timu

(Dyba et al., 2008). Skupen imata poudarek na komunikaciji z naročnikom programske opreme od prve do zadnje vrstice kode, kar je radikalna sprememba za tiste, ki prihajajo iz tradicionalnega »slapovnega« razvoja.

Namen članka je izpostaviti problem planiranja projekta z uporabniškimi zgodbami ter ponuditi rešitev za učinkovitejši zajem uporabniških zgodb na začetku projekta s pomočjo BPMN-modelov procesov. Ker je definiranje uporabniških zgodb v rokah naročnika, predlagamo, da ta predhodno v posebnem projektu popiše svoje poslovne procese. S tem se pri odločanju o njihovi informatizaciji lažje odloča, katere aktivnosti je treba računalniško podpreti. Dodatno pa tudi razvijalci lažje razumejo poslovanje podjetja, katerega bodo računalniško podprli.

V prvem razdelku predstavljamo uporabniške zgodbe ter obravnavano problematiko. Nato sledijo

napotki iz literature za sestavljanje imen uporabniških zgodb, katere smo kasneje uporabili na primeru, ter tehnike za definiranje celotnega seznama. Poudarili smo slabosti najpopularnejše tehnike. Za premagovanje teh slabosti smo ponudili rešitev s področja menedžmenta poslovnih procesov (MPP), natančneje s pomočjo notacije BPMN (Business Process Modeling Notation).

2 UPORABNIŠKE ZGODBE

V splošnem so agilne metode učinkovit pristop k razvoju manjših informacijskih sistemov (Oz, 2009). Zanje je značilna racionalnost, timsko delo, prilagodljivost, strukturiranost, manj dokumentacije, preprostost, kreativnost in improvizacija (Papadopoulos et al., 2008). Iz tega izhajajo štiri glavna načela (Krisper et al., 2003):

- posamezniki in njihova komunikacija so pomembnejši kot sam proces in orodja;
- delujoča programska oprema je pomembnejša kot popolna dokumentacija;
- vključevanje (sodelovanje) uporabnika je pomembnejše kot pogajanje na podlagi pogodb;
- upoštevanje sprememb je pomembnejše od sledenja planu.

Agilni metodi, kot sta Scrum in XP, uporabljata modelirno tehniko, imenovano uporabniške zgodbe (Ambler, 2002), za komuniciranje med naročnikom programske opreme in razvojnim timom. Uporabniška zgodba je kratka besedilna specifikacija neke interakcije s sistemom (Deckter et al., 2007). Cohn (2004) dodaja, da predstavlja neko vrednost uporabniku ali stranki. Tipično so na uporabniški kartici poleg imena še drugi vidiki (Cohn, 2004):

- morebitni dodatni opis kot opomnik zaključene vsebine,
- pogovori, ki razvojnika spomnijo na detajle,
- testi, ki preverjajo delovanje razvitih detajlov.

Cilj uporabniških zgodb ni dokumentiranje vsakega detajla neke funkcionalnosti, temveč zapis v nekaj stavkih, ki stranko in razvojnika spomni, o čem je treba še razpravljati. Veliko komunikacije med njima se odvija po elektronski pošti, za kompleksnejše primere pa je priporočeno osebno srečanje.

Razlogi, zakaj se mnogi odločajo za agilni razvoj z uporabniškimi zgodbami, so (Cohn, 2004):

- ker so razumljive tako razvojnikom kot naročnikom;

- ker predstavljajo obvladljivo zaključeno celoto, so primerne obsega za programiranje, testiranje in planiranje izdaj programske opreme;
- ker spodbujajo iterativnost komunikacije med razvojnikom in uporabnikom;
- ker so opisane v različnih nivojih podrobnosti glede na prioriteto obravnave;
- ker uporabniki sodelujejo pri načrtovanju prototipa in vplivajo nanj;
- ker vplivajo na povečevanje tihega znanja: več kot razvojniki in stranka sodelujejo iz oči v oči, več znanja nastane znotraj tima.

2.1 Priprava seznama uporabniških zgodb po navodilih literature

Cohn (2004) predlaga, da se zbiranje uporabniških zgodb začne z definiranjem uporabniških vlog, katerim bo namenjena nova programska oprema. Avtor predlaga imenovanje uporabniških zgodb po predlogi Connextra (Cohn, 2004; Cohn, 2006): »Jaz kot (uporabniška vloga) želim (funkcija), da bi lahko (dodana vrednost funkcije).«

Priporočeno je, da je vsaka zgodba namenjena enem uporabniku. Idealno je, če naročnik sam definira začetni seznam uporabniških zgodb. Večkrat v praksi pri tem pomagajo razvojniki z organiziranjem in vodenjem delavnic; naročnikom, ki že sami predlagajo uporabniške zgodbe, pa pomagajo z različnimi predlogi za izboljšanje seznama. Odgovornost za pripravo uporabniških zgodb ostaja v rokah naročnikov in se ne more prenesti na razvojnike. Naročnik je tudi tisti, ki določa prioriteto uporabniških zgodb. Iz teh nekaj pravil Scruma lahko razberemo, kako je pomembno, da naročnik razume, kaj želi imeti že ob koncu projekta.

Na voljo so različne tehnike za definiranje začetnega seznama uporabniških zgodb (Cohn, 2004).

▪ Intervjuji

Intervjuvani so zaposleni, tj. uporabniki programske opreme, ki imajo različne vloge v podjetju. Najboljši način za pridobivanje informacij o uporabniških potrebah je uporaba odprtih vprašanj brez konteksta. Iz teh izhajajo specifična odprta vprašanja. Na podlagi odgovorov lahko sestavimo prvi seznam uporabniških zgodb.

▪ Ankete

S pomočjo vprašanj v anketi lahko pridobimo več informacij o uporabniških zgodbah, definiranih z intervjuji, in jim določimo pomembnost.

▪ Delavnice

Na delavnici so navzoči deležniki, ki lahko pomagajo sestaviti seznam uporabniških zgodb. Tehnika kombinira elemente viharjenja možganov in prototipiranja. Medtem ko udeleženci viharijo možgane na temo, kaj vse bi radi počeli z novo programsko opremo, nekdo zgradi prototip nižje reprezentativnosti (angl. low-fidelity prototype; v nadaljevanju prototip). Za razliko od tradicionalnega prototipa ni cilj identificirati zaslonske maske, temveč delovni tok (t. i. proces).

Na začetku razvoja prototipa nizke reprezentativnosti se odločimo, katero uporabniško vlogo bomo obdelali. Za vsako uporabniško vlogo namreč narišemo svojega. Skico začnemo s praznim kvadratom in vprašanjem, »kaj lahko uporabnik (natančno določena uporabniška vloga) naredi na prvi strani aplikacije«. Udeleženci nato s pomočjo tehnike viharjenja možganov nizajo aktivnosti uporabnika, ki so samo zabeležene. Poudarek je na količini, ne na kakovosti. Vsako aktivnost uporabnika zabeležimo v novem povezanem kvadratu, ki predstavlja novo uporabniško aktivnost. Sprehod skozi tok dela pomaga pri identificiranju uporabniških zgodb. Sprašujemo se o morebitnih uporabniških napakah, kaj bi lahko zmedlo uporabnika, kaj lahko nato stori, kaj je opcijsko in kaj ne, kaj se dogaja pogosto in kaj poredkoma, katere dodatne informacije uporabnik potrebuje, da opravi nalogo ipd.

Po besedah avtorja so delavnice najučinkovitejši način pridobivanja uporabniških zgodb, ki pa ima določene slabosti. Prva slabost je ta, da je sledenje toku uporabniških zgodb do konca težko sledljivo (Cohn, 2004). Druga slabost je, da s prototipiranjem že na začetku prisilimo stranko, da razmišlja o svojem bodočem grafičnem vmesniku (angl. Graphical user interface, GUI). Agilne metodologije učijo razvojnike, da grafični vmesnik naredijo čim bolj proti koncu projekta. Ravno zato naročnikove prototipe takoj po definiranju začetnega (prvotnega) seznama uporabniških zgodb »zavržejo« oz. jih ne uporabijo nikoli več. Naročnik si lahko s takim pristopom ustvari napačno pričakovanje o reprezentativnosti končnega izdelka že na začetku projekta, kar se lahko kasneje pokaže kot nezadovoljstvo. Tretja slabost pa je ta, da se pri obstajajočih tehnikah velikokrat pojavijo prevelike uporabniške zgodbe, katere morajo razvojniki kasneje »razbijati« na manjše. To povzroča spremembe na seznamu vseh uporabniških zgodb in posledično tudi na planu poteka projekta.

3 PROBLEMSKA DOMENA – PLANIRANJE AGILNEGA RAZVOJNEGA PROJEKTA Z UPORABNIŠKIMI ZGODBAMI

Temeljni problem uporabniških zgodb z vidika planiranja (predvsem pri agilni metodologiji Scrum) so spremembe na seznamu uporabniških zgodb med projektom. Zato smo se v članku osredinili na izboljšanje začetne faze agilnega razvoja, in sicer natančneje na boljše definiranje začetnega seznama uporabniških zgodb. Poiskali smo kritično slabost in ponudili boljše rešitev.

Spoznali smo, da se kljub mnogim prednostim uporabe uporabniških zgodb pojavljajo težave. Lee idr. (2003) so identificirali neprimeren mehanizem za zajem in organizacijo zahtev kot temeljno slabost dela z uporabniškimi zgodbami. Po njihovem mnenju literatura ponuja nezadovoljive rešitve na tem področju. Brez primerne mehanizma za zajem in organizacijo uporabniških zahtev pa je naročniku nemogoče dostaviti natančno tisto, kar je želel v pogodbi.

Tako predstavlja iterativnost med razvojnikom in uporabnikom na eni strani prednost, saj med projektom omogoča večnivojsko definiranje uporabniških zgodb ter dodajanje le-teh na seznam vseh uporabniških zgodb in odzemanje z njega. Na drugi strani pa lahko to spreminjanje seznama uporabniških zgodb med izvedbo razvojnega projekta definiramo kot izvor marsikatero težave planiranja projekta. Vsako dodajanje, odzemanje in spreminjanje uporabniških zgodb na seznamu lahko privede do časovnega zamika projekta (ob predpostavki, da ostanejo drugi viri projekta nespremenjeni). Naročnik rad vnaprej v pogodbi definira, kdaj lahko pričakuje zelene rezultate.

Nerealno je pričakovati, da bi bile v začetku projekta dokončno definirane vse zahteve (oz. v našem primeru vse uporabniške zgodbe) nove programske opreme. Tako je cilj članka prikazati pristop za zmanjšanje spreminjanja elementov seznama uporabniških zgodb in s tem izboljšanje planiranja razvojnega projekta. Za izpolnitev cilja ponujamo modele procesov, izdelane v notaciji BPMN, katerih (izbrane) aktivnosti smo definirali kot uporabniške zgodbe in tako zagotovili pregled nad potrebami. Vendar pa dostikrat samo procesni pogled in modeli niso dovolj, da bi prikazali vse pomembne elemente in poslovni kontekst, ki ga mora pokrivati informacijska rešitev (Vara et al., 2008; Green et al., 1999). Zato predlagamo, da na podlagi modelov procesov pripravimo tudi uporabniške zgodbe.

3.1 Modeliranje poslovnih procesov z notacijo BPMN

Poslovni proces je zaporedje aktivnosti, ki privede do neke dodane vrednosti (Kovačič et al., 2005). V informacijskih sistemih zasledimo različne delitve procesov. V splošnem pa velja, da obstajajo tri vrste poslovnih procesov: podporni, temeljni in vodstveni. Ločijo se glede na namembnost, kot je npr. za temeljne procese značilna stvaritev dodane vrednosti za zunanjo stranko. Podporni procesi tečejo zato, da podpirajo temeljne procese, medtem ko imajo vodstveni procesi značilnost podpornega procesa z razliko v višji pomembnosti za poslovanje podjetja.

Mnoga podjetja nimajo formaliziranih (popisanih) poslovnih procesov (Kovačič et al., 2005), kar pa ne pomeni nujno, da ne poslujejo uspešno. Pomeni pa, da so lahko ključne informacije o načinu poslovanja ujete v glavi nekaj ključnih zaposlenih, katerih odhod iz podjetja je kritične narave. Ponekod zaposleni na istem delovnem mestu opravljajo neko aktivnost različno, saj je več načinov, kako opraviti nekaj zadovoljivo dobro. V takih primerih je treba pred informatizacijo poslovanja pridobiti znanje o poslovanju podjetja od različnih zaposlenih in formalno enovito zapisati poslovne procese v modelih. Modeli namreč predstavljajo sliko realnega sveta v specifičnem trenutku kot pogled specifične osebe za vnaprej določene namene modeliranja. Podjetja imajo tri osnovne razloge, zakaj modelirati poslovne procese (Kovačič et al., 2005):

- dokumentiranje: za doseganje zahtev standarda ISO9001 in drugih zahtev standardov ali regulatorja;
- analiziranje: za optimizacijo poslovanja;
- informatizacija: za podporo poslovanja z informacijsko tehnologijo.

Modeliranje za informatizacijo poslovanja je običajno del celostne prenove poslovnih procesov, ki predvideva štiri korake (Kovačič et al., 2005):

1. intervjuvanje zaposlenih;
2. modeliranje trenutnega poslovanja z modeli procesov »as-is«;
3. poenostavljanje poslovanja oz. optimiziranje (ne vključuje sprememb v informacijski tehnologiji);
4. modeliranje bodočega optimiziranega poslovanja z modeli procesov »to-be« (vključuje spremembe

v informacijski tehnologiji). Pred informatizacijo z informacijsko tehnologijo je tako smiselno preveriti, kje smo in kam želimo priti.

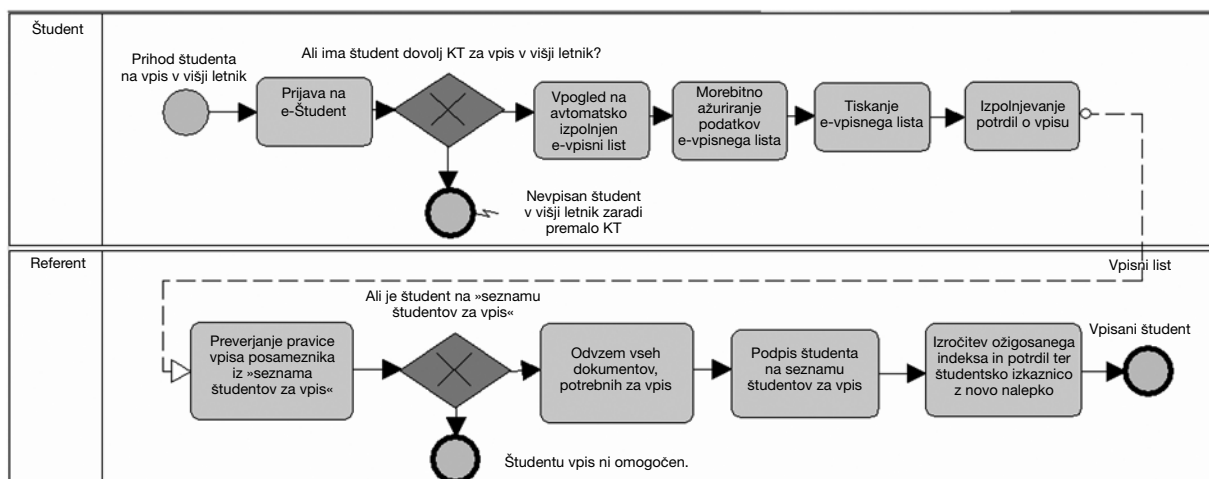
BPMN je notacija za grafično predstavitev poteka poslovnih procesov (Dijkman et al., 2008; Jurič et al., 2008; Lankhorst, 2005) z elementi, kot so dogodki, ki prožijo oz. končujejo aktivnosti, aktivnosti, ki jih izvajajo zaposleni, tokovi, ki povezujejo aktivnosti s ciljem dodajanja vrednosti stranki, in odločitve, ki spreminjajo tokove aktivnosti. Z zaporedjem aktivnosti nekega poslovanja predstavimo poslovni proces, katerega cilj je na podlagi vhodov ustvariti izhode, ki predstavljajo dodano vrednost za zunanjo ali notranjo stranko (Kovačič et al., 2005). Izvedba (angl. execution) poslovnih procesov mora podpirati tako avtomatizirane aktivnosti (natančneje opravila) kot aktivnosti, izvedene s strani zaposlenega (Sasa et al., 2008).

4 REŠITEV NA PRAKTIČNEM PRIMERU

Za ponazoritev ideje smo pripravili poenostavljen model obstoječega procesa (model »as-is«) vpisa študenta v višji letnik dodiplomskega študija (slika 1). Naročnik že ima informacijsko podporo za procese: prijava na e-Študent, vpogled na avtomatsko izpolnjeni e-vpisni list, morebitno ažuriranje podatkov e-vpisnega lista, tiskanje vpisnega lista. V novem informacijskem sistemu želi imeti dodatno podprte še te aktivnosti: izpolnjevanje potrdil o vpisu, preverjanje iz seznama študentov za vpis, ali ima študent pravico za vpis. Želi tudi, da se aktivnost podpis študenta na seznam študentov za vpis ne izvaja več ter da se ta sprememba zabeleži na modelu prenovljenega procesa (model »to-be«).

Za vsako aktivnost, ki jo bomo računalniško podprli, naredimo uporabniško zgodbo. Ime uporabniške zgodbe kreiramo po zgledu »Jaz kot (uporabniška vloga) želim (funkcija), da bi lahko (dodana vrednost funkcije)« iz treh podatkov, ki jih ponuja model BPMN: delovno mesto, navedeno v stezi (angl. lane) oz. bazenu (angl. pool) za »uporabniško vlogo«, ime aktivnosti za »funkcijo« in ime procesa za »dodano vrednost funkcije«.

Glede na želje naročnika iz zgornjega primera je nastal seznam uporabniških zgodb, ki ga prikazuje tabela 1.



Slika 1: »As-is« BPMN-model procesa vpisa študenta v višji letnik študija

Tabela 1: Seznam uporabniških zgodb iz procesa vpisa študenta v višji letnik

Podatki BPMN			Uporabniške zgodbe
Steza (angl. lane)	Ime aktivnosti	Ime procesa	
Študent	Prijava na e-Študent	Vpis dodiplomskega študenta v višji letnik	Jaz kot študent se želim prijaviti na e-Študent, da bi se lahko vpisal v višji letnik študija.
Študent	Vpogled na avtomatsko izpolnjeni e-vpisni list	Vpis dodiplomskega študenta v višji letnik	Jaz kot študent želim vpogledati na avtomatsko izpolnjeni e-vpisni list, da bi se lahko vpisal v višji letnik študija.
Študent	Morebitno ažuriranje podatkov e-vpisnega lista	Vpis dodiplomskega študenta v višji letnik	Jaz kot študent želim po potrebi ažurirati podatke na e-vpisnem listu, da bi se lahko vpisal v višji letnik študija.
Študent	Tiskanje e-vpisnega lista	Vpis dodiplomskega študenta v višji letnik	Jaz kot študent želim natisniti e-vpisni list, da bi se lahko vpisal v višji letnik študija.
Študent	Izpolnjevanje potrdil o vpisu	Vpis dodiplomskega študenta v višji letnik	Jaz kot študent želim avtomatsko izpolnjena e-potrdila o vpisu za tiskanje, da bi vpis v višji letnik študija potekal čim hitreje.
Referent	Preverjanje pravice vpisa posameznika iz seznama študentov za vpis	Vpis dodiplomskega študenta v višji letnik	Jaz kot referent želim preveriti pravico vpisa posameznika iz seznama študentov za vpis, da bi lahko v višji letnik vpisal samo tiste študente, ki nimajo blokade.

5 SKLEP

Ker odgovornost za sestavo seznama uporabniških zgodb leži na plečih naročnika (Cohn, 2004), je pomembno, da ta zna pravilno izraziti svoje zahteve. Pri agilnih metodah, kot je npr. Scrum, svoje uporabniške zahteve za novo programsko opremo naročnik izrazi z uporabniškimi zgodbami, na katerih temelji planiranje projekta. Da bi lahko čim bolj učinkovito planirali projekt agilnega razvoja, morajo biti uporabniške zgodbe definirane čim bolj natančno čim bolj na začetku projekta.

V članku smo izpostavili problem zagotavljanja globalnega pregleda nad zahtevami uporabnika na začetku projekta. Kot rešitev smo predstavili novo

tehniko definiranja začetnega seznama, ki je bolj strukturirana od tehnike delavnic, ki kombinira tehniki viharjenje možganov in prototipiranje. Slabosti zadnje smo namreč premagali z modeli poslovnih procesov. Modeli BPMN natančno ponazorijo zaporedje aktivnosti, ki privede do dodane vrednosti za notranjo oz. zunanjo stranko. S pomočjo omenjene notacije so procesi predstavljeni enolično. S tem so obvladovani prehodi med aktivnostmi zaposlenih pri doseganju dodane vrednosti. Ključno za uspeh razvojnega projekta je, da razvijalci poznajo in razumejo poslovanje, ki ga bodo računalniško podprli, saj bo stranka le tako dobila to, kar hoče. Iterativna komunikacija, ki jo promovirajo agilne meto-

dologije, pa je s pomočjo preglednih modelov poslovanja hitrejša in učinkovitejša. Zaradi predhodno narisanih modelov procesov se razvojniki na intervjujih z uporabniki lažje pogovarjajo, saj bolje poznajo celotno sliko poslovanja ter na podlagi te postavljajo relevantna vprašanja za konkretno uporabniško zgodbo.

6 LITERATURA

- [1] Ambler, S. (2002). Agile modeling: Effective practices for eXtreme Programming and the Unified Process. New York, John Wiley & Sons, Inc.
- [2] Cohn, M. (2004). User stories applied: for agile software development. Boston, Pearson Education, Inc.
- [3] Cohn, M. (2006). Agile Estimating and Planning, Pearson Education, Inc.
- [4] Deckter, B. et al. (2007). Wiki-Based Stakeholder Participation in Requirements Engineering. IEEE Software, 24(2), 28–35.
- [5] Dijkman, R. M. et al. (2008). Semantics and analysis of business process models in BPMN. Information and Software Technology, 50(12), 1281–1294.
- [6] Dyba, T. et al. (2008). Empirical studies of agile software development: A systematic review. Information and software technology, 50(9–10), 833–859.
- [7] Green, P. et al. (1999). An Ontological Analysis of Integrated Process Modelling. Paper presented at the Advanced Information Systems Engineering, Heidelberg.
- [8] Jurič, M. B. et al. (2008). Business process driven SOA using BPMN and BPEL. Birmingham, Packt Publishing.
- [9] Kovačič, A. et al. (2005). Management poslovnih procesov: Prenova in informatizacija poslovanja. Ljubljana, GV Založba.
- [10] Krisper, M. et al. (2003). Enotna metodologija razvoja informacijskih sistemov. Ljubljana, Vlada Republike Slovenije, Center Vlade RS za informatiko.
- [11] Lankhorst, M. et al. (2005). Enterprise architecture at work: Modelling, Communication, and Analysis. Berlin, Springer-Verlag Berlin Heidelberg.
- [12] Lee, C. et al. (2003). An Agile Approach to Capturing Requirements and Traceability. Paper presented at the Proceedings of the 2nd International Workshop on Traceability in Emerging Forms of Software Engineering.
- [13] Oz, E. (2009). Management Information Systems, Sixth Edition. Boston, Thomson course technology.
- [14] Papadopoulos, G. A. et al. (2008). Information Systems Development: Towards a Service Provision Society. Paper presented at the The 17th International conference on Information Systems Development, Cyprus.
- [15] Sasa, A. et al. (2008). Service-Oriented Framework for Human Task Support and Automation. IEEE transactions on industrial informatics, 4(4), 292–302.
- [16] Vara, J. L. et al. (2008). Improving Requirements Analysis through Business Process Modelling: A Participative Approach. Paper presented at the Business Information Systems Innsbruck.

Marina Trkman je leta 2007 pridobila naziv dipl. inž. računalništva in informatike, leta 2008 pa še uni. dipl. ekonomist. Istega leta se je zaposlila na Fakulteti za računalništvo in informatiko kot mlada raziskovalka. Raziskovalno se ukvarja s poslovno-informacijskimi arhitekturami, z metodologijami ugotavljanja uspešnosti informacijskih sistemov, z Web 2.0 in z metodologijami razvoja. Za svoje delo je leta 2009 prejela Trimovo raziskovalno nagrado ter leta 2010 nagrado AD FUTURE za trajnostni razvoj.

Marjan Krisper je izredni profesor na Fakulteti za računalništvo in informatiko Univerze v Ljubljani, kjer je tudi predstojnik katedre za informatiko in predstojnik laboratorija za informatiko. Njegova bibliografija obsega več kot dvesto strokovnih člankov in znanstvenih razprav. Vodi številne projekte razvoja informacijskih sistemov, elektronskega poslovanja in metodologij razvoja informacijskih sistemov v gospodarstvu, državni upravi in javnem sektorju. Je ustanovni član mednarodnega združenja za informacijske sisteme AIS (Association for Information Systems) in član izvršnega odbora Slovenskega društva Informatika.