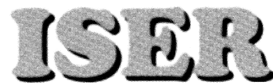
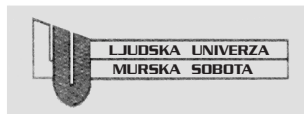


2017 ŠTEVILKA 1 < JAN. FEB. MAR. < LETNIK XXV < ISSN 1318-1882

01 UPORABNA INFORMATIKA

Izpitni centri ECDL

ECDL (European Computer Driving License), ki ga v Sloveniji imenujemo evropsko računalniško spričevalo, je standardni program usposabljanja uporabnikov, ki da zaposlenim potrebno znanje za delo s standardnimi računalniškimi programi na informatiziranem delovnem mestu, delodajalcem pa pomeni dokazilo o usposobljenosti. V Evropi je za uvajanje, usposabljanje in nadzor izvajanja ECDL pooblaščen ustanova ECDL Foundation, v Sloveniji pa je kot član CEPIS (Council of European Professional Informatics) to pravico pridobilo Slovensko društvo INFORMATIKA. V državah Evropske unije so pri uvajanju ECDL močno angažirane srednje in visoke šole, aktivni pa so tudi različni vladni resorji. Posebno pomembno je, da velja spričevalo v 148 državah, ki so vključene v program ECDL. Doslej je bilo v svetu izdanih že več kot 11,6 milijona indeksov, v Sloveniji več kot 17.000, in podeljenih več kot 11.000 spričeval. Za izpitne centre v Sloveniji je usposobljenih osem organizacij, katerih logotipe objavljamo.



U P O R A B N A I N F O R M A T I K A

2017 ŠTEVILKA 1 JAN/FEB/MAR LETNIK XXV ISSN 1318-1882

Znanstveni prispevki

Matjaž Roblek, Tomaž Kern

Pristop k prenovi in informacijski podpori razvojnega procesa v srednje velikem podjetju 3

Mateja Grobelnik, Jurij Jaklič

Znanja in sposobnosti podatkovnih znanstvenikov: pregled in analiza stanja v Sloveniji 17

Damijan Kozina, Davorin Kofjač, Andrej Škraba

Razvoj modela sistemske dinamike investiranja in donosnosti kapitala v organizaciji in realizacija v obliki spletne aplikacije 45

Strokovni prispevki

Boris Ovčjak, Tatjana Welzer Družovec, Gregor Polančič

Primerjava zmogljivosti večplatformsko razvitih mobilnih aplikacij 58

Informacije

Iz slovarja 73

Ustanovitelj in izdajatelj

Slovensko društvo INFORMATIKA
Litostrojska cesta 54, 1000 Ljubljana

Predstavniki

Niko Schlamberger

Odgovorni urednik

Jurij Jaklič

Uredniški odbor

Marko Bajec, Vesna Bosilj Vukšič, Sjaak Brinkkemper, Gregor Hauc, Jurij Jaklič, Andrej Kovačič, Jan von Knop, Jan Mendling, Miodrag Popović, Katarina Puc, Vladislav Rajkovič, Ivan Rozman, Pedro Simões Coelho, John Taylor, Mirko Vintar, Tatjana Welzer Družovec

Recenzenti

Alenka Baggia, Marko Bajec, Marko Bohanec, Renato Burazer, Janez Demšar, Dejan Dinevski, Saša Divjak, Nadja Dobnik, Jure Erjavec, Aleksandar Gavrič, Miro Gradišar, Aleš Groznik, Tanja Grublješič, Mojca Indihar Štemberger, Jurij Jaklič, Mirjana Kljajič Borštnar, Monika Klun, Andrej Kovačič, Nives Kreuh, Marjan Krisper, Robert Leskovar, Luka Pavlič, Aleš Popovič, Uroš Rajkovič, Vladislav Rajkovič, Živa Rant, Andrej Robida, Niko Schlamberger, Marina Trkman, Peter Trkman, Tomaž Turk, Mirko Vintar, Borut Werber, Boštjan Žvanut

Tehnična urednica

Mira Turk Škraba

Lektoriranje

Mira Turk Škraba (slov.)
Marvelingua (angl.)

Oblikovanje

KOFEIN DIZAJN, d. o. o.

Prelom in tisk

Boex DTP, d. o. o., Ljubljana

Naklada

600 izvodov

Naslov uredništva

Slovensko društvo INFORMATIKA
Uredništvo revije Uporabna informatika
Litostrojska cesta 54, 1000 Ljubljana
www.uporabna-informatika.si

Revija izhaja četrtletno. Cena posamezne številke je 20,00 EUR. Letna naročnina za podjetja 85,00 EUR, za vsak nadaljnji izvod 60,00 EUR, za posameznike 35,00 EUR, za študente in seniorje 15,00 EUR. V ceno je vključen DDV.

Revija Uporabna informatika je od številke 4/VII vključena v mednarodno bazo INSPEC.

Revija Uporabna informatika je pod zaporedno številko 666 vpisana v razvid medijev, ki ga vodi Ministrstvo za kulturo RS.

Revija Uporabna informatika je vključena v Digitalno knjižnico Slovenije (dLib.si).

© Slovensko društvo INFORMATIKA

Vabilo avtorjem

V reviji Uporabna informatika objavljamo kakovostne izvirne članke domačih in tujih avtorjev z najširšega področja informatike v poslovanju podjetij, javni upravi in zasebnem življenju na znanstveni, strokovni in informativni ravni; še posebno spodbujamo objavo interdisciplinarnih člankov. Zato vabimo avtorje, da prispevke, ki ustrezajo omenjenim usmeritvam, pošljejo uredništvu revije po elektronski pošti na naslov ui@drustvo-informatika.si.

Avtorje prosimo, da pri pripravi prispevka upoštevajo navodila, objavljena v nadaljevanju ter na naslovu <http://www.uporabna-informatika.si>.

Za kakovost prispevkov skrbi mednarodni uredniški odbor. Članki so anonimno recenzirani, o objavi pa na podlagi recenzij samostojno odloča uredniški odbor. Recenzenti lahko zahtevajo, da avtorji besedilo spremenijo v skladu s priporočili in da popravljeni članek ponovno prejmejo v pregled. Uredništvo pa lahko še pred recenzijo zavrne objavo prispevka, če njegova vsebina ne ustreza vsebinski usmeritvi revije ali če članek ne ustreza kriterijem za objavo v reviji.

Pred objavo članka mora avtor podpisati izjavo o avtorstvu, s katero potrjuje originalnost članka in dovoljuje prenos materialnih avtorskih pravic. Nenaročeni prispevkov ne vračamo in ne honoriramo. Avtorji prejmejo enoletno naročnino na revijo Uporabna informatika, ki vključuje avtorski izvod revije in še nadaljnje tri zaporedne številke.

S svojim prispevkom v reviji Uporabna informatika boste prispevali k širjenju znanja na področju informatike. Želimo si čim več prispevkov z raznoliko in zanimivo tematiko in se jih že vnaprej veselimo.

Uredništvo revije

Navodila avtorjem člankov

Članke objavljamo praviloma v slovenščini, članke tujih avtorjev pa v angleščini. Besedilo naj bo jezikovno skrbno pripravljeno. Priporočamo zmernost pri uporabi tujk in – kjer je mogoče – njihovo zamenjavo s slovenskimi izrazi. V pomoč pri iskanju slovenskih ustreznih priporočamo uporabo spletnega terminološkega slovarja Slovenskega društva Informatika Islovar (www.islovar.org).

Znanstveni članek naj obsega največ 40.000 znakov, strokovni članki do 30.000 znakov, obvestila in poročila pa do 8.000 znakov.

Članek naj bo praviloma predložen v urejevalniku besedil Word (*.doc ali *.docx) v enojnem razmaku, brez posebnih znakov ali poudarjenih črk. Za ločilom na koncu stavka napravite samo en prazen prostor, pri odstavkih ne uporabljajte zamika.

Naslovu članka naj sledi za vsakega avtorja polno ime, ustanova, v kateri je zaposlen, naslov in elektronski naslov. Sledi naj povzetek v slovenščini v obsegu 8 do 10 vrstic in seznam od 5 do 8 ključnih besed, ki najbolje opredeljujejo vsebinski okvir članka. Pred povzetkom v angleščini naj bo še angleški prevod naslova, prav tako pa naj bodo dodane ključne besede v angleščini. Obratno velja v primeru predložitve članka v angleščini. Razdelki naj bodo naslovljeni in oštevilčeni z arabskimi številkami.

Slike in tabele vključite v besedilo. Opremite jih z naslovom in oštevilčite z arabskimi številkami. Vsako sliko in tabelo razložite tudi v besedilu članka. Če v članku uporabljate slike ali tabele drugih avtorjev, navedite vir pod sliko oz. tabelo. Revijo tiskamo v črno-beli tehniki, zato barvne slike ali fotografije kot original niso primerne. Slik zaslonov ne objavljamo, razen če so nujno potrebne za razumevanje besedila. Slike, grafikoni, organizacijske sheme ipd. naj imajo belo podlago. Enačbe oštevilčite v oklepajih desno od enačbe.

V besedilu se sklicujte na navedeno literaturo skladno s pravili sistema APA navajanja bibliografskih referenc, najpogosteje torej v obliki (Novak & Kovač, 2008, str. 235). Na koncu članka navedite samo v članku uporabljeno literaturo in vire v enotnem seznamu po abecednem redu avtorjev, prav tako v skladu s pravili APA. Več o sistemu APA, katerega uporabo omogoča tudi urejevalnik besedil Word 2007, najdete na strani <http://owl.english.purdue.edu/owl/resource/560/01/>.

Članku dodajte kratek življenjepis vsakega avtorja v obsegu do 8 vrstic, v katerem poudarite predvsem strokovne dosežke.

▣ Pristop k prenovi in informacijski podpori razvojnega procesa v srednje velikem podjetju

Matjaž Roblek, Tomaž Kern, Univerza v Mariboru
matjaz.roblek@um.si; tomaz.kern@um.si

Izvleček

Podjetja, ki rastejo iz malega v srednje veliko, sprejemajo pomembne odločitve o spreminjanju organiziranosti procesov in o ustreznosti informacijske podpore. Povečanje števila različnih procesov in izjem v poslovanju zahteva prenovu poslovanja, sicer se pojavijo težave in napake pri izvajanju posla, npr. napačne interpretacije dogovora s kupcem, založenost ključnih podatkov, ponavljanje slabo izvedenih opravil, zamude, dolgotrajna komunikacija med izvajalci. Eden izmed načinov reševanja omenjene problematike, ki pri izvajalcih ni priljubljen, je vpeljava določene stopnje formalizacije v poslovanje, kar obsega tudi prepoznavanje in standardizacijo poslovnih procesov. Ugotavljamo, da je poseben izziv razviti in uporabiti ustrezen pristop povečevanja formalizacije znanjsko intenzivnih procesov v srednje velikih podjetjih, med katera uvrščamo proces razvoja novega proizvoda. Zato smo na podlagi dognanj obstoječih znanj oblikovali nov, prilagojen pristop k prenovi poslovnih procesov, ki je primeren za srednje velika podjetja in opredeljuje, kako prenoviti, organizirati in informacijsko podpreti razvojni proces, da se ohranijo poglobitvene prednosti zagonskega podjetja (npr. fleksibilnost, odzivnost, inovativnost), hkrati pa je vzpostavljena učinkovita izraba virov in omogočeno sočasno vodenje neprestanega razvoja večjega števila novih izdelkov. V prvem delu prispevka predstavljamo metodološka izhodišča, ki so bila podlaga za oblikovanje pristopa k prenovi procesa razvoja novega proizvoda v srednje velikem podjetju. V drugem delu prispevka pa predstavljamo rezultate validacije predlaganega pristopa na praktičnem primeru.

Ključne besede: prenova poslovnih procesov, razvoj novega proizvoda, informacijska podpora, srednje veliko podjetje.

Abstract

Approach to reengineering and information support of new product development process in a middle-sized company

When the company outgrows the size of a small business it must carry out significant changes in its organization and IT support. An increased number of process variants and exceptions in the business requires process reengineering, otherwise problems and mistakes occur, e.g. misinterpretation of the agreement with the buyer, misplace of key data, repetition of poor performed tasks, additional time spent on solving collusion between employees. The action of management that is not well accepted by employees is implementation of a certain degree of formalization into the business, including the identification and standardization of business processes. We found particularly challenging to develop and apply the appropriate approach to increase formalization of knowledge intensive processes in a medium-sized enterprises, especially the process of developing a new product (NPD). This article describes new, tailored approach to business process reengineering, which is suitable for medium-sized enterprises. It was developed on the basis of new findings and existing knowledge. It is possible to renew organization and information support of the renewed development process in order to preserve the main advantages of start-up companies (e.g. flexibility, responsiveness, and innovation), and simultaneously avoid explosion of operation costs in terms of rational use of resources when we manage continuous and parallel development of several new products. In first part of the article, we present methodology, which has been used as a basis for reengineering of NPD process in a medium-sized company. In the second part of the article we present practical results and validation of the proposed approach.

Keywords: business process reengineering, new product development, information systems, medium-sized enterprises.

1 UVOD

Rast podjetja prinaša pomembne spremembe v organizacijo in informacijsko podporo podjetja. Poveča se potreba po formalizaciji oz. potreba po vnaprej predpisani stopnji detajlnosti procesa. Poveča se potreba po tipiziranosti oz. potreba po razvrščanju procesov v različne kategorije, ki jih različno vodimo in informacijsko podpiramo. Poveča se stopnja standardiziranosti oz. potreba po izvajanju procesa po vnaprej predpisanem načinu, ki naj bi bil optimalen z zelo različnih vidikov, npr. z vidika učinkovite izrabe človeških virov, fizičnih sredstev, informacijske tehnologije, zagotavljanja kakovosti izdelka procesa. Procesov v lasti podjetja je vedno več in niso več samostojni, ampak so v smislu predhodnih in posledičnih procesnih povezav integrirani v procesno mrežo. To pomeni, da lahko napaka ali nekontrolirana izboljšava v enem procesu povzroči skrite težave v celotni mreži. Zato se vodenje in delovanje zagonskega podjetja bistveno razlikuje od poslovanja zrelega podjetja.

Ena izmed možnosti, da natančno opredelimo, kaj je majhno in kaj srednje veliko podjetje, je sklicevanje na Zakon o gospodarskih družbah (ZGD, 2016, člen 55). Z vidika organiziranosti podjetja je pomemben tisti del, v katerem zakon govori o številu zaposlenih. Zagonska podjetja so ob ustanovitvi običajno razvrščena med mikro podjetja, če imajo povprečno število delavcev v poslovnem letu manjše od deset. V primeru rasti preidejo zakonsko med majhna podjetja, ko imajo povprečno število delavcev v poslovnem letu večje od deset in manjše od petdeset.

Zanimivo je (Poličar, 2008), da se organiziranost podjetij, ki imajo deset ali manj zaposlenih, občutno ne razlikuje od tistih, ki imajo do petdeset zaposlenih. Organiziranost je bistveno drugačna pri velikih podjetjih z dvesto petdeset ali več zaposlenimi. Za mikro in majhna podjetja je dokazano (Kern, 1998), da za uspešno delovanje nimajo izrazite potrebe po formalizaciji poslovnih procesov: organizacijska struktura in procesi so nestrukturirani in dinamični, odvisni od pokritja zahtevanega znanja z razpoložljivim znanjem zaposlenih. Za velika podjetja je nasprotno dokazano (Stevenson, 2015), da mora obstajati večja formalizacija, tipizacija in standardizacija procesov z namenom doseganja uspešnega in hkrati učinkovitega poslovanja. Isto delovno mesto zaseda več različnih oseb, zato organizacijska struktura ni enaka kadrovski strukturi. Procesni so kompleksni, njihova medsebojna odvisnost pogosto ni transparentna.

Srednje velika podjetja (angl. Medium Enterprise, ME) se po številu zaposlenih nahajajo na preho-

du med malimi in velikimi, in zaposlujejo med 50 in 250 delavcev. Zanimalo nas je, ali imajo srednje velika podjetja potrebo po prepoznavanju in prenovi poslovnih procesov, kakšna je prava stopnja formalizacije njihovih procesov, kako fleksibilna naj bo informacijska podpora in predvsem ali je splošni metodološki pristop k prenovi procesov lahko enak kot pri velikih podjetjih.

V literaturi zasledimo trditve (Snidermann, 2012), da so prednosti srednje velikih podjetij pred mikro podjetji v lažjem spopadanju s poslovnimi izzivi, preprosteje pridobijo finančna sredstva za financiranje inovacij in lažje vzgajajo ter zadržujejo ključne talente v podjetju. Prednosti srednje velikih pred velikimi podjetji pa so v večji agilnosti ob spremembah v poslovnem okolju, so bližje kupcu in lažje ohranjajo izvorni navdih ustanovitve podjetja pri zaposlenih, npr. zaposleni se ob zahtevanih novi ali dodatni nalogi ne sprašuje, ali je to delo v njegovem opisu del in nalog, ampak opravi dodeljeno nalogo.

Iz pregleda raziskav smo dobili potrditev (Basu idr., 2003), da tudi srednje velika podjetja načrtno izvajajo projekte prenove procesov in njihove informacijske podpore. Ugotavljajo pa, da ta podjetja k prenovi pristopajo mnogo bolj racionalno kot velika podjetja. Strogo omejujejo finančne vire in potrebni čas za izvedbo sprememb v procesih, zato so te vpeljane hitreje in bolj preprosto. Prenova procesov je pogosto podrejena zmožnostim sodobnih informacijskih rešitev, najpogosteje tistim, ki so trenutno v branži, v kateri deluje podjetje, prepoznana kot najboljša praksa. Ugotovljeno je tudi, da srednje veliko podjetje začne prepoznavati z rastjo obsega poslovanja različne vrste procesov in njihovo delitev na vsebinsko sorodne sklope – podprocese.

V članku najprej predstavljamo pomen znanjsko intenzivnih procesov za srednje velika podjetja, nato prikažemo metodološke razlike med prenovami procesov za večja podjetja in prenovami, ki upoštevata posebnosti srednje velikih podjetij. Sledi vrednotenje uspešnosti oblikovanega pristopa na primeru srednje velikega podjetja in razprava o rezultatih ob upoštevanju trenutnih omejitev, ki bodo predmet nadaljnjih raziskav.

2 ZNANJSKO INTENZIVNI PROCESI IN NJIHOVA FORMALIZIRANOST

Znanjsko intenzivnost poslovnega procesa opišemo z dinamiko in inovativnostjo eksplicitnega, implicitnega in kulturnega znanja (Drakos, 2014):

- Eksplicitno znanje procesa je zapisano na dokumentih, fotografijah, zvočnih zapisih, risbah, modelih, tehnoloških postopkih. Sem spadajo tudi poslovna pravila, organizacijska navodila in standardne procedure, ki poosebljajo način, kako podjetje deluje in koordinira delo. Eksplicitno znanje lahko neprestano zapisujemo in prenašamo med zaposlene.
- Implicitno ali skrito znanje procesa je tisto, zaradi katerega je določeni posameznik pri opravljanju svojega dela boljši od drugega kljub enakovrednemu dostopu do eksplicitnega znanja in ob predpostavki enakih izhodiščnih sposobnosti posameznikov. Izkazujejo ga dobra presoja, spretnost, strokovnost, kreativnost in skrito znanje, prepoznamo ga v vrhunski izvedbi določene naloge. Implicitno znanje je težko zapisati, opredeliti s pravili ali enačbami, in se ga je težko naučiti. Gre za kombinacijo izkušenj in celovitega poznavanja področja ob hkratnem odličnem poznavanju najmanjše podrobnosti za izvedbo naloge (mojstrstvo).
- Kulturno znanje procesa sestavljajo neformalno oblikovana stališča, prepričanja, odnosi, norme, vrednote, rituali, simboli (logotipi) in kultne osebnosti, kot so zgodbe ustanoviteljev podjetja. To vpliva na oblikovanje odnosa in obnašanja zaposlenih do podjetja. Kulturno znanje se prenaša s sodelovanjem med zaposlenimi, preko opazovanja načina izvedbe dela cenjenih delavcev mentorjev in lahko povzroča neformalen način izvedbe nalog mimo predpisanih pravil.

V mikro ali majhnem podjetju znanjsko intenzivnost obvladujeta implicitno in kulturno znanje. Če bi se prevladovanje teh dveh vrst znanja nadaljevalo tudi v srednje velikem podjetju, bi zaposleni ob povečanem obsegu dela v znanjsko intenzivnih procesih večino časa:

- iskali nove informacije in poizvedovali za informacijami, ki niso zapisane,
- opravljali nepotrebne poti in neformalno sestankovali, ker ne vemo, ali so vsi seznanjeni s problematiko, če to ni zapisano,
- se redundantno in nasprotujoče odločali, npr. ponavljali že dane ukaze ali izdajali različna navodila za isti primer, ker odzivi niso zapisani.

Pomanjkanje eksplicitnega znanja pomeni več napak v procesih, več napačnih odločitev, po nepotrebem povečano umsko obremenjenost zaposlenih

in nepotrebne povratne zanke pri izvedbi dela. Ko reševanje teh težav pomembno zmanjša učinkovitost dela zaposlenih, kar se dogodi v primeru prehoda podjetja v srednje veliko podjetje, je treba »prenoviti« proces tako, da določimo eksplicitno znanje (kaj), ki ga je treba obvladovati, oz. moramo vzpostaviti določeno stopnjo formaliziranosti in standardiziranosti procesa (kje, kdo, zakaj). Težava je v tem, da zaposleni težko dojamajo, da jim bo formaliziranost vrnila izgubljeno učinkovitost, še posebno zaposleni na znanjsko intenzivni procesih. Formalizacijo in standardizacijo, ki jo v majhnem podjetju ni, prepoznajo kot še dodatno poslabšanje učinkovitosti oz. kot povečano »birokratizacijo« dela. Ta odziv lahko omilimo s primerno informacijsko rešitvijo, ki mora prispevati k preprostosti avtomatizacije zapisovanja eksplicitnega znanja in dostopanja do njega.

Na podlagi raziskanega lahko ugotovimo, da nadaljnji uspešen razvoj srednje velikih podjetij ni mogoč brez začetka uvajanja formalizacije in informacijske podpore v znanjsko intenzivnih procesih.

2.1 Proces razvoja novega izdelka

V proizvodnih podjetjih je predstavnik skupine znanjsko intenzivnih procesov, pri katerih pričakujemo od zaposlenih izrazito eksplorativnost, proces razvoja novega izdelka (RNP). V majhnem podjetju v procesu razvoja novega izdelka dominirata implicitno in kulturno znanje, ker je proces v večini primerov oblikoval ustanovitelj podjetja (Poličar, 2008). Z večanjem obsega dela v primeru srednje velikega podjetja in s prisotnostjo podjetja v globalnih oskrbovalnih verigah, ki so podvržena odličnosti preko upoštevanja določenih panožnih standardov, se povečuje potreba po obvladovanju eksplicitnega znanja. Postavlja se vprašanje, kako doseči »ravno pravo« stopnjo formalizacije procesa razvoja novega izdelka v srednje velikem podjetju, da ohranimo ustanovitveni duh, zagon in sodelovanje zaposlenih, kot je bilo pri izvedbi prve ponovitve razvoja izdelka, in zagotovimo učinkovito ponovljivo razvijanje novih izdelkov, ne da bi eksploatacijski procesi (proizvodnja, nabava, prodaja, računovodstvo ipd.), pri katerih dominirajo pravila, ponovljivost, rutina, omejevali izvajanje procesa razvoja novega izdelka.

Srednje veliko podjetje ima v primerjavi z mikro ali malim podjetjem več hkrati aktivnih razvojnih timov, od katerih zahtevajo stalne inovacije, veliko novih izdelkov, stroge omejitve pri doseganju zahte-

vanih rokov plasiranja novega izdelka na trg, strog nadzor proračuna razvojnega dela in stroškov razvojnih projektov. Drži tudi, da se pri srednje velikih podjetjih stopnja novosti v novih izdelkih od projekta do projekta zelo razlikuje in da obstajajo različne kategorije »novih« izdelkov (FactBook.org, 2015), ki zopet vplivajo na vzpostavitev večje ali manjše stopnje formalnosti v procesu razvoja novega izdelka:

- Radikalno novih izdelkov je malo, a so ključni za trajno rast prihodkov podjetja; izdelki pomenijo nove sistemske rešitve, so prožilci novih platform. V razvoj je vključenih zelo veliko zaposlenih (tudi zunanjih) zelo različnih poklicnih profilov.
- Veliko je izpeljank izdelkov, pri katerih kupci zahtevajo majhne spremembe v obliki ali funkciji izdelka. Prožilec spremembe na izdelku je lahko tudi posledica optimizacije lastnega proizvodnega procesa. To so različice znanih izdelkov ali izboljšave obstoječih, dovoljujejo hiter razvoj, v razvoj je vključenih malo zaposlenih ozkega profila. Ti izdelki zagotavljajo kratkoročen denarni tok in ohranjanje tržnih deležev obstoječih izdelkov.
- Industrializacija izdelka oz. razvoj procesa izdelave že razvitega izdelka kupca je pogosto dominanten pristop v srednje velikih podjetjih. Pri tem podjetje uporabi lastno »razvojno znanje« za razvoj procesa izdelave in povezanih gradnikov (orodij, avtomatizacije transporta, podajalnikov, strojev, robotov). Če srednje velika podjetja uspejo dobro razviti proces, tudi prevzamejo proizvodnjo izdelkov, kar pomeni dodaten »razvoj« vseh kontrolnih, oskrbovalnih, logističnih, transport-

nih in distribucijskih elementov izdelave izdelka. Tudi te »nove izdelke« srednje veliko podjetje uvršča v proces razvoja novega izdelka.

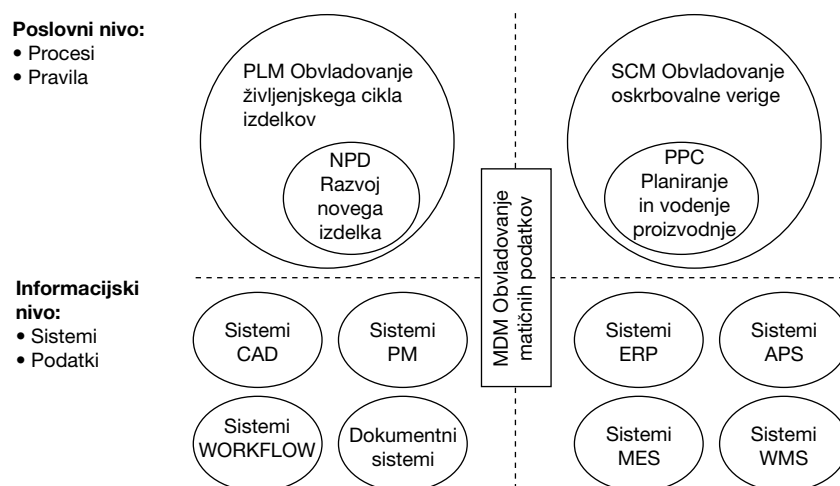
Postavlja se novo vprašanje, kako v srednje velikem podjetju vzpostaviti učinkovito formalizacijo procesa razvoja novega izdelka, ki bo omogočala obvladovanje tako raznovrstnih razvojnih projektov razvoja novega izdelka.

2.2 Informatiziranost razvojnega procesa

Za strukturni pregled informatiziranosti procesa razvoja novega izdelka smo uporabili bazo posnetih razvojnih procesov slovenskih proizvodnih podjetij, ki se ukvarjajo z razvojem (Roblek, Kern, 1998–2016). V raziskavi je bilo zajetih osem podjetij, ki so ustrezala kriteriju mikro ali majhnega podjetja, sedem podjetij, ki so ustrezala kriteriju srednje velikega podjetja, in šest velikih podjetij.

V analiziranih majhnih in mikro podjetjih sestavljajo informacijski sistem za podporo procesa razvoja novega izdelka orodja za računalniško podprto načrtovanje (angl. Computer Added Design, CAD), s katerimi je podprto ekspertno delo znanjskih delavcev razvijalcev. Poleg tega uporabljajo standardne informacijske rešitve za podporo komuniciranju (e-pošta, spletni portali naročnikov) in računalniške preglednice (MS Excel).

V srednje velikem podjetju je informacijska podpora procesa razvoja novega izdelka že precej bolj kompleksna (slika 1). Vsa analizirana srednje velika podjetja na strateškem nivoju prepoznajo dve področji procesov: obvladovanje življenjskega cikla



Slika 1: **Kompleksnost informacijskega sistema za podporo procesu razvoja novega izdelka v srednje velikem podjetju**

izdelka (angl. Product Lifecycle Management, PLM) ter obvladovanje oskrbovalnih verig (angl. Supply Chain Management, SCM). Na operativnem nivoju prepoznajo proces razvoja novega izdelka (angl. New Product Development, NPD) ter proces planiranja in vodenja proizvodnje (angl. Production Planning and Control, PPC). Med njima zahtevajo učinkovito procesno in podatkovno povezavo.

Na informacijskem nivoju (sistemi in podatki) so v srednje velikem podjetju poleg orodij CAD prisotni projektni informacijski sistemi (angl. Project Management, PM; Project Portfolio Management, PPM), informacijski sistemi za vodenje in obvladovanje poteka tehnične dokumentacije (npr. Windchill) in vodenje druge razvojne dokumentacije (angl. Document Management Systems, DMS).

Vsa analizirana srednje velika podjetja izkazujejo veliko večjo potrebo po ustreznem obvladovanju matičnih podatkov kot mala podjetja (angl. Master Data Management, MDM) in po vzpostavitvi zanesljive podatkovne povezave med razvojnim in produkcijskim delom celovitega informacijskega sistema podjetja, ki vključuje poslovni informacijski sistem (angl. Enterprise Resource Planning, ERP), napredne sisteme za planiranje in vodenje proizvodnje (angl. Advance Planning and Scheduling, APS; Manufacturing Execution System, MES) in sisteme za vodenje skladišča (angl. Warehouse Management System, WMS).

Ključni poudarek uporabnikov razvojnih informacijskih sistemov v srednje velikem podjetju je, da se zaradi uporabe različnih informacijskih sistemov in povezav med njimi ne smejo oslabiti sposobnosti procesa razvoja novega izdelka in z njim povezanih procesov.

2.3 Formaliziranost organizacije razvojnega procesa

Čeprav v raziskavi povezujemo velikost podjetja s številom zaposlenih in stopnjo formaliziranosti organizacije procesa razvoja novega izdelka, je dokazano (Horie in Ikawa, 2014), da je pomembno upoštevati tudi, ali so podjetja nova (navadno manjša) ali obstoječa (navadno večja, kar pa ni pravilo). Avtorji dokažejo, da je vztrajanje na obstoječem procesu razvoja novega izdelka neugoden dejavnik za podjetje, ko to raste iz malega v srednje veliko.

Študija (Woschke idr., 2016) ugotavlja, da srednje velika podjetja podjetja začenjajo meriti »izgube« v procesu razvoja novega izdelka, kot so čakalni časi, podvajanje dela, materialni stroški, in da rešitve iščejo tudi v formalizaciji in standardizaciji tega procesa.

Študija, opravljena na področju Irske (Robbins, 2014), dokazuje, da je dobra učinkovitost procesa razvoja novega izdelka v majhnih in srednje velikih podjetjih v splošnem odvisna od dobro opredeljene strategije inovativnosti in dobro formaliziranega procesa. Ta študija ugotavlja, da dve tretjini inovacijsko aktivnih srednje velikih podjetij na Irskem nima inovacijske strategije, še manj pa jih ima formaliziran proces razvoja novega izdelka. Posebno je zanimivo, da podjetja, ki imajo formalizirano organizacijo tega procesa, izkazujejo višje donose in dosegajo bolj radikalne inovacije.

V kontekstu raziskave je pomembna tudi študija Leitholda idr. (2016), opravljena v Nemčiji, ki ugotavlja, kako morajo biti formalno organizirani procesi razvoja novega izdelka v srednje velikih podjetjih. Avtorji povezujejo standardiziranost procesa razvoja novega izdelka z uspešnostjo podjetja in dokažejo, da je ključna lastnost dobro organiziranega procesa razvoja novega izdelka v srednje velikem podjetju sposobnost, da ga je mogoče hitro prilagajati. V smislu formalizacije mora biti »uzakonjena« prilagodljivost strukture procesa razvoja novega izdelka.

Iz navedenega lahko ugotovimo, da srednje velika podjetja morajo formalizirati proces razvoja novega izdelka. Iz omenjenih in drugih raziskav (Poetz in Schreier, 2012; Sosa idr., 2015; Woschke idr., 2016; Rajasekaran idr., 2016) je tudi razvidno, kako formaliziran mora biti proces razvoja novega izdelka, da bi bil učinkovit.

Glede na težave srednje velikega podjetja pri zupanju v pozitiven učinek formaliziranosti procesa razvoja novega izdelka nismo zasledili raziskav, ki bi predlagale, kako naj se izvede prehod iz neformaliziranega v formalizirano stanje. Mala podjetja pri prehodu v srednje velika v večini primerov nimajo vzpostavljene vsaj temeljne formalizacije procesov, zato ne moremo uporabiti klasične metode prenove poslovnih procesov za velika podjetja, v katerih so obstoječi procesi vnaprej znani, zapisani in obvladovani ter v katerih zaposleni razumejo njihovo korist pri dvigovanju učinkovitosti procesov in načrtovanju njihove informatizacije.

2.4 Spreminjanje razvojnega procesa

Kot izhodiščno bazo metodologij za oblikovanje pristopa k prenovi procesa razvoja novega izdelka v srednje velikih podjetjih smo uporabili nabor po Harmonu (Harmon, 2015):

- metode reševanja splošnih procesnih problemov; v grobem jih razvrščamo v dve podskupini:
 - »metode od zgoraj navzdol« oz. metode, ki celovito obravnavajo reševanje problemov v procesih: metoda Rummler-Brache, metoda prenove poslovnih procesov (angl. Business Process Re-engineering, BPR), upravljanje poslovnih procesov (angl. Business Process Management, BPM),
 - »metode od spodaj navzgor« oz. metode, ki procesno problematiko obravnavajo in rešujejo lokalno: 6 sigma (angl. Six sigma), vitko razmišljanje (angl. Lean Thinking), metodologija Toyotinega proizvodnega sistema (angl. Toyota Production System, TPS);
- metode za reševanje specifičnih problemov; vezane so na reševanje problemov na specifičnem strokovnem področju podjetja:
 - metode za izboljševanje strukturiranih procesov, npr. stroškovna analiza na podlagi aktivnosti (angl. Activity Based Costing, ABC),
 - metode za izboljševanje proizvodnih in oskrbovalnih procesov, npr. teorija omejitev (angl. Theory of Constraints), referenčni modeli za izboljšanje oskrbovalne verige (angl. Supply Chain Operations Reference, SCOR),
 - metode za izboljševanje procesov razvoja novega izdelka, npr. teorija organizacije izvajanja aktivnosti, pri kateri zahtevamo inovacije (angl. Theory of the resolution of invention-related tasks, TRIZ).

Pri pripravi primerne pristopa k prenovi procesa razvoja novega izdelka v srednje velikem podjetju smo upoštevali te posebnosti: poslovna fleksibilnost srednje velikega podjetja zahteva veliko dinamiko in hitrost pri uvajanju sprememb v proces razvoja novega izdelka, proces razvoja novega izdelka v srednje velikem podjetju je zelo znanjsko intenziven in nima standardnega izhoda (ki pa mora biti v skladu s standardi oskrbovalne verige), proces razvoja novega izdelka je odvisen od dobre podpore povezanih procesov, in preprosta uporaba razvojnih informacijskih sistemov je ključna za vzpostavitev formalizacije procesa razvoja novega izdelka, ki jo bodo sprejeli izvajalci.

Postavlja se vprašanje, katero metodologijo izbrati za spreminjanje procesa razvoja novega izdelka in jo nadgraditi za posamezen primer, da bomo ohranili vse dobre lastnosti procesa razvoja novega izdelka v srednje velikem podjetju in odpravili vse slabosti, ki so posledica rasti podjetja.

3 METODOLOGIJA PRENOVE PROCESA RAZVOJA NOVEGA IZDELKA V SREDNJE VELIKEM PODJETJU

Pred začetkom prenove smo oblikovali tri značilne skupine procesov, ki si sledijo po pomembnosti upravljanja procesov (Kern, 1998):

- procese, ki zagotavljajo ključno konkurenčno prednost srednje velikega podjetja v poslovanju in so v lasti srednje velikega podjetja, npr. proces razvoja novega izdelka; imenujemo jih tudi identifikacijski procesi;
- procese, ki morajo prejšnji skupini procesov zagotoviti hitro podporo in odziv ter so zaradi tega prav tako organizacijsko vključeni v prenovno procesov srednje velikega podjetja, npr. strateško prodajanje, strateško nabavljanje, upravljanje kadrov, zagotavljanje kakovosti; imenujemo jih tudi temeljni procesi;
- procese, ki jih srednje veliko podjetje zaradi ekonomičnosti poslovanja kot storitve najpogosteje najema v poslovnem okolju, npr. pravni procesi, vzdrževanje informacijske podpore; imenujemo jih tudi podporni procesi.

Drugo in tretjo skupino procesov smo v skladu s priporočili literature prenovili z uporabo informacijskih rešitev najboljše prakse. S tem smo dosegli hitro implementacijo preverjeno koristnih sprememb v poslovanje.

Za prvo skupino procesov, ki so praviloma znanjsko intenzivni in vsebujejo »znanje« podjetja, ki srednje veliko podjetje najbolj varuje pred konkurenčnimi podjetji, je bilo treba prilagoditi metodo prenove. To so »najstarejši procesi« podjetja, s katerimi se zaposleni najbolj identificirajo. Imenujemo jih identifikacijski procesi in s seboj v rast podjetja prinašajo prakso poslovanja iz časa ustanovitve podjetja. Izživ za prenavljanje pomenijo zato, ker so omogočili uspeh podjetja, torej so kot »ustrezno« prepoznani v poslovnem sistemu in zunaj njega, npr. pri razvojnih partnerjih. Oboji z rastjo podjetja spoznavajo (zlasti menedžment srednje velikega podjetja), da jih je treba prilagoditi novi, mnogo večji frekvenci ponavljanja »razvoja« in kupcem z zahtevnejšimi pogoji (npr. raznimi certifikati), pri čemer izvajalci v procesu kljub dokazanemu pozitivnim učinkom ne verjamejo, da sta rešitev formaliziranost in standardiziranost.

Postavlja se izziv, kako metodološko pristopiti k prenovi znanjsko intenzivnih procesov v srednje velikem podjetju (prva skupina) ob predpostavki, da so

spremembe v procesih druge in tretje skupine podrejene procesom prve skupine.

3.1 Izdelava predštudije

Pri odločanju o stopnji formalizacije procesa razvoja novega izdelka oz. vzpostavitve zajemanja eksplicitnega znanja v srednje velikem podjetju smo najprej v okviru predštudije ugotavljali, koliko so izvajalci v procesu že zreli za tak prehod. Pomagali smo si s prilagojeno matriko (Drakos, 2014) na sliki 2, ki prikazuje, kako zaposleni v procesu razvoja novega izdelka interpretirajo pomembnost evidentiranja, prenašanja in ponovne uporabe različnih vrst znanja pri prehodu iz malega v srednje veliko podjetje.

Z lastniki ali vodstvom srednje velikega podjetja oz. naročniki uvajanja formalizacije v procese smo pred prenavo procesa razvoja novega izdelka preverili, kakšen je trenutni pogled zaposlenih na pretočnost oz. prenos znanja med zaposlenimi v tem procesu. Tako lahko ugotovimo, kolikšen vpliv ima »organiziranost« procesa razvoja novega izdelka iz časa malega podjetja na trenutno delo zaposlenih v razvoju in ali so zreli na vzpostavitev formalizacije procesa. Na sliki 2 so rezultati raziskave SME (Drakos, 2014) in razlik med njimi.

- Raziskava trdi, da je za zaposlene v procesu razvoja novega izdelka majhnega podjetja najpomembnejše implicitno strokovno znanje, ki omogoča razvoj novega izdelka. Zaradi nezaupanja, npr. bojazen pred krajo implicitnega znanja, ideje, majhen tim (do deset članov), deluje tako, da vsak član tima prispeva svoje »specialno« znanje. Pri tem ne čutijo potrebe po evidentiranju, niti po prenosu tega znanja drugim članom, niti po po-

novni uporabi. Cilj tima je plasirati na trg prvi nov izdelek brez razmisleka, ali bo kdaj prišlo do ponovitve razvojnega procesa in bi potrebovali bazo znanja. Tim pogosto deluje usklajeno po normah vodje tima (navadno lastnika), zato je pomemben prenos kulturnega znanja med sodelujočimi.

- V srednje velikem podjetju, ki v skrajnem primeru po ZGD zaposluje že blizu 250 ljudi, sodeluje v procesu razvoja novega izdelka 20 do 30 in več zaposlenih. Ker so razvijalci medsebojno odvisni od informacij, postaneta pomembna prenos in hranjenje eksplicitnega znanja (baza znanja) za učinkovito in kakovostno ponavljanje procesov razvoja novega izdelka.

Če imamo opravka s srednje velikim podjetjem, v katerem z vidika formaliziranosti procesa razvoja novega izdelka pri zaposlenih še vedno prevladuje način razmišljanja malega podjetja, potem moramo ta razkorak v miselnosti pomembno upoštevati pri spreminjanju formaliziranosti procesa. Ta organizacijski vidik smo preučili v okviru predštudije in je bil podlaga za oblikovanje pristopa k prenovi procesa razvoja novega izdelka v obravnavanem srednje velikem podjetju.

K spremembi organizacije procesov v obravnavi srednje velikega podjetja smo pristopili na projektni način (Petrovič idr., 2010). Vodstvo se je zavedalo omenjenega razkoraka v miselnosti izvajalcev (opisano na sliki 2), kar smo ugotovili s predštudijo. Predštudija je bila opravljena v obliki projektne naloge, pri kateri smo s pomočjo vodenih intervjujev s ključnimi udeleženci v procesu razvoja novega izdelka in vodilnimi delavci v podjetju izdelali grobe modele procesov (angl. Function Tree, FT; Value

Pomembnost obvladovanja znanja v procesu RNP (malo podjetje)		Znanjski tokovi		
		Nastanek	Prenos	Večkratna raba
Vrste znanja	Eksplicitno	•	•	•
	Implicitno	•	•	•
	Kulturno	•	•	•

Pomembnost obvladovanja znanja v procesu RNP (ME)		Znanjski tokovi		
		Nastanek	Prenos	Večkratna raba
Vrste znanja	Eksplicitno	•	•	•
	Implicitno	•	•	•
	Kulturno	•	•	•

Legenda:
 • Nepomembno
 • Pomembno
 • Zelo pomembno

Slika 2: **Strinjanje izvajalcev srednje velikega podjetja (ME) po obvladovanje znanja (eksplicitnega, implicitnega in kulturnega) v procesu razvoja novega izdelka (RNP)**

Added Chain, VAC) in jih ob tem spraševali, v kateri smeri vidijo rešitev težav v procesu razvoja novega izdelka. V predštudiji je bil ugotovljen velik razkorak pri razumevanju koristi, potreb in načina formalizacije oz. zajema eksplisitnega znanja, njegovega prenosa in ponovne uporabe tako eksterno (zahteve strank, pričakovanja dobaviteljev) kot interno (med lastniki, menedžmentom podjetja in izvajalci). Rezultat predštudije je bila pripravljena projektna definicija projekta prenove.

3.2 Popis in analiza procesov

Zaradi zahteve srednje velikega podjetja po hitri in pomembni izboljšavi procesa razvoja novega izdelka je bila za spreminjanje procesov izbrana metoda BPR (Kern, 2003). Podjetje pred projektom ni prepoznavalo procesov (zgolj z grobim zapisom v ISO-standardih), zato je bilo treba najprej vzpostaviti menedžment poslovnih procesov. V prvem delu projekta spreminjanja organizacije procesov sta bili izvedeni dve fazi:

- posnetek stanja; identifikacija procesov in lastnikov procesov, podroben opis procesov, popis dokumentnega in informacijskega sistema, popis obstoječe organizacijske strukture in kadrov, vzpostavitev mreže procesov; posnetek je vključeval izdelavo organizacijskih in kadrovskih modelov, procesnih modelov v tehniki razširjene dogodkovne procesne verige (angl. Extended Event Process Chain, eEPC) v enot-

nem procesnem repozitoriju, ki je bil zgrajen v orodju ARIS;

- analiza; vključevala je analizo pomembnosti procesov (za kupca, za podjetje), strokovno in strukturno analizo konkurenčnega stanja procesov (interno in eksterno problematiko, izzive in priložnosti za izboljšave) ter izdelavo prioritete liste procesov za spreminjanje. V odvisnosti od zmožnosti virov podjetja in podrejenosti procesov so bili oblikovani sklopi povezanih procesov, ki se bodo skupaj prenavljali glede na radikalnost potrebnih sprememb. Izbrani so bili sklopi procesov, ki se bodo spreminjali po metodi BPR (radikalno), in sklopi procesov, ki se bodo spreminjali po majhnih, postopnih korakih. Nekateri od teh so se spreminjali že med analizo v obliki hitrih ukrepov (angl. Quick Wins), če je bilo prek mreže procesov ugotovljeno, da na njih ne bo vplivalo spreminjanje procesov z zahtevo po radikalni prenovi.

Slika 3 prikazuje sedem faktorjev, po katerih smo ocenjevali in definirali ključne procese za prenovu, od katerih ima vsak faktor več podrejenih kriterijev (Kokalj, 2012). Proces razvoja novega izdelka je bil po večkriterijski analizi izbran kot ključni proces srednje velikega podjetja, ki ga je treba hitro in radikalno spremeniti, da bo čim hitreje zagotavljal zahtevano eksplisitno znanje vsem deležnikom procesa (internim in eksternim).

A1	faktor	B	C	D	E	F	G
	faktor	vrednost	%variance	utež	min	Privzeto	max
2	1 Organiziranost poslovnega sistema	3,635	24,237	31,45%	0%	0%	100,00%
3	2 Kompleksnost poslovnih procesov	1,885	12,565	16,30%	0%	0%	100,00%
4	3 Dokumentiranost opravljenega dela	1,549	10,328	13,40%	0%	0%	100,00%
5	4 Obsežnost poslovnih procesov	1,395	9,303	12,07%	0%	0%	100,00%
6	5 Medsebojna povezanost procesov	1,296	8,643	11,21%	0%	0%	100,00%
7	6 Podprtost z informacijsko tehnologijo	0,943	6,286	8,16%	0%	0%	100,00%
8	7 Ustvarjanje dodane vrednosti	0,856	5,706	7,41%	0%	100,00%	100,00%
9			77,068	100%			

Slika 3: Faktorji za razvrstitev procesa med ključne za prenovu

Stolpec % variance predstavlja porazdelitev variabilnosti osnovnih spremenljivk. Navedeni so faktorji, ki skupaj pojasnjujejo 77,068 % variance (nastavljeno glede na raziskavo Urh, 2011). Privzete uteži posameznih faktorjev so preračunane tako, da 77,068 predstavlja 100 %. Privzete vrednosti uteži je mogoče spreminjati glede na branžo podjetja z drsniki. Vrednost faktorja predstavljajo vrednosti podrejenih izmerjenih spremenljivk analiziranega procesa, pomnožene z utežjo. V konkretnem primeru je proces razvoja novega izdelka dobil skupno oceno 11,559 (vsota vrednosti vseh faktorjev). Višja kot je skupna ocena glede na druge procese, bolj je proces ključen za prenavo glede na ostale.

3.3 Prenova procesa razvoja novega izdelka

V prvem koraku prenovne smo določili značilnosti procesa razvoja novega izdelka glede na strukturiranost procesa/izhoda procesa. Ugotovili smo, da gre metodološko za delno strukturiran »case-like« proces, ki ima lastnosti:

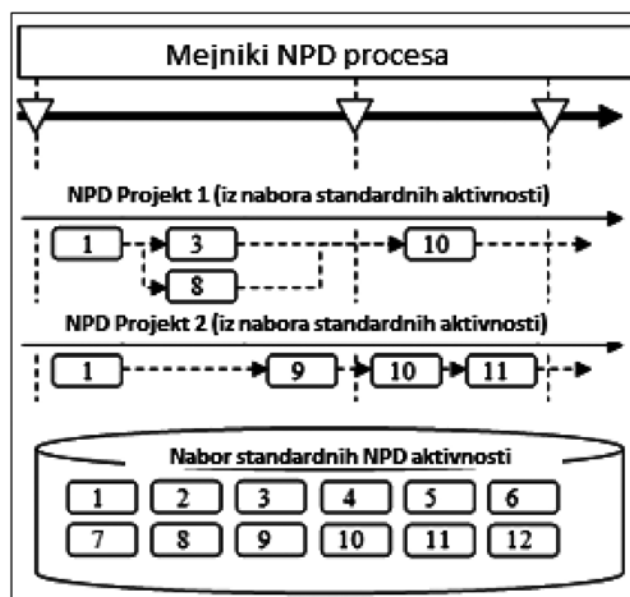
- od vsebine razvojnega naročila je odvisen nabor standardnih aktivnosti procesa,
- večina aktivnosti v procesu je znanjsko intenzivnih (so dominantne v procesu),
- pomembna je odvisnost procesa od delno ali povsem nestrukturiranih informacij,
- način dela je predpisan z ohlapnimi pravili, ki predstavljajo usmeritve za izvajanje,
- proces ima dolge pretočne čase in nepredvidljive povratne zanke (stopnja novosti),
- izdelek procesa razvoja novega izdelka je sestavljen iz nestandardnih elementov, odvisnih od primera (nastanejo pretežno v nestrukturiranem delu procesa), in standardnih odvisnih elementov (izdelki strukturiranega dela procesa in povezanih podpornih poslovnih procesov),
- v eni ponovitvi procesa nastopa mnogo različnih znanjskih izvajalcev.

To pomeni, da ima proces razvoja novega izdelka vse lastnosti za vodenje na projektni način z nekaj »posebnostmi«: baza razpoložljivih aktivnosti projekta je fiksna, uporabljeni resursi so vedno enaki, nekateri deli procesa so strukturirani in vedno enaki. Zato smo smernice prenovne procesa razvoja novega izdelka najprej začrtali z uporabo sistema mejnikov (slika 4). Ta koncept nam omogoča, da zelo raznovrstne zahteve po razvoju, ki zahtevajo različen nabor razvojnih aktivnosti, spremljamo in

vodimo na projektni način prek standardnega nabora »nadzornih točk« (mejniov), ki nam s procesnega vidika omogočajo medsebojno primerjavo in nadzor nad uspešnostjo in učinkovitostjo ponovitve procesa razvoja novega izdelka. Vsak mejnik ima določeno odgovorno osebo (skrbnika), nabor dokumentov in njihove statuse, pravila in kriterije, s katerimi dovoljuje nadaljevanje izvajanja procesa razvoja novega izdelka. Pri tem pa sistem mejnikov ne sme ovirati sočasnosti razvojnih aktivnosti in projektov.

Posebno pozornost potrebujejo mejniki, ki predstavljajo »povezave« oz. »vrata« v proces razvoja novega izdelka ali iz njega v podporne procese, kot so nabava materialov, obračun plač, likvidacija računov ipd. Ti podporni procesi so visoko strukturirani in zahtevajo podatkovno precizno strukturirane izhodne informacije iz mnogo manj strukturiranih informacij (»case-like«) procesov, kot je razvoj novega izdelka. Z vidika razvijalcev je to nepotrebna preciznost oz. birokracija. Hkrati so neprecizne povezave procesa razvoja novega izdelka do podpornih procesov zelo dovzetne za generiranje nepotrebnih novih variant v podpornih procesih, za podatkovna neskladja, neizvedljivost (nezaželeno pojavljanje alternativ).

V drugem koraku smo določili mejnike, da so skladni z vsebino procesa, torej z izrabo najboljše prakse na področju organizacije procesa razvoja novega izdelka. Ker pri srednje velikem podjetju želimo



Slika 4: **Koncept mejnikov procesa razvoja novega izdelka (angl. NPD)** (Roblek idr., 2012)

hitro uvesti spremembe, zmanjšati vpliv obstoječega kulturnega znanja in razkriti implicitno znanje, so zaposleni občutljivi na strokovno usposobljenost vodje preнове. Zato morajo imeti vse predlagane spremembe teoretično in metodološko podlago vsaj v:

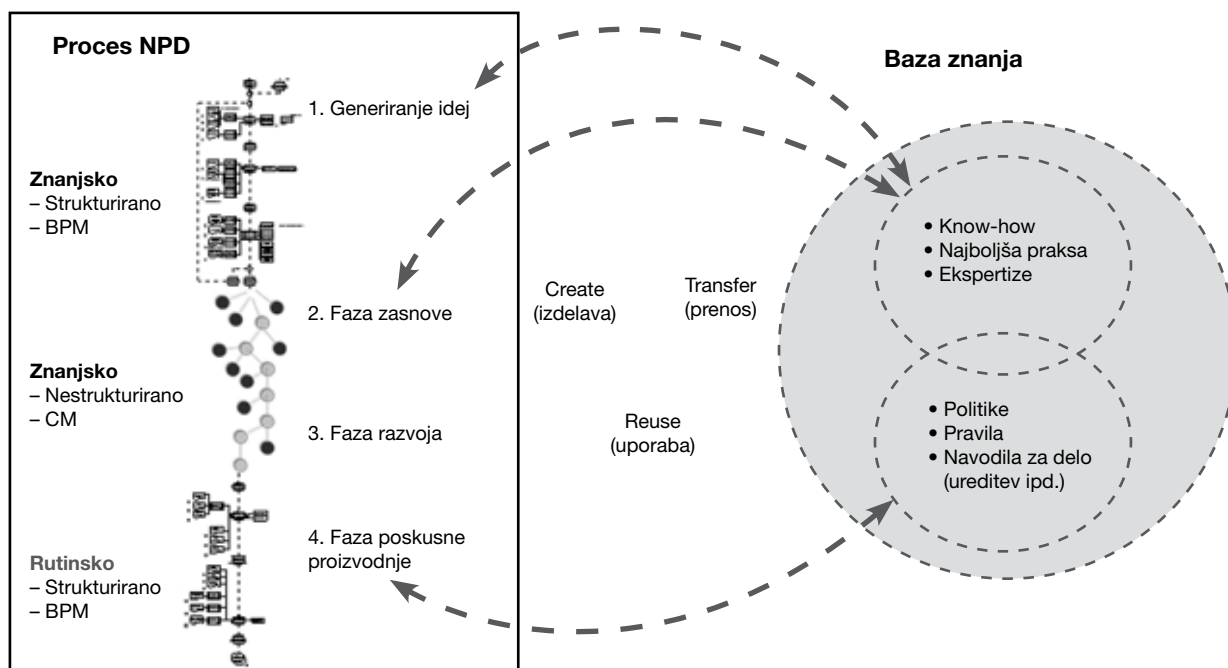
- splošno dokazanih dosežkih raziskav na področju organizacije procesa razvoja novega izdelka, npr. določitev razvojnih faz v tem procesu (Kennet, 2013),
- najboljši praksi na področju organizacije procesa razvoja novega izdelka, ki je neodvisna od panoge, npr. najboljša praksa pri organizaciji procesov, ki obvladujejo življenjski cikel izdelka (angl. Product Lifecycle Operations Reference, PLCOR) (APICS, 2014),
- panožni najboljši praksi na področju organizacije procesa razvoja novega izdelka; za razvoj v avtomobilski panogi npr. metoda APQP (Morris, 2012).

Strokovna podkovanost svetovalcev, ki prenavljajo proces razvoja novega izdelka, omogoča, da bodo zaposleni s svojim implicitnim in kulturnim znanjem zaupali v spremembe in jih sprejeli. S tem tudi presežemo dolgotrajno razpravo o »internih posebnostih« v procesu razvoja novega izdelka in dokažemo, da posebnosti niso strokovno utemeljene.

Drugi korak metodologije vsebuje tudi določitev ključnih aktivnosti, vhodno-izhodnih dokumentov,

pravil, formul, delovnih vlog, s katerimi je opremljena posamezna aktivnost procesa razvoja novega izdelka, da je dobro opravljena v smislu optimuma celotnega procesa (slika 5). V strukturiranem delu procesa razvoja novega izdelka je natančno določeno tudi zaporedje aktivnosti. Nato sledi še zadnji del preнове, zasedanje delovnih vlog z zaposlenimi, njihovimi lastnimi kompetencami, ki so značilnosti konkretnega srednje velikega podjetja. To lahko povzroči tudi spremembo organizacijske strukture podjetja.

V tretjem koraku preнове smo pristopili k avtomatizaciji spremenjene organizacije procesa razvoja novega izdelka. Za srednje velika podjetja so investicije v informacijsko-komunikacijsko tehnologijo relativno veliko finančno breme pri prenovi procesa. Zato so podjetja pripravljena sprejeti kompromise med načrtano strateško prenovno procesa (ekskluzivnostjo) in operativno sposobnostjo (ceno) informacijske rešitve. To pomeni, da se najnižja raven spreminjanja procesa (zapisi podatkov, izmenjave informacij med sistemi) dogaja sočasno z implementacijo ali prilagoditvijo že izbranega informacijskega sistema. To je dejansko že prva iteracija postopne izboljšave novega procesa razvoja novega izdelka (novi kot bo). V tem koraku preнове je treba določiti, katere aktivnosti procesa razvoja novega izdelka zahtevajo:



Slika 5: **Strukturiranje procesa razvoja novega izdelka (angl. NPD) v skladu z želeno formalizacijo eksplicitnega znanja**

- transakcijski način podpore; to velja predvsem za strukturirane dele procesa razvoja novega izdelka, pri katerih je prisotna velika frekvenca generiranja podatkov in informacij, ki so na voljo podpornim oz. povezanim procesom srednje velikega podjetja, npr. moduli ERP (obračunski del, plače), specialni moduli ERP (npr. za vodenje projektov);
- dokumentni način podpore; to je primerno predvsem za delno in popolnoma nestrukturirane dele procesa razvoja novega izdelka, pri katerih vodimo verzije datotek in skrbimo, da se te v nekem dogovorjenem zaporedju potrjujejo, npr. sistemi DMS in Workflow;
- znanjski način podpore; to so sistemi CAD/CAM, simulatorji, ki služijo za podporo razvoja in izdelave izdelka;
- analitični način podpore; pri čemer gre za orodja poslovne inteligence (angl. Business Intelligence, BI), s katerimi na poslovni ravni spremljamo kazalnike uspešnosti in učinkovitosti procesa razvoja novega izdelka.

4 REZULTATI PRENOVE PROCESA RAZVOJA NOVEGA IZDELKA

Podjetje, v katerem je bila opravljena prenova, je po slovenski zakonodaji srednje veliko podjetje in deluje na področju razvoja in proizvodnje sestavnih delov kot sistemski dobavitelj za avtomobilsko in druge industrije. Podjetje je pretežno izvozno naravnano in sodeluje v mednarodnih razvojno oskrbovalnih verigah.

Od zaključka uvedbe sprememb v procesu razvoja novega izdelka v referenčnem srednje velikem podjetju je preteklo pol leta. Po raziskavi McAfee (2002) je preteklo premalo časa, da bi lahko verodostojno analizirali spremembo klasičnih ekonomskih

kazalnikov pred prenovu in po njej. V prispevku predstavljamo preliminarne rezultate v spremembi izbranih procesnih strukturnih kazalnikov, ki dokazano (Urh idr., 2015) ustrezno napovedujejo kasnejše dejanske spremembe klasičnih ekonomskih kazalnikov. Analiza strukturnih kazalnikov je bila izvedena s pomočjo analitičnega modula orodja ARIS 9.8., v katerem jih tudi izračunavamo.

4.1 Analiza spremembe organizacijskih elementov (izvajalcev) v procesu

Z analizo določimo za vse zaporedno izvajane aktivnosti (v nadaljevanju funkcije) izbranega procesa, ali obstaja pri tranziciji med dvema funkcijama sprememba izvajalca v organizacijski enoti, delovni skupini, delovni vlogi ali osebi.

Organizacijski elementi in funkcije procesa so povezani prek dveh tipov povezav: ali sodelujejo pri izvedbi ali so odgovorni za izvedbo. Če imajo vsi organizacijski elementi tip povezave »sodeluje« na funkcijo, so vsi organizacijski elementi potrebni, da se funkcija izvede. Če imajo vsi tip povezave »odgovoren za izvedbo«, je za izvedbo funkcije potreben le en organizacijski element, ostali predstavljajo alternativne izvajalce.

Kot rezultat izvajanja funkcij z alternativnimi organizacijskimi elementi izračunavamo dve vrednosti menjav organizacijskih elementov v zaporedju procesa: če je sprememba možna in neobvezna (alternativa), se minimalna vrednost ne poveča, poveča se maksimalna vrednost. V primeru obvezne spremembe se povečata obe vrednosti. Zaradi poenostavitve pri vrednotenju pri analizi vsa razvejanja v procesu obravnavamo kot operator IN. Pojavna kopija objekta predstavlja večkratno pojavljanje objekta na analiziranem modelu, ki je enolično zapisan v repozitoriju objektov.

Tabela 1: **Organizacijska sprememba: primerjava kot je in kot bo**

Stopnja organizacijske spremembe/izboljšave v procesu RNP	Kot je	Kot bo
a) Število procesnih funkcij	144	77
b) Število različnih organizacijskih vlog	43	38
c) Število menjav organizacijskih vlog (pojavnih kopije) pri prehodu med funkcijami procesa (tranzicije)	125	96
č) Minimalno število organizacijskih sprememb	73	86
d) Razmerje min. št. org. sprememb (č) / tranzicij (c)	0,58	0,9
e) Maksimalno število organizacijskih sprememb	90	90
f) Razmerje maks. št. org. sprememb (e) / tranzicij (c)	0,72	0,94

Analiza spremembe nosilcev informacij v procesu

Za vse funkcije v zaporedju izvajanja izbranega procesa izračunavamo spremembe v nosilcih informacij (npr. dokumentih, e-pošti, datoteki). Menjava obstaja, če dve zaporedni funkciji (ena tranzicija) v procesu nimata skupnega vsaj enega identičnega nosilca informacij.

Tabela 2: **Dokumentacijska sprememba: primerjava kot je in kot bo**

Stopnja dokumentne spremembe/izboljšave v procesu RNP	Kot je	Kot bo
a) Število različnih nosilcev informacij	99	106
b) Število tranzicij nosilcev informacij	50	81
c) Tranzicije z menjavami nosilcev informacij	20	28
č) Razmerje menjav nosilcev informacij (c) / število tranzicij (b)	0,4	0,35

Analiza spremembe informacijskih sistemov v procesu

Izračunava za vse funkcije procesa, ali so podprte z informacijskim sistemom. Poleg tega se za vse zaporedne funkcije preverja, ali obstaja zamenjava informacijskega sistema. Zamenjava obstaja, če dve zaporedni funkciji (ena tranzicija) nimata vsaj enega identičnega informacijskega sistema.

Tabela 3: **Informacijska sprememba: primerjava kot je in kot bo**

Stopnja informacijske spremembe/izboljšave v procesu RNP	Kot je	Kot bo
a) Število različnih informacijski sistemov ali modulov	44	17
b) Funkcije z informacijskimi sistemi v %	45,65	62,34
c) Število tranzicij informacijskih sistemov	91	63
č) Število menjav informacijskih sistemov	32	17
d) Razmerje menjav števila informacijskih sistemov (č) / število tranzicij (c)	0,35	0,27

5 RAZPRAVA

Iz primera in na podlagi ugotovitev, ki so navedene v nadaljevanju, lahko sklepamo, da je predstavljeni pristop mogoče uporabiti v okoljih, v katerih potekajo projekti razvoja novih proizvodov.

- Iz tabele 1 je razvidno, da je bil prenovljeni proces poenoten: število funkcij v novem procesu se je zmanjšalo za 67 glede na obstoječi proces pri isti konvenciji zapisa procesa (za lažjo predstavbo navajamo, da ima povprečna razvojna kontrolna lista APQP, ki zagotavlja odlično razvit izdelek in njegov proces, prek 300 kontrol dokumentov in aktivnosti (pri tem povratne zanke oz. neuspešni

prehodi niso upoštevani); to pomeni, da jih novi proces obdela mnogo bolj vitko in učinkovito. To smo storili pretežno s poenotenjem in združevanjem dela zaposlenih v formaliziranem »strukturiranem delu« procesa razvoja novega izdelka. V proces je alociranih za 12 odstotkov manj različnih delovnih vlog (*kot bo* 38 proti *kot je* 43). Razmerje med minimalnim in maksimalnim številom delovnih vlog glede na število funkcij je v procesu *kot bo* poenoteno, kar kaže na odpravo alternativnih poti oz. ad hoc procesnih bližnjic. Ker je razmerje v obeh primerih v *kot bo* večje od *kot je* (v *kot bo* je razmerje med min. in maks. skoraj 1 : 1), je ob upoštevanju zmanjšanja števila delovnih vlog razvidno, da je odgovornost za izvedbo posamezne aktivnosti razvoja novega izdelka jasno alocirana posameznemu izvajalcu.

- Iz tabele 2 je razvidno, da se je število nosilcev informacij povečalo s *kot je* 99 na *kot bo* 106. To utemeljuje večjo formalizacijo procesa oz. višjo stopnjo kreiranja eksplicitnega znanja, kot se zahteva od srednje velikega podjetja glede na malo podjetje (kar je bil cilj projekta). Da stopnja formalizacije ni poslabšala procesa kot celote, kaže razmerje števila menjav nosilcev informacij glede na število tranzicij funkcij v procesu, ki se je v *kot bo* znižalo za 5 odstotkov. To pomeni, da je kljub povečanju števila nosilcev informacij na posamezni funkciji prišlo do relativnega zmanjšanja števila nosilcev, ki se pretakajo skozi proces in tako ne motijo izvajanja toka celotnega procesa. Poleg tega je v procesu *kot bo* zasnova polnjenja nosilcev informacij (sporočil) urejena tako, da se podatek zapiše na mestu nastanka, se ažurira na enem mestu in se »uporabi« na različnih nosilcih informacij, kar bistveno olajša elektronsko obvladovanje dokumentacije.
- Iz tabele 3 je razvidno, da se je v procesu *kot bo* zelo poenotila informacijska tehnologija. Število uporabljenih različnih informacijskih sistemov ali modulov se je zmanjšalo za 61,5 odstotka (na 17 v *kot je*; pojasniti je treba, da se po konvenciji zapisa npr. vsak Excel vrednoti kot svoj element, zato tako izrazit upad s 44 na 17 elementov). Pri tem pa se je povečala stopnja avtomatizacije procesa, saj je v novem procesu informacijsko podprtih cca. 17 odstotkov več funkcij glede na število funkcij v celotnem procesu. Pri tem se je tudi sistemsko poenotila uporaba določene programske opreme,

kar kaže znižano razmerje menjav informacijskih sistemov glede na število funkcij v procesu. Pri tem moramo dodati, da je v času pisanja prispevka cca. polovica procesa razvoja novega izdelka še neoptimizirana z vidika informacijske podpore in da po optimizaciji pričakujemo še boljše rezultate.

- Izračuni *kot bo* niso rezultati simulacije preнове, temveč podatki modela živega novega procesa. Na podlagi doseženih vrednosti navedenih strukturnih kazalnikov lahko potrdimo, da nam je z omenjenim pristopom in metodologijo preнове uspelo doseči, da smo dovolj dobro formalizirali proces razvoja novega izdelka v obravnavanem srednje velikem podjetju (kot priporoča Robbins, 2014) ter zagotovili, da se večina pomembnega implicitnega znanja zapisuje in je na voljo kot eksplicitno znanje oz. kot baza znanja za hitrejše in kakovostnejše kasnejše ponovitve razvoja novih izdelkov. S tem ko smo del procesa razvoja novega izdelka pustili v nestrukturirani obliki, smo hkrati omogočili, da se ohrani fleksibilnost in agilnost razvoja, kot priporočajo Leithold idr. (2016).

6 SKLEP

V prispevku predstavljeni pristop smo oblikovali, ker je kljub prisotnosti velikega števila dobrih praks za velika podjetja zelo malo primerov najboljše prakse organizacijskega spreminjanja procesa razvoja novega izdelka pri tranziciji iz majhnega podjetja (t. i. obrtniške organiziranosti) v industrijsko obliko organiziranosti v srednje velikem podjetju. V tovrstni transformaciji je trenutno v Sloveniji relativno veliko podjetij.

Poudariti želimo, da je v prispevku prikazani primer uporabe pristopa na specifičnem področju, na katerem deluje konkretno podjetje. Primer nam je služil predvsem za specifikacijo razlik in novitet novega pristopa v primerjavi z že znanimi. Hkrati je treba poudariti, da je bil razvoj pristopa postopen in je bil doslej v različnih evolutivnih fazah večkrat preizkušen v podobnih podjetjih. Ob upoštevanju nadgradnje in postopne izboljšave ter rezultatov posameznih prenov lahko ovrednotimo predstavljeni pristop kot učinkovit in zato kot pomembno metodološko noviteto na tem področju. Ocenjujemo, da je pristop dovolj posplošen, da ga je ob prilagoditvah posebnostim konkretnega projekta mogoče koristno uporabiti za prenavljanje in pomoč pri

prilaganju informacijske podpore razvojnih procesov v poljubnem srednje velikem podjetju.

Z objavo prispevka želimo spodbuditi nadaljnje raziskave in testiranja opisanega pristopa v drugih okoljih, kar bi lahko koristilo strokovnjakom v srednje velikih podjetjih pri reševanju podobnih težav.

7 LITERATURA

- [1] APICS, (2014). *PLCOR Product Lifecycle Operations Reference Model*, APICS Supply Chain Council.
- [2] Basu, S., Palvia P., Chen, L. (2003). *Encyclopedia of Library and Information Science*, Second Edition.
- [3] Drakos, N. (2014). *How to Choose the Right Technology for a Knowledge Management Program*. Gradivo dostopno na <http://www.gartner.com>.
- [4] FactBook.org (2015). *Newness To Market*. Gradivo dostopno na <http://www.factbook.org>.
- [5] Harmon, P. (2015). *Once More on Business Process Methodologies. BPR Trends*. Gradivo dostopno na <http://www.bp-trends.com/bpt/wp-content/uploads>.
- [6] Horie, N., in Ikawa, Y. (2014). *Knowledge integration in a product development organization for new businesses: A case study of a precision device manufacturer*. Management of Engineering & Technology (PICMET), Portland International Conference.
- [7] Kennet, K. (2013). *The PDMA handbook of new product development* (3. izd.). Hoboken, New Jersey: John Wiley & Sons Inc., 34.
- [8] Kern, T. (2003). The model of dynamic adaptation of business structures to changing processes. V: Palvia, Prashant (ur.). *Proceedings of the Fourth Annual Global Information Technology Management World Conference*, Calgary, Alberta, Canada, June 8, 9, 10, 2003. Greensboro (NC): Global Information Technology Management Association, 278–281.
- [9] Kern, T. (1998). *Procesna organizacija: oblikovanje organizacije poslovnih sistemov na osnovi modela strukturiranih organizacijskih procesov*. Doktorska disertacija. Univerza v Mariboru, Fakulteta za organizacijske vede Kranj, 202.
- [10] Kokalj, Š. (2012). *Mehanizem za ocenjevanje strukturne učinkovitosti procesov*. Magistrsko delo, Univerza v Mariboru, Fakulteta za organizacijske vede Kranj.
- [11] Leithold, N., Woschke, T., Haase, H., Kratzer, J. (2016). Optimising NPD in SMEs: a best practice approach. *Benchmarking: An International Journal*, 23, 262–284.
- [12] McAfee, A. (2002). The impact of enterprise information technology adoption on operational performance: An empirical investigation. *Production and Operations Management*, 11, 33–53.
- [13] Morris, M. (2012). *Advanced Product Quality Planning and Control Plans based on APQP*. 2nd Edition. ASQ Automotive Division.
- [14] Petrović, D., Jovanović, P., Raković, R. (2010). *Upravljanje projektnim rizicima*. Udruženje za upravljanje projektima Srbije. Beograd, 239.
- [15] Poetz, M. K., in Schreier, M. (2012). The Value of Crowdsourcing: Can Users Really Compete with Professionals in Generating New Product Ideas?. *The Journal of Product Innovation Management*.
- [16] Poličar, M. (2008). *Organizacijska kultura v družinskih podjetjih*. Diplomsko delo. Univerza v Mariboru, Fakulteta za organizacijske vede Kranj.

- [17] Rajasekaran, K. G., Bhaskar, G. B., Murali, S., Chandrasekaran, M. (2016). Identification and Prioritisation of Supplier, Customer and Organization Collaborating Factors Influencing New Product Development. *Indian Journal of Applied Research*, 7.
- [18] Robbins, P. (2014). *How having an innovation strategy and process can improve NPD outcomes for Irish SME's*. The XXV ISPIIM Conference – Innovation for Sustainable Economy & Society, Dublin.
- [19] Roblek, M., Kern, T., Zajec, M. (2012). Knowledge management of the new product development process with process-knowledge allocation model. V: *The 6th IEEE International Conference on Management of Innovation and Technology, Sanur Bali, Indonesia. ICMIT 2012*, 214–219.
- [20] Roblek, M., in Kern, T. (1998–2016). *ARIS procesni repozitoriji podjetij*. LIPPS – Laboratorij za inženiring poslovnih in produkcijskih sistemov, Univerza v Mariboru, Fakulteta za organizacijske vede. Interno gradivo.
- [21] Sinderman, B. (2012). *Three Things Mid-Sized Companies Do Better*. *Forbes*. Gradivo dostopno na <http://www.forbes.com/sites/forbesinsights/2012/04/20/three-things-mid-size-companies-do-better>.
- [22] Sosa, E. M., Gargiulo, M., Rowles, C. (2015). Can Informal Communication Networks Disrupt Coordination in New Product Development Projects?. *Organization Science*, 4, 1059-1078.
- [23] Stevenson, W. J. (2015). *Operations Management*. 12th Edition. McGraw-Hill, New York, 236–262.
- [24] Urh, B. (2011). *Predvidevanje uspešnosti poslovnega sistema z vidika obvladovanja učinkovitosti poslovnih procesov*. Doktorska disertacija. Univerza v Mariboru, Fakulteta za organizacijske vede Kranj.
- [25] Urh, B., in Kern, T. (2015). Is the structural efficiency of business processes connected to the effectiveness. *Internationalization and cooperation: proceedings of the 34th International Conference on Organizational Science Development. Moderna organizacija. Kranj*, 1221–1230.
- [26] ZGD (2016). *Zakon o gospodarskih družbah*. Gradivo dostopno na <https://zakonodaja.com/zakon/zgd-1/55-clen-mikro-majhne-srednje-in-velike-druzbe>.
- [27] Woschke, T., Haase, H., Lautenschläger, A. (2016). Waste in NPD processes of German SMEs. *International Journal of Productivity and Performance Management*, 4, 532–553.

■

Matjaž Roblek je zaposlen na Fakulteti za organizacijske vede Univerze v Mariboru kot visokošolski učitelj. Habilitiran je v naziv docent za področje inženiring poslovnih in delovnih sistemov. Pedagoško delo opravlja pri predmetih Poslovni in proizvodni informacijski sistemi, Menedžment oskrbovalne verige ter Planiranje in vodenje proizvodnje. V sodelovanju z gospodarstvom ima končanih več kot šestdeset raziskovalnih in aplikativnih projektov s področja prenove in informatiziranosti poslovnih procesov. Je avtor ali soavtor več kot sto znanstvenih, strokovnih in drugih publikacij. Trenutno je predsednik akademskega zbora Fakultete za organizacijske vede.

■

Tomaž Kern je zaposlen na Fakulteti za organizacijske vede Univerze v Mariboru kot visokošolski učitelj. Habilitiran je v naziv redni profesor za področje organizacijskih in informacijskih sistemov. Pedagoško delo opravlja pri predmetih na dodiplomski in podiplomski stopnji. Je avtor ali soavtor več kot štiristo petdeset znanstvenih in strokovnih člankov in drugih publikacij. Je vodja več raziskovalnih projektov in član raziskovalnih skupin v raziskovalnih projektih. Aktivno sodeluje pri prenosu raziskovalnega znanja v prakso. Med drugim je bil prodekan za raziskovalne zadeve, predstojnik inštituta, član upravnega odbora univerze, prorektor za informatiko. Trenutno je član senata univerze.

▣ Znanja in sposobnosti podatkovnih znanstvenikov: pregled in analiza stanja v Sloveniji

¹Mateja Grobelnik, ²Jurij Jaklič

¹Petrol, d. d., Dunajska cesta 50, 1000 Ljubljana

²Univerza v Ljubljani, Ekonomska fakulteta, Kardeljeva ploščad 17, 1000 Ljubljana
mateja.grobelnik@gmail.com; jurij.jaklic@ef-uni-lj.si

Izveleček

Masovne podatke in znanost o podatkih so organizacije prepoznale kot vira novih konkurenčnih prednosti. Z namenom izkoriščanja tega potenciala se je povečalo povpraševanje po posameznikih s specifičnimi znanji in sposobnostmi, ki so sposobni iz množice raznolikih podatkov pridobiti koristne informacije in jih na razumljiv način implementirati v obstoječe procese in aktivnosti v organizaciji, po t. i. podatkovnih znanstvenikih. Z neprestanim razvojem področja znanosti o podatkih in ob raznolikih potrebah po analitičnih znanjih prihaja do različnega razumevanja vloge podatkovnih znanstvenikov, hkrati pa se večja vrzel med ponudbo in povpraševanjem po takšnih posameznikih. Namen prispevka je zato prispevati k boljšemu razumevanju ter opredeliti znanja in sposobnosti podatkovnih znanstvenikov, s pomočjo raziskave v Sloveniji pa ugotoviti trenutna znanja in sposobnosti ter identificirati segmente podatkovnih znanstvenikov v Sloveniji. Na podlagi razvrščanja v skupine samoocene znanj je bilo identificiranih in opisanih pet skupin: trženjski raziskovalci analitiki, podatkovni analitiki, raziskovalci, programerji in podatkovni znanstveniki, skupaj s priporočili za njihov nadaljnji razvoj.

Ključne besede: masovni podatki, znanost o podatkih, podatkovni znanstvenik, sposobnosti, znanja, razvrščanje v skupine, empirična raziskava.

Abstract

Knowledge and Skills of Data Scientists: Overview and Analysis of Current Situation in Slovenia

Big data and data science have been recognized by organizations as sources of a new competitive advantage. In order to exploit their potential, there has been an increase in demand for individuals with specific knowledge and skills who are capable of obtaining useful information from a set of diverse data and implement it into existing processes and activities in an organization, that is so-called data scientists. Due to the continuous development of the data science field and the diverse needs for analytical knowledge, the understanding of the role of data scientists deviates greatly, while the gap between supply and demand for such individuals is also increasing. Therefore, the objective of this paper is to contribute to a better understanding and definition of the knowledge and skills of data scientists, and to identify the current knowledge and skills together with the segments of data scientists through empirical research in Slovenia. Five segments were identified and described: "Marketing researchers – analysts", "Data analysts", "Researchers", "Developers" and "Data Scientists", together with recommendations for their future development.

Keywords: big data, data science, data scientist, skills, knowledge, clustering, empirical research.

1 UVOD

Stroškovno učinkovito shranjevanje podatkov, konvergenca pametnih naprav, družbenih omrežij, širokopasovnih komunikacij in analitike so na novo definirali odnose med proizvajalci, distributerji ter potrošniki izdelkov in storitev, hkrati pa ustvarili nove izzive in priložnosti. Olofson in Vesset (2012) to konvergenco imenujeta inteligentna ekonomija. Sama

zmožnost shranjevanja in dostop do podatkov namreč nista dovolj, šele ko imamo možnost podatke analizirati in na podlagi rezultatov sprejemati boljše odločitve, ustvarjamo konkurenčno prednost (Olofson in Vesset, 2012). Povečali smo si možnosti zajema veliko večje količine podatkov različnih tipov, ki nastajajo z veliko hitrostjo.

Ti podatki prihajajo iz različnih (ne)zaupanja vrednih virov, ki lahko organizaciji prinesejo dodano vrednost. Navedeni koncepti se povezujejo s pojmom masovni podatki (angl. big data) in znanost o podatkih (angl. data science). Če so se na eni strani povečale možnosti zbiranja in shranjevanja podatkov, so se po drugi strani razvile tudi nove tehnologije na področju strojne in programske opreme za analizo in obdelavo večje količine podatkov.

Skupaj z razvojem tega področja se je pojavila potreba po specifičnih znanjih, s pomočjo katerih je mogoče iz velike količine različnih podatkov pridobiti uporabne informacije za izboljšanje poslovnih odločitev. Kot enega glavnih izzivov pri uvajanju strategije masovnih podatkov in znanosti o podatkih navajajo pomanjkanje posameznikov s specifičnimi znanji in sposobnostmi ustvarjanja dodane vrednosti s pomočjo manipulacije s podatki (Big Data Executive Survey, 2012). Na podlagi te potrebe po novih znanjih s področja analitike masovnih podatkov se povečuje povpraševanje po podatkovnih znanstvenikih (angl. data scientists).

Raziskava MGI in McKinsey's Business Technology Office (Manyika idr., 2011) napoveduje, da bo do leta 2018 samo v ZDA nastala od 50- do 60-odstotna vrzel med ponudbo in povpraševanjem po poglobljenem analitičnem znanju, torej pomanjkanje od 140.000 do 190.000 posameznikov z naprednimi znanji na področju statistike ali strojnega učenja in 1,5 milijona menedžerjev in analitikov s sposobnostmi uporabe analitike masovnih podatkov za sprejemanje učinkovitih odločitev. Čeprav so analizo opravili v ZDA, menijo, da bo pomanjkanje poglobljenega analitičnega talenta svetovni trend. Države z višjim številom posameznikov s poglobljenim analitičnim znanjem na prebivalca bodo v prihodnosti privlačen vir teh sposobnosti za druga geografska področja prek migracije ali prek zaposlitev v organizacijah. Po Gartnerju so napovedali, da bo do leta 2015 4,4 milijona delovnih mest na področju informatike po svetu namenjenih podpori delu z masovnimi podatki (Chordas, 2014, str. 23). Povpraševanje zajema vse od inženirjev masovnih podatkov (angl. big data engineer), podatkovnih analitikov (angl. data analyst) do poslovnih analitikov (angl. business analyst). Največji izziv pa naj bi bilo najti podatkovne znanstvenike, saj gre za posameznike, ki imajo ključno vlogo pri uporabi masovnih podatkov: podatke priskrbijo in jih uporabijo za poslovne odločitve (Chordas, 2014,

str. 23). V sklopu raziskave New Vantage Partners Big Data Executive Survey (2012, str. 8) so ugotovili, da ima kar 70 odstotkov vprašanih odločevalcev v organizacijah na področju masovnih podatkov namen zaposliti podatkovne znanstvenike, vendar jih 80 odstotkov meni, da je to zanje velik izziv.

Namen prispevka je zato prispevati k razumevanju ter opredelitvi znanja in sposobnosti podatkovnih znanstvenikov, s pomočjo raziskave med podatkovnimi znanstveniki v Sloveniji pa ugotoviti trenutna znanja in sposobnosti ter z uporabo razvrščanja v skupine identificirati segmente podatkovnih znanstvenikov v Sloveniji. Z opredelitvijo masovnih podatkov, prek procesa izvajanja znanosti o podatkih ter s pregledom literature in raziskav so bile identificirane različne sposobnosti in področja znanj podatkovnih znanstvenikov. 92 posameznikov v Sloveniji, ki se večino časa ukvarjajo s podatki, je izpolnilo strukturiran spletni vprašalnik, ki je vključeval vprašanja, povezana z dimenzijami masovnih podatkov, samooceeno znanj, pomembnostjo znanj in sposobnosti, njihovimi dosedanjimi izkušnjami ter načini izobraževanja. Z uporabo razvrščanja v skupine so bili identificirani segmenti podatkovnih znanstvenikov.

V drugem razdelku je predstavljen koncept masovnih podatkov in znanosti o podatkih. V nadaljevanju so opredeljena znanja in sposobnosti podatkovnih znanstvenikov, ki so bili uporabljeni pri oblikovanju vprašalnika. Tretji razdelek vključuje metodologijo empirične raziskave o znanjih in sposobnostih podatkovnih znanstvenikov v Sloveniji, značilnosti vzorca ter rezultate raziskave. Na koncu so podane sklepne ugotovitve.

2 MASOVNI PODATKI IN ZNANOST O PODATKIH

2.1 Opredelitev masovnih podatkov

Posamezniki in organizacije z vsakodnevnim delovanjem danes ustvarimo več podatkov kot kadar koli do sedaj. Podatki nastajajo povsod: na družbenih medijih (angl. social media), kot so Twitter, Facebook, LinkedIn, Instagram itd., spletnih straneh, ob izvedbi nakupnih transakcij, ob aktivaciji GPS signalov mobilnih telefonov, z uporabo RFID značk, na mobilnih aplikacijah, in prav vse te podatke je mogoče shraniti v digitalni obliki. Masovni podatki danes veljajo za popularen trend, ki se v bistvu nanaša predvsem na problem volumna/hitrosti/raznolikosti podatkov (angl. Volume/Velocity/Variety problem). Glavna

prednost masovnih podatkov je, da lahko s pomočjo analize le-teh pridobimo zanimive vzorce in informacije, ki so bili poprej skriti, saj jih zaradi velike količine dela in časa ni bilo mogoče pridobiti. Sedaj pa jih lahko uporabimo za analizo, sprejemanje odločitev ter razvoj novih produktov, kar pomeni znatno konkurenčno prednost (Lorica, Howard in Dumbill, 2012).

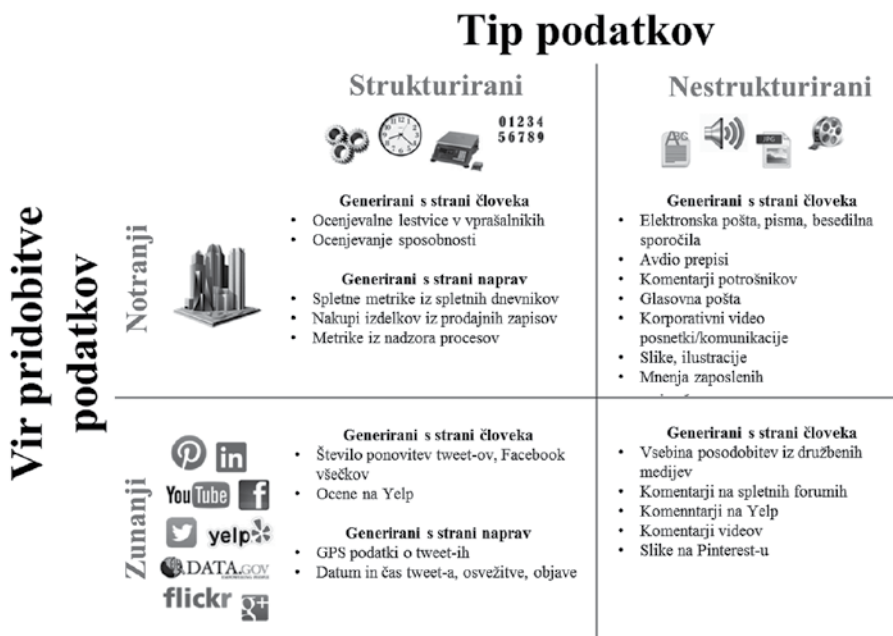
Zaradi dejstva, da pojem masovni podatki in vse, kar dojemamo pod strategijo vpeljave masovnih podatkov, vpliva in zajema širok nabor poslovnih procesov, tehnologij in strokovnih znanj, lahko na izraz masovni podatki gledamo v ožjem in širšem smislu. Če na izraz gledamo v ožjem smislu, gre predvsem za opredelitev tega, kakšne značilnosti morajo imeti podatki, da jih lahko opredelimo kot masovne. Večina definicij masovnosti podatkov ne povezuje le s količino podatkov (volumen), temveč so enako pomembne tudi druge dimenzije podatkov: hitrost, s katero nastajajo podatki, raznolikost virov/strukture podatkov ter vrednost podatkov (najpogosteje omenjene).

Volumen kot dimenzija masovnih podatkov se nanaša na velike količine podatkov, ki se dejansko tudi shranijo, saj so se stroški shranjevanja podatkov občutno pocenili (npr. danes lahko shranimo celotno svetovno zalogo glasbe na napravo v vrednosti 500 dolarjev) (Dhar, 2013, str. 67). Glavna prednost je, da lahko s pomočjo večjih vzorcev, ki so boljši približek

populaciji, gradimo bolj natančne napovedne modele. Večanje nabora podatkov pomeni izziv obstoječim tradicionalnim strukturam informacijskih tehnologij, saj masovni podatki zahtevajo razširljivo skladiščenje in porazdeljen pristop k poizvedovanju. Rešitve so na voljo v obliki podatkovnih skladišč ali rešitev, ki izhajajo iz Apache Hadoop (Lorica, Howard in Dumbill, 2012).

Hitrost se nanaša na vedno večjo stopnjo hitrosti, s katero pridobivamo podatke. Pomembnost dimenzije hitrosti masovnih podatkov leži v hitrosti povratne zanke. To pomeni, da je treba delovati in ukrepati na podlagi podatkov v realnem času. Bolj tesna je zanka, večja je konkurenčna prednost (Lorica, Howard in Dumbill, 2012). Tehnologija dimenzije hitrosti masovnih podatkov sega od paketne obdelave (angl. batch processing) ob določenih intervalih do konstantnega toka podatkov (angl. streaming data) v realnem času (Olofson in Vesset, 2012, str. 5).

Raznolikost podatkov lahko opredelimo z vidika več različnih dimenzij. Podatki so lahko opredeljeni z vidika podatkovnega tipa (strukturirani, delno strukturirani in nestrukturirani ipd.), vira pridobitve (notranji, zunanji) ter izvora (generirajo jih naprave ali človek) (Hayes, 2014a). Bistvo tretje dimenzije masovnih podatkov, tj. raznolikosti podatkov, je v tem, da lahko kljub različnim virom, različnemu izvoru in nestrukturiranosti podatkov iz njih izluščimo



Slika 1: Ogradje za opredelitev raznolikost podatkov
Vir: B. E. Hayes, The what and where of big data: A data definition framework, 2014a.

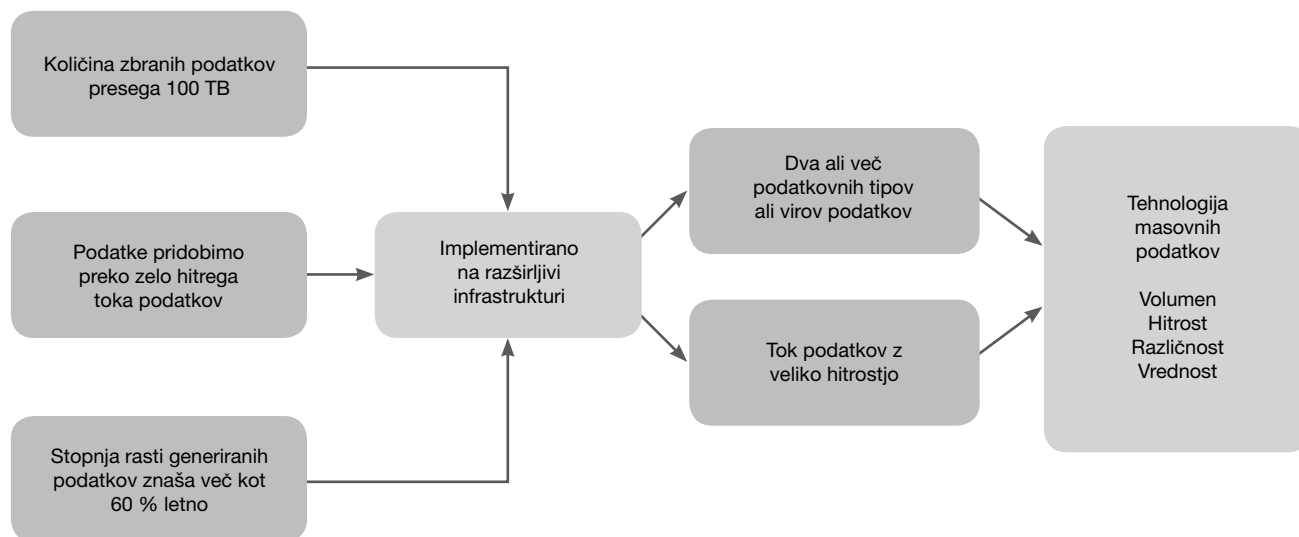
urejeno znanje, ki ga lahko uporabi človek ali pa ga uporabimo kot vhodne podatke v aplikacijo (Lorica, Howard in Dumbill, 2012). Slika 1 prikazuje ogrodenje za opredelitev raznolikosti podatkov po Hayesu (2014a). Model razločuje tip podatkov od vira pridobitve podatkov. Stolpca predstavljata tip podatkov (strukturirani ali nestrukturirani), vrstice pa vir pridobitve podatkov (zunanj ali notranji). Podatki so tako lahko razporejeni v enega od štirih kvadrantov, pri čemer je nadaljnja razdelitev narejena še na ravni izvora podatkov (ki jih generirajo naprave ali človek).

Vrednost podatkov se nanaša na stroške tehnologije in na vrednost, ki jo lahko pridobimo iz masovnih podatkov. Stroški so pomembni predvsem zato, ker pomenijo ključni faktor novosti v masovnih podatkih. Kombinacija razpoložljive programske opreme in nižanje cen strojne opreme je povzročila, da so tehnologije lažje dostopne. Vrednost masovnih podatkov lahko opredelimo z nižjimi stroški kapitala (programske in strojne opreme ter infrastrukture), operativno učinkovitostjo (nižji stroški dela zaradi uporabe bolj učinkovitih metod za integracijo, menedžment, analizo in dostavo podatkov) in izboljšanjem poslovnih procesov (povečanje prihodkov ali dobička na račun novih ali boljših načinov poslovanja, vključujoč izboljšave v komercialnih transakcijah, trajnem menedžmentu skupnosti in primerni distribuciji socialnih, zdravstvenih in

izobraževalnih storitev) (Olofson in Vesset, 2012, str. 6).

Kako kvantificirano opredeliti volumen, hitrost in raznolikost, povzema klasifikacija po IDC (Vesset idr., 2012, str. 2), prikazana v sliki 2. V sklop trga masovnih podatkov se vključuje podatkovne baze (ne glede na to, ali gre za relacijske ali ne), ki presegajo 100 TB, ki so implementirane na razširljivi arhitekturi in ki vsebujejo podatke iz dveh ali več virov/podatkovnih tipov ali ko je stopnja hitrosti zbiranja podatkov zelo visoka. Podobno lahko za masovne opredelimo podatkovne baze, katerih količina je sicer manjša kot 100 TB, vendar je letna stopnja rasti generiranih podatkov 60-odstotna ali več, poleg tega so implementirane na razširljivi infrastrukturi in vključujejo podatke dveh ali več tipov/virov, ali ko je stopnja hitrosti zbiranja podatkov zelo visoka (Vesset idr., 2012, str. 2). Na podlagi opisanih dimenzij lahko ugotovimo, da lahko podatke opredelimo kot masovne v ožjem smislu takrat, ko ustrezajo vsaj dvema od treh dimenzij masovnih podatkov (volumen, hitrost, raznolikost), vendar vedno z namenom prinašanja vrednosti (četrt dimenzija) organizaciji v obliki nižjih stroškov, večje učinkovitosti ali izboljšanja poslovnih procesov.

Iz opisanih značilnosti masovnih podatkov lahko ugotovimo tudi, da ni dovolj, da jih le opredelimo. Treba jih je shraniti, imeti možnost dostopa do njih ter znanja, sposobnosti in orodja, da jih lahko



Slika 2: Kriteriji za opredelitev masovnih podatkov po IDC

Vir: D. Vesset idr., Worldwide Big Data Technology and Services 2012-2016 Forecast., 2012, str. 2, slika 1.

ustrezno obdelamo in rezultate obdelav uporabimo za sprejemanje boljših odločitev. Zaradi omenjene večdimenzionalnosti podatkov in novega pristopa k obdelavi je očitno, da masovni podatki potrebujejo nov pristop, novo strojno in programsko opremo ter druge spremembe, povezane z organizacijo, kulturo in sprejemanjem odločitev. Masovni podatki tako v širšem smislu po IDC (Vesset idr., 2012, str. 1) »predstavljajo novo generacijo tehnologij in arhitekturnih rešitev, katerih namen je pridobiti ekonomsko vrednost iz velike količine različnih tipov podatkov s pomočjo visoko intenzivnega shranjevanja, raziskovanja in analize teh podatkov«. Tudi Boyd in Crawford (2012) v članku *Critical questions for Big Data*, objavljenem v *Information, Communications and Society Journal*, definirata masovne podatke kot kulturni, tehnološki in znanstveni fenomen, ki temelji na prepletanju tehnologije (maksimiziranje računalniške moči in natančnosti algoritmov za zbiranje, povezovanje in primerjavo velikega nabora podatkov), analize (zmožnost iz velikega nabora podatkov identificirati vzorce z namenom, da pridobimo ekonomsko, tehnično, družbeno ali pravno prednost/odločitev) in mitologije (splošno razširjenega spoznanja, da veliki nabori podatkov zagotavljajo višjo stopnjo inteligence in znanja, ki lahko ustvarijo vpoglede, ki so bili prej nemogoči, v duhu resnice, natančnosti in objektivnosti).

2.2 Znanost o podatkih

Ravno v širšem smislu razumevanja masovnih podatkov so ti tesno povezani s področjem znanosti o podatkih. Področje znanosti o podatkih predstavlja rešitev, kako odkriti potencialne vpoglede, ki se skrivajo v masovnih podatkih, in kako premostiti izziv volumna/hitrosti/raznolikosti/vrednosti masovnih podatkov (Voulgaris, 2014, str. 15). Masovni podatki predstavljajo gonilo sprememb na področjih zbiranja, shranjevanja, menedžmenta, analiziranja in vizualizacije podatkov. Vendar pa masovni podatki potrebujejo znanost o podatkih z namenom, da (Somohano, 2013):

- na podlagi podatkov odkrijemo, česar ne vemo,
- pridobimo napovedni vpogled v podatke, na podlagi katerega lahko sprejemamo boljše odločitve,
- ustvarimo nove izdelke in storitve na podlagi podatkov (angl. data products), ki imajo takojšen vpliv na poslovanje,
- komuniciramo uspešne poslovne zgodbe na podlagi podatkov,

- gradimo zaupanje v sprejemanje odločitev, ki prinašajo poslovno vrednost.

Definicije znanosti o podatkih danes večinoma govorijo o interdisciplinarnem področju – kombinaciji znanj in sposobnosti z različnih področij za obdelavo (masovnih) podatkov. Stanton (2013, str. ii) opredeli znanost o podatkih kot nastajajoče področje delovanja, ki se ukvarja z zbiranjem, pripravo, analizo, vizualizacijo, menedžmentom in ohranitvijo velikega nabora informacij. Čeprav znanost o podatkih najbolj tesno povezujemo s področjem baz podatkov in informatiko, je potrebnih še veliko drugih znanj in sposobnosti. O'Reilly (Lorica, Howard in Dumbill, 2012) definira znanost o podatkih kot disciplino, ki kombinira znanja iz matematike, programiranja in znanosti. Raziskava *Big data executive survey* (2012, str. 8) pa je znanost o podatkih opredelila kot disciplino, ki združuje uporabo različnih stopenj statistike, podatkovne vizualizacije, računalniškega programiranja, podatkovnega rudarjenja, strojnega učenja in arhitekture podatkovnih baz z namenom reševanja kompleksnih podatkovnih problemov.

3 ZNANJA IN SPOSOBNOSTI PODATKOVNIH ZNANSTVENIKOV

3.1 Podatkovni znanstveniki

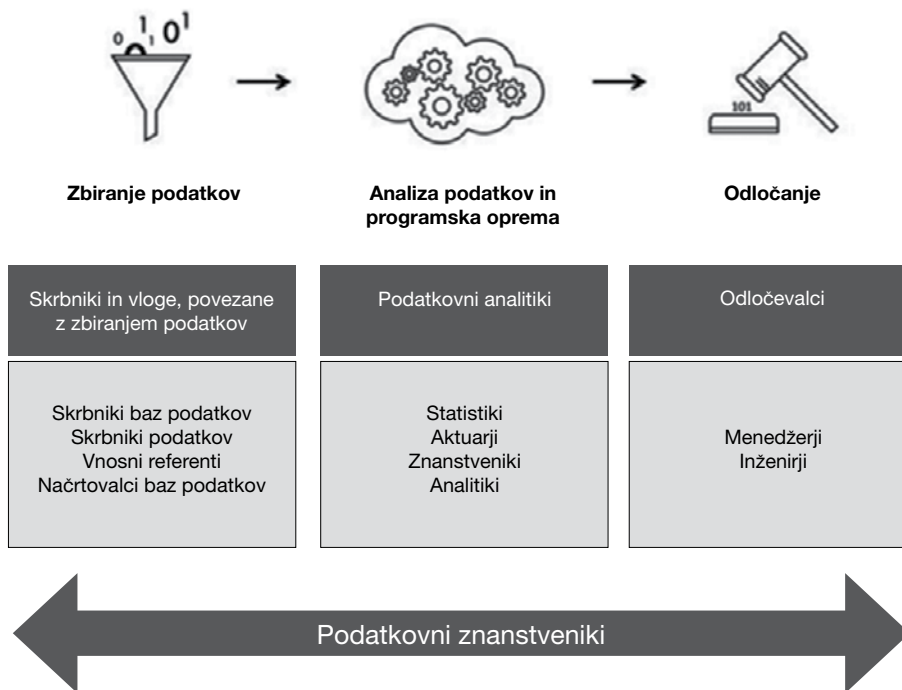
Vedno večja vloga informacijsko-komunikacijskih tehnologij in konvergenca različnih znanstvenih disciplin, kot so matematika in statistika ter tudi naravoslovne in družbene vede z informatiko, pomenita po Organizaciji za ekonomsko sodelovanje in razvoj (v nadaljevanju OECD) (2015, str. 261) pomemben trend v poklicih, povezanih s podatki. Ta konvergenca je omogočila tudi pojav novega razreda podatkovnega strokovnjaka – podatkovnega znanstvenika –, katerega naziv še ni v celoti sprejet, vendar ga različni avtorji uporabljajo za opis »nove« discipline, kategorije dela oziroma karijerne poti, katere pomembnost raste skupaj z masovnimi podatki (OECD, 2015, str. 261). OECD (2015, str. 254) hkrati opozarja, da trenutno tudi še ne obstaja splošno sprejeta definicija znanj in sposobnosti podatkovnih znanstvenikov. Rivera in Haverson (2014) prav tako omenjata, da trenutno še ni standarda glede uporabe nazivov podatkovni znanstvenik in drugih nazivov, povezanih s podatki (podatkovni analitik, podatkovni rudar, podatkovni inženir, statistik, analitik ipd.), ki bi jasno razločeval med različnimi vlogami. Vzroki za slabo definicijo

tega področja so povezani s tem, da gre za relativno nova področja, ki v literaturi niso še dobila dovolj pozornosti v primerjavi z drugimi informacijsko-komunikacijskimi znanji in sposobnostmi, ter s tem, da se področje še vedno razvija (OECD, 2015, str. 254). Thomas H. Davenport in D. J. Patil sta v članku Harvard Business Review: Data Scientist: The Sexiest Job of the 21st Century definirala podatkovne znanstvenike kot posameznike, ki uporabljajo tako podatke kot znanost, da ustvarijo nekaj novega (Davenport in Patil, 2012). Podobno definicijo je podal Voulgaris (2014, str. 18): podatkovni znanstveniki so posamezniki, ki poiščejo smisel v masovnih podatkih. S pomočjo uporabe visoko naprednih tehnologij, znanj in sposobnosti izpeljejo uporabne informacije iz masovnih podatkov, po navadi v obliki novega izdelka ali storitve na podlagi podatkov (angl. data product).

Znanja in sposobnosti podatkovnih znanstvenikov med drugim izhajajo iz osnovne smeri izobrazbe, delovnih nalog, značilnosti in odgovornosti delovnega mesta, na katerem dela posameznik, ter drugih dejavnikov. Zavedati se je treba, da posamezniki, ki se v večji meri ukvarjajo z znanostjo o podatkih, lahko opravljajo delo analitika, programerja, vodje,

menedžerja, profesorja, svetovalca, podjetnika itd. Za ilustracijo obsežnosti področja je Granville (2013) zbral 115 različnih nazivov delovnih mest, neposredno povezanih z masovnimi podatki ali znanostjo o podatkih na podlagi 7500 kontaktov na LinkedInu. Posamezniki z omenjenimi nazivi se lahko v različni meri ukvarjajo z znanostjo o podatkih: od popolnoma operativnih nalog v procesu znanosti o podatkih do vodstvenih položajev, ki zahtevajo dodatna znanja in sposobnosti. Vsem navedenim nazivom strokovnjakov na področju podatkov je skupno to, da delo s podatki zavzema večinski delež njihovega dela (OECD, 2015, str. 255).

Slika 3 prikazuje, kako lahko naziv podatkovni znanstvenik zajema širok spekter nazivov in vlog v procesu znanosti o podatkih prek življenjskega cikla vrednosti podatkov (angl. data value cycle) (OECD, 2015, str. 255). Nazivi vključujejo vloge, ki zbirajo podatke (skrbniki baz podatkov, skrbniki podatkov, vnosni referenti ali načrtovalci baz podatkov), analizirajo podatke s pomočjo analitike (statistiki, aktuarji, znanstveniki, analitiki), ter do določene mere takšni, ki sprejemajo odločitve na podlagi podatkov (menedžerji, inženirji) (OECD, 2015, str. 254).



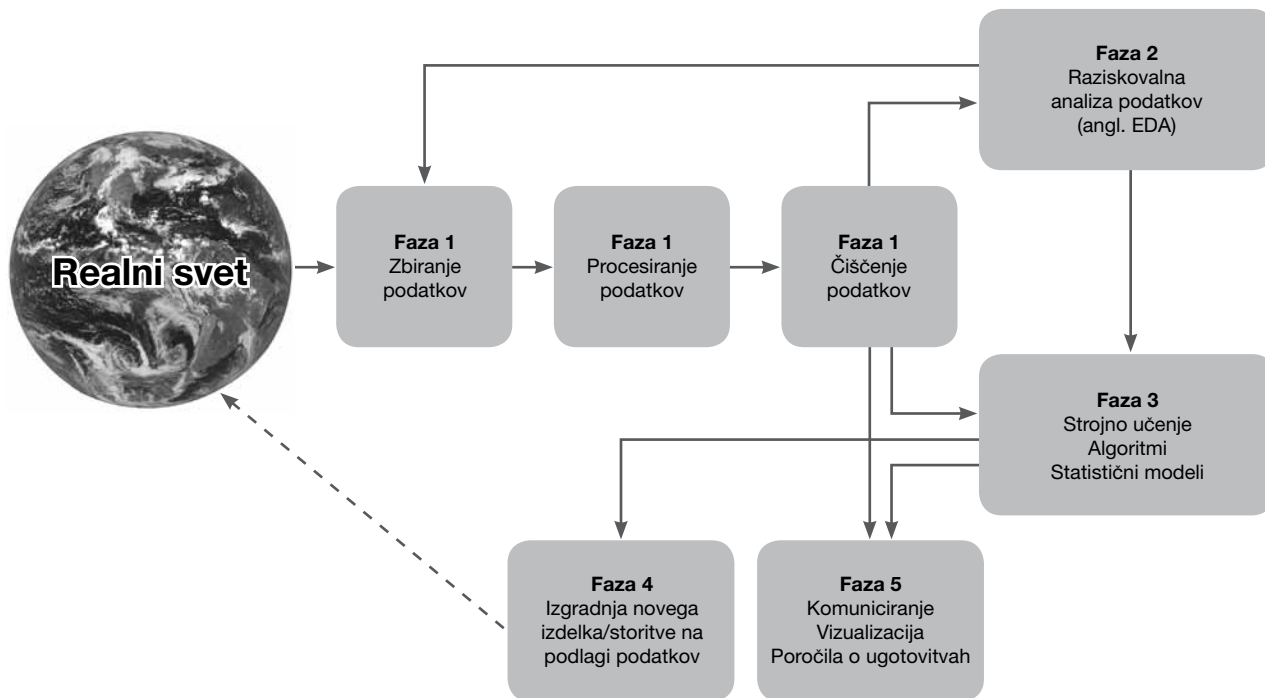
Slika 3: Faze v življenjskem ciklu vrednosti podatkov v povezavi s ključnimi tipi podatkovnih znanstvenikov
Vir: OECD, Data-driven innovation: Big Data for Growth and Well-Being, 2015, str. 255, slika 6.5.

Podatkovni znanstvenik je zato v tem prispevku opredeljen kot strokovnjak, ki se večino svojega časa ukvarja s podatki, pri čemer uporablja raznolika znanja in sposobnosti z več različnih znanstvenih področij z namenom, da iz surovih (masovnih) podatkov prek znanosti o podatkih pridobi dodano vrednost. Na podlagi napisanega je ključno, da ima podatkovni znanstvenik ustrezna znanja in sposobnosti, da lahko izpelje vse faze procesa znanosti o podatkih: od pridobivanja podatkov do končne vizualizacije ali novega izdelka/storitve na podlagi podatkov.

Za bolj podroben pregled in razumevanje področja dela, ki ga opravlja podatkovni znanstvenik, je v nadaljevanju predstavljen osnovni okvir za proces izvajanja znanosti o podatkih (prirejeno po O’Neill in Schutt, 2013, str. 36–41; Voulgaris, 2014, str. 133–149; Somohano, 2013; The field guide to data science, 2015, str. 29–33), ki je prikazan na sliki 4.

Kot je razvidno iz prikaza (slika 4), so faze med seboj ciklično prepletene. To pomeni, da si osnovne faze sicer sosledno sledijo, hkrati pa nekatere predstavljajo tudi korak, v katerem podatkovni znanstvenik sprejme odločitev, ali naj ponovi kakšno prejšnjo fazo ali nadaljuje z naslednjo.

Faza priprave podatkov je verjetno najbolj časovno zahtevna in najmanj zanimiva faza v procesu znanosti o podatkih. Vendar gre za zelo pomemben korak, saj ta faza predstavlja temelj za vse nadaljnje korake v procesu. Faza priprave podatkov vključuje zbiranje, procesiranje in čiščenje podatkov. Podatki v osnovi izhajajo iz realnega sveta, v katerem posamezniki in organizacije izvajamo svoje aktivnosti (uporabljamo Facebook, Twitter, opravljamo spletne nakupe, pošiljamo elektronsko pošto, pregledujemo spletne strani, opravljamo telefonske pogovore ipd.). S pomočjo shranjevanja teh aktivnosti v obliki podatkov podatkovni znanstvenik pridobi določen nabor surovih podatkov (transakcije, kliki na spletne povezave in dnevniške datoteke, podatki iz senzorjev, mobilnih telefonov, dokumenti, elektronska pošta, zapisi na družbenih medijih). Te podatke je treba najprej pregledati in prečistiti, da bodo primerni za analizo. Podatkovni znanstvenik lahko pri pripravi podatkov uporabi različne načine procesiranja podatkov: Hadoop Definition File System (v nadaljevanju HDFS) za shranjevanje podatkov za nadaljnje analize, Extract Transform Load (v nadaljevanju ETL) in MapReduce za branje podatkov, redukcijo dimenzij, vzorčenje, združevanje (angl. joining), strganje (angl.



Slika 4: **Proces izvajanja znanosti o podatkih**
Vir: C. O’Neill in R. Schutt, Doing Data Science, 2013, str. 41, slika 2-2.

scraping) ali mešetarjenje (angl. wrangling). Za procesiranje podatkov že obstajajo različna programska orodja, kljub temu pa mora podatkovni znanstvenik podatke pripraviti v .JSON ali v katerem drugem podobnem tipu podatkov. Če so podatki v popolnoma unikatnem tipu, mora podatkovni znanstvenik napisati lastni program za dostop in prestrukturiranje podatkov v obliko, ki bo razumljiva sistemom za branje podatkov. Pri branju velike količine podatkov je priporočljivo, da najprej pripravimo poskusno branje na relativno majhnem naboru podatkov. S tem podatkovni znanstvenik zagotovi, da bo končni nabor podatkov lahko uporaben za načrtovane analize. V sklopu priprave podatkov podatkovni znanstvenik izvede tudi čiščenje podatkov, ki zahteva določeno raven razumevanja podatkov. Pri čiščenju podatkov zapolni manjkajoče vrednosti, preveri relevantnost podatkov in izloči podatke, ki vključujejo napačne ali problematične podatke, izvede normalizacijo podatkov ter preveri neodvisnost podatkov. Čiščenje podatkov vključuje tudi obdelavo osamelcev (angl. outliers). Te lahko iz nabora podatkov odstranimo ali pa prilagodimo model, da ustreza obstoju osamelcev. Odločitev temelji na podlagi različnih faktorjev, kot so število osamelcev, podatkovni tip podatkov in občutljivost modela na njihov obstoj. Za čiščenje in transformacijo podatkov podatkovni znanstvenik uporablja različna orodja ali programske jezike, kot so Python, R skripte, poizvedbeni jezik SQL ali vse našteje. Priporočljivo je, da si podatkovni znanstvenik posamezne korake te faze shrani za primer, če jih bo treba ponoviti ali jih opisati v poročilu. Rezultat faze priprave podatkov je strukturirana oblika podatkov, pripravljena za nadaljnje analize.

Preden se podatkovni znanstvenik loti modeliranja, je potrebna izvedba t. i. raziskovalne analize podatkov (angl. Exploratory Data Analysis – v nadaljevanju EDA). Raziskovalna analiza podatkov je ključni del procesa izvajanja znanosti o podatkih in je primarno namenjena predvsem samemu podatkovnemu znanstveniku. Gre za sistematičen pregled podatkov s prikazom distribucij spremenljivk, transformacijo podatkov, iskanjem potencialnih povezav med spremenljivkami z uporabo razsevnih grafikonov in z generiranjem opisnih statistik za te spremenljivke (srednje vrednosti, mere razpršenosti, identifikacija osamelcev). Pri raziskovalni analizi ne gre le za uporabo orodij, temveč tudi za razumevanje odnosa podatkovnega znanstvenika do podatkov, ki

jih analizira. Če želi podatke razumeti, mora pridobiti intuicijo, razumeti oblike in povezati razumevanje procesa, kako so bili podatki pridobljeni, s samimi podatki. Na podlagi rezultatov raziskovalne analize podatkov lahko podatkovni znanstvenik ugotovi, da podatki dejansko niso ustrezni zaradi podvojenih, manjkajočih, neustreznih vrednosti, ali da podatki sploh niso bili zajeti ali pa so bili zajeti napačno. V tem primeru se mora podatkovni znanstvenik ponovno vrniti k viru podatkov in zbrati večjo količino podatkov ali več časa nameniti čiščenju podatkov. To je lahko iterativen proces, dokler podatki niso ustrezni in primerni za modeliranje.

V fazi učenja iz podatkov podatkovni znanstvenik pripravi model, ki predstavlja poskus razumevanja in predstavitve narave realnosti z določenega (matematičnega) vidika. Gre za umetno ustvarjen konstrukt, v katerem so odstranjene vse odvečne podrobnosti. Podatkovni znanstvenik oblikuje model z uporabo različnih orodij s področij statistike in strojnega učenja: opisne statistike in statističnega sklepanja, klasifikacije in segmentacije, regresijske analize in napovedovanja. Izbira modela je odvisna od vsebine problema, ki ga obravnava podatkovni znanstvenik. Lahko gre za klasifikacijski problem, napovedni problem ali osnovni opisni problem.

Podatkovni znanstvenik v zadnji fazi ugotovitve interpretira, pripravi vizualizacije in poročila ter rezultate na ustrezen način predstavi nadrejenim in sodelavcem ali jih objavi v publikaciji. Namen znanosti o podatkih je namreč določiti in razumeti, kaj vse se skriva pod površjem in kakšno uporabno vrednost lahko prinese do končnih uporabnikov. Proces znanosti o podatkih je ponavljajoč se razvojni proces, ki vključuje odkrivanje in učenje na podlagi podatkov. Vizualizacija vključuje grafično predstavbo pomena analiziranih podatkov na intuitiven, zanimiv in relevanten način do končnega uporabnika, ki je lahko tudi interaktiven. S pomočjo vizualizacije lahko podatkovni znanstvenik pridobi boljšo predstavbo, česa še ne ve, in lahko tako bolje razume omejitve modelov, vrednost podatkov in bolje obvladuje negotovost v podatkih. Cilj analize je alternativno lahko tudi izgradnja prototipa na podlagi analiziranih podatkov (angl. data product). Novi izdelek ali storitev, ki temelji na kombinaciji podatkov in algoritmov, je dodana vrednost organizaciji. Primeri takšnih izdelkov ali storitev na podlagi podatkov so klasifikator nezaželene elektronske pošte, algoritem

za rangiranje spletnih strani v spletnih iskalnikih z relevantnimi rezultati na podlagi spletnega iskanja, sistem za priporočanje (angl. recommendation system), mrežna statistika in grafikoni, ki jih LinkedIn prikazuje svojim uporabnikom, ali geografski informacijski sistem, kot je MapQuest, ki potrošnikom daje uporabne geografske informacije o določeni lokaciji. Tisto, kar razlikuje znanost o podatkih od statistike, je, da se takšen »podatkovni produkt« vgradi nazaj v realni svet, v katerem potrošniki interaktivno uporabljajo produkt, kar posledično generira več podatkov, ki podatkovnemu znanstveniku omogočajo izboljšave tega produkta.

3.2 Znanja in sposobnosti podatkovnih znanstvenikov

3.2.1 Znanja podatkovnih znanstvenikov

Veliko različnih tehnologij in tehnik je bilo razvitih in prilagojenih z namenom združevanja, manipuliranja, analiziranja in vizualizacije masovnih podatkov (Manyika idr., 2011, str. 27). Seznam znanj trenutno ne daje celostnega pogleda na znanja podatkovnih znanstvenikov, saj se metode in orodja neprestano razvijajo z namenom reševanja vedno novih problemov (Manyika idr., 2011, str. 27). Prav tako različni problemi zahtevajo uporabo različnih tehnik in tehnologij z različnih področij glede na naravo problema in cilje projekta. Pri opredelitvi znanj podatkovnih znanstvenikov so bila ta na podlagi literature, raziskav in izkušenj razporejena v skupine po naslednjih področjih: znanstvena metoda, programiranje, menedžment podatkov, baze podatkov, statistika, matematika, strojno učenje ter domenska znanja s pripadajočimi tehnikami. Iz nabora znanj so bile izključene opredelitev znanj, vezane na specifične programske rešitve (SPSS, SAS, Orange, RapidMiner, Weka, Tableau, Excel itd.), saj so te bolj tehnološko orodje, s katerim podatkovni znanstvenik izvede določeno fazo v znanosti o podatkih. Poleg tega bi to lahko pristransko vplivalo na rezultate, saj bi bili ti vezani na popularnost in dostopnost posameznega orodja.

Znanstvena metoda v najširšem smislu vključuje vse postopke in tehnike za objektivno raziskovanje pojavov (Toš in Hafner-Fink, 1998). Hayes (2014b) verjame, da znanstvena metoda predstavlja ključno vlogo v razumevanju katerih koli podatkov, ne glede na njihovo velikost, hitrost ali raznolikost. Podatki namreč ne »govorijo« sami zase, temveč jim pomen

dajo ljudje prek ustvarjanja, zbiranja in interpretacije podatkov. Ljudje pa so na žalost tudi vir (namerne ali nenamerne) pristranskosti, ki lahko poslabša kakovost podatkov (Hayes, 2014b). Načrtovanje poskusov/eksperimentov (angl. experimental design) je raziskovalna tehnika, ki se uporablja v vzročnem raziskovanju (angl. causal research) za vzpostavljanje vzročno-posledičnega odnosa med spremenljivkami (Malhotra, 2012, str. 221). Podatkovni znanstvenik naj bi v sklopu načrtovanja poskusov poznal koncepte neodvisnih, odvisnih in zunanjih spremenljivk, testnih enot in ključne razdelitve na poskusno in kontrolno skupino. Pri izvedbi poskusa podatkovni znanstvenik namreč določi testne enote in način, kako so te enote razdeljene na homogene podskupine, določi, katere neodvisne spremenljivke bo spreminjal, manipulira eno ali več neodvisnih spremenljivk in nato opazuje in meri učinke teh sprememb na odvisne spremenljivke, ob tem pa preverja vpliv zunanjih ali tujih spremenljivk (Malhotra, 2012, str. 222–223). Zagotavljanje ponovljivosti raziskav (angl. reproducible research) pomeni ključni koncept znanstvene metode. Vključuje koncepte in orodja, ki jih uporablja podatkovni znanstvenik, da lahko znanstvene ugotovitve objavi skupaj s podatki, viri podatkov, programsko kodo ter podrobnimi navodili za izvedbo analize podatkov z namenom, da je raziskavo mogoče ponoviti, bolje razumeti ali preveriti njeno veljavnost (Kuhn, 2015).

Z znanjem programiranja lahko podatkovni znanstvenik pokrije celotni proces izvajanja znanosti o podatkih – kadar koli lahko napiše program, ki pridobi podatke iz baze podatkov, zažene algoritme strojnega učenja na naboru podatkov (Ultimate skills checklist for your first data analyst job, 2015, str. 5), razvije produkt/storitev na podlagi podatkov ali pripravi vizualizacijo podatkov. Priporočljivo je znanje vsaj enega ali več programskih jezikov, ki so robustni, popularni in razširljivi – sploh pri velikem naboru podatkov (Voulgaris, 2014, str. 53). Priporočljivo je tudi, da ima podatkovni znanstvenik dober pregled nad področjem knjižnic in paketov, povezanih s programskimi jeziki, ki se najpogosteje uporabljajo za izvajanje znanosti o podatkih (Ultimate skills checklist for your first data analyst job, 2015, str. 5). Med programske jezike, ki jih najbolj pogosto uporablja več kot 90 odstotkov podatkovnih znanstvenikov, uvrščamo R, SAS in Python (Piatetsky, 2014). Voulgaris (2014, str. 54) omenja tudi Java, C+, C# in Perl,

ki so objektivno orientirani jeziki (angl. object-oriented languages), katerih prednost je v tem, da omogočajo enostavno ustvarjanje kompleksne programske kode. Proces znanosti o podatkih je mogoče izvajati tudi v drugih programskih jezikih: Julia, Scala, Stata, Hadoop programski jeziki (Pig Latin, HiveQL idr.), Java, Unix shell/awk/sed, MATLAB, C/C++, Perl, Octave, Ruby, Lisp/Clojure, F# itd. (Piatetsky, 2014). Znanja iz zalednega in čelnega programiranja se izkažejo za koristna predvsem pri implementaciji produkta/storitve na podlagi podatkov v produkcijsko okolje – uporabniško aplikacijo. Podatkovni znanstvenik naj bi predvsem imel osnovna znanja kot podlago za komunikacijo in usklajevanje analitične rešitve z zalednim in čelnim razvijalcem.

Podatkovni znanstvenik ima s svojim delovanjem in izvajanjem procesa znanosti o podatkih pregled, možnost in vpliv na obvladovanje podatkov, arhitekturo, varnost, povezovanje, shranjevanje in kakovost podatkov ter druge vidike t. i. menedžmenta podatkov. Z ustreznimi znanji s področja menedžmenta podatkov lahko podatkovni znanstvenik poveča učinkovitost in uspešnost izvajanja procesa znanosti o podatkih. Menedžment podatkov (angl. data management) vključuje in opisuje procese za načrtovanje, definiranje, kreiranje, pridobivanje, vzdrževanje, uporabo, arhiviranje, nadzor in integracijo podatkov (DAMA, 2014, str. 5). Po definiciji DAMA (2014, str. 10) se menedžment podatkov deli na več področij: obvladovanje podatkov (angl. data governance), menedžment podatkovne arhitekture (angl. data architecture management), razvoj in oblikovanje podatkov (angl. data modeling and design), shranjevanje podatkov (angl. data storage and operations), menedžment varnosti podatkov (angl. data security management), integracija in interoperabilnost podatkov (angl. data integration and interoperability), menedžment dokumentov in vsebine (angl. document and content management), menedžment matičnih in referenčnih podatkov (angl. reference and master data management), menedžment podatkovnih skladišč in poslovne inteligence (angl. data warehousing and business intelligence management), menedžment metapodatkov (angl. metadata management) ter menedžment kakovosti podatkov (angl. data quality management). Oblikovanje informacij (vizualizacija) – v sklopu znanj podatkovnega znanstvenika govorimo o podpodročju vizualizacije podatkov, in sicer o področju vizualizacije/oblikovanja

informacij (angl. information visualization/design). Znanja iz oblikovanja informacij lahko podatkovni znanstvenik uporablja v več različnih fazah izvajanja procesa znanosti o podatkih. V sklopu raziskovalne analize podatkov uporablja različne vizualizacijske tehnike z namenom razumevanja podatkov in njihovih zakonitosti ter identifikacije negotovosti v podatkih (npr. gruča primerov, osamelce, trende in relacije med spremenljivkami) (Leban, 2007, str. 2). Informacije oblikuje tudi z namenom razumevanja rešitev analize, domnev in algoritmov podatkovnega rudarjenja (kjer je to mogoče) ter za predstavitev rezultatov procesa znanosti o podatkih ali za kreiranje izdelka/storitve, ki temelji na podatkih. »Glavna prednost uporabe vizualizacije je njena interpretabilnost – odkrite zakonitosti lahko dejansko vidimo, zaradi česar je njihovo razumevanje neprimerno boljše.« (Leban, 2007, str. 2) Podatkovni znanstvenik naj bi zato imel znanja iz celostnega pristopa k oblikovanju informacij, poznavanja posameznih vizualizacijskih metod in njihovih zakonitosti, izbire ustrezne vizualizacijske metode in znanja iz oblikovanja interaktivnosti (razvoj interaktivnih rešitev, produktov in vmesnikov).

Prva faza v procesu znanosti o podatkih je ročno pridobivanje, shranjevanje in čiščenje podatkov v obliko, ki bo primerna za izvoz ali nadaljnje analize (Ultimate skills checklist for your first data analyst job, 2015, str. 12). Ta proces je v sklopu znanosti o podatkih znan kot mešetarjenje podatkov (angl. data wrangling, data munging, data scraping) (Ultimate skills checklist for your first data analyst job, 2015, str. 12). Gre za nalogo, ki podatkovnemu znanstveniku lahko zavzame od 50 do 80 odstotkov njegovega časa (Ultimate skills checklist for your first data analyst job, 2015, str. 12). Zaradi tega je pomembno, da ima podatkovni znanstvenik znanja, kako dostopati do podatkov, jih pridobiti, shraniti ter odpraviti nepopolnosti, za kar potrebuje znanja s področja baz podatkov. Med znanja s področja baz podatkov spadajo poznavanje sistemov baz podatkov, ki temeljijo na strukturiranih ali delno strukturiranih podatkih (centralni repozitorij za shranjevanje podatkov, katerih osnova je SQL), nestrukturiranih podatkih (baze podatkov, katerih osnova je NoSQL), masovnih in distribuiranih podatkih (Hadoop, MapReduce), poizvedbenih programskih jezikov SQL, HiveQL ter osnovna znanja iz systemske administracije.

Podatkovni znanstvenik naj bi imel osnovno znanje iz statistike ter poznavanje določenih kon-

ceptov in terminologije, ki jo uporabljajo statistiki (Granville, 2014, str. 4): metod vzorčenja, opisne statistike, verjetnostnih porazdelitev, statističnega preizkušanja domnev, redukcije dimenzij, analize časovnih vrst, prostorske statistike ipd. Pri tem je predvsem pomemben vidik znanja in razumevanja, kdaj je določena tehnika primeren ali neprimeren pristop k problemu (Ultimate skills checklist for your first data analyst job, 2015, str. 7).

Podatkovni znanstvenik naj bi bil sposoben prevesti besedne probleme v matematične izraze, reševati enačbe, manipulirati algebrične izraze in imel naj bi osnovno znanje iz teorije matrik (Ultimate skills checklist for your first data analyst job, 2015, str. 9; Granville 2014, str. 4). Prav tako naj bi imel znanja, kako narisati grafikone za različne tipe funkcij z razumevanjem odnosa med grafično funkcijo in njeno enačbo. Priporočljiva so tudi znanja odvodov in integralov, optimizacije in linearne algebre. Ta področja matematike so osnova za razumevanje strojnega učenja in učinkovitega manipuliranja podatkov v podatkovnih modelih (Ultimate skills checklist for your first data analyst job, 2015, str. 9).

Strojno učenje je poddomena računalništva (področja umetne inteligence), ki se ukvarja z zasnovo in razvojem algoritmov, ki omogočajo računalnikom razvoj akcij na podlagi empiričnih podatkov (Manyika idr., 2011, str. 29). Poudarek strojnega učenja je na avtomatiziranem učenju in prepoznavi kompleksnih vzorcev z namenom sprejemanja inteligentnih odločitev na podlagi podatkov (Manyika idr., 2011, str. 29). Strojno učenje je torej podatkovno intenzivni razvoj algoritmov (kot podatkovno rudarjenje) s poudarkom na prototipiranju algoritmov za produkcijsko okolje, za obdelavo velikih količin podatkov, na podlagi katerih je mogoče narediti napovedi (angl. predict), klasifikacijo (angl. classify), segmentacijo (angl. cluster) in/ali izračunati predloge za ukrepanje na podlagi obdelanih podatkov (Ultimate skills checklist for your first data analyst job, 2015, str. 10; O'Neill in Schutt, 2013, str. 52). Strojno učenje se ukvarja tudi z razvojem avtomatiziranih sistemov (prepoznavanje slik, govora, algoritmi za generiranje ponudb, angl. bidding algorithms, algoritmi za targetirano oglaševanje, angl. ad targeting algorithms), ki se samodejno osvežijo, neprestano preizkušajo, ponovno učijo in osvežujejo nabore podatkov za učenje, preverjajo veljavnost in izboljšujejo ali odkrivajo nova pravila. Poddomena strojnega učenja, zelo bli-

zu umetni inteligenci (angl. artificial intelligence), je poglobljeno učenje (angl. deep learning) (Granville, 2014). Za podatkovnega znanstvenika ni nujno, da ustvarja popolnoma nove algoritme za strojno učenje, vendar pa mora poznati najbolj pogoste algoritme in tehnike za strojno učenje, od zmanjšanja dimenzij (metoda glavnih komponent) do nadzorovanega (klasifikacija) in nenadzorovanega učenja (razvrščanje v skupine). Ni v celoti potrebno poznavanje teorije in podrobnosti implementacij v ozadju teh algoritmov. Je pa potrebno poznavanje prednosti in slabosti teh algoritmov, kot tudi, kdaj jih je smiselno uporabiti glede na kontekst problema ter kdaj ne (O'Neill in Schutt, 2013, str. 54; Ultimate skills checklist for your first data analyst job, 2015, str. 10).

Znanja poslovnega področja, dejavnosti ali domene, iz katere izhaja problem, so izjemne vrednosti in zelo nenadomestljiva (The field guide to data science, 2015, str. 96). Poslovna oziroma domenska znanja vključujejo poznavanje metod agilnega pristopa, pristopa Waterfall, razvoja izdelkov/storitev, razumevanje delovanja organizacije, poznavanje dejavnosti, poznavanje dobrih praks metodologij podatkovnega rudarjenja (CRISP-DM, SEMMA, DMAIC) ter vsa druga poslovna znanja (finance, trženje, trženjsko raziskovanje, logistika, razvoj izdelka itd.), ki so relevantna za organizacijo ali dejavnost (Voulgaris, 2014, str. 150). Omogočajo poglobljeno razumevanje podatkov in dejavnikov, ki vplivajo na analitični cilj, velikokrat pa so ključni diferenciator uspeha celotne ekipe, ki se ukvarja z znanostjo o podatkih (The field guide to data science, 2015, str. 96). Domenska znanja vplivajo na to, kako podatkovni znanstvenik izbira lastnosti, pripisuje podatke, izbira algoritme, in posredno vplivajo tudi na uspešnost projektov. Posameznik žal ne more biti domenski strokovnjak na vsakem področju. Zato se podatkovni znanstveniki pogosto obračajo na druge analitike, domenske strokovnjake ter druge sekundarne vire z namenom izgradnje razumevanja domenskega področja problema (The field guide to data science, 2015, str. 96).

3.2.2 Sposobnosti podatkovnih znanstvenikov

Znanja z različnih področij so pomembna, vendar niso dovolj. Znanost o podatkih zahteva bolj sistematično razmišljanje ter kombiniranje kreativnega pristopa k definiranju in reševanju problemov skupaj z obvladovanjem časa. Podatkovni znanstvenik je posameznik, ki ga označuje nabor specifičnih

značilnosti, sposobnosti in načina razmišljanja, ne samo nabor znanj (Voulgaris, 2014, str. 37).

Podatkovni znanstvenik naj bi bil predvsem radeveden glede stvari, ki jih opazuje, kot so vzorci in odnosi ter razmerja med različnimi značilnostmi (Voulgaris, 2014, str. 38). Radevednost je ključna, da lahko podatkovni znanstvenik razstavi problem in razišče odnose med podatki, ki na prvi pogled delujejo nepovezani (The field guide to data science, 2015, str. 42). Radevednost dopolnjujejo disciplina, analitične sposobnosti in sposobnosti reševanja problemov. To vključuje vse – od želje po raziskovanju in razčlenitvi problema do zelo jasno definiranega nabora domnev, ki jih je mogoče preveriti (Lorica, Howard in Dumbill, 2012). Podatkovni znanstveniki rešujejo probleme uporabnikov podatkov. Vendar preden lahko rešijo problem, ga je treba ustrezno identificirati, kar pa ni vedno lahko (Stanton, 2013, str. 14). Za pristop k problemu in reševanju problemov sta ključni tudi eksperimentiranje in kreativnost – sposobnost pogleda na problem na različne, kreativne načine, ki v preteklosti niso še bili uporabljeni v takšnem kontekstu (angl. thinking outside the box) (Chordas, 2014, str. 24; Lorica, Howard in Dumbill, 2012; The field guide to data science, 2015, str. 42). Pomembna sposobnost sta tudi fleksibilnost in osredotočenost na cilj, ko je podatkovni znanstvenik sposoben premagati napake, opustiti idejo, ki ne deluje, se iz tega nekaj naučiti in poskusiti z novim pristopom. Znanost o podatkih je namreč serija »slepih ulic«, dokler prava pot ni identificirana. To zahteva unikaten set osebnostnih lastnosti – potrpežljivost in vztrajnost (The field guide to data science, 2015, str. 42).

Da podatkovni znanstvenik razume svojo vlogo in pomen, so pomembne tudi sposobnosti, povezane z njegovo podjetno naravnostjo, ki vključujejo poslovni čut, prebrisanost (angl. cleverness) in vztrajnost (Granville, 2014, str. 3; Chordas, 2014, str. 23; Lorica, Howard in Dumbill, 2012). Ključna sta tudi sposobnost sprejemanja odločitev in pogum – sposobnost podatkovnega znanstvenika, da izrazi svoje mnenje, poišče rešitev ter prepriča in motivira menedžerje v smeri prave rešitve, včasih tudi v nasprotju z njihovo voljo, v dobro organizacije, uporabnikov ali deležnikov (Granville, 2014, str. 4). Granville (2014, str. 4) omenja, da bi podatkovni znanstvenik moral biti tudi strateg, tako v poslovnem smislu kot v tem, da je sposoben razviti strategijo zbiranja podatkov z namenom pridobiti podlago za odločitve,

ki omogočajo poslovni učinek. Ko podatkovni znanstvenik razvije razumevanje domenskega znanja, mora imeti sposobnost predstave, kako se podatki prenašajo preko različnih sistemov in uporabnikov. Pri tem, opozarja Stanton (2013, str. 6), je potrebno, da podatkovni znanstvenik dovolj pozornosti nameni kakovosti. Ne glede na nabor podatkov, ki jih imamo, ti ne bodo nikoli popolni. Podatkovni znanstveniki morajo poznati omejitve podatkov, s katerimi delajo, znati morajo kvantificirati njihovo natančnost in na podlagi analize podati predloge za izboljšanje kakovosti podatkov v prihodnje (Stanton, 2013, str. 6). Zato je sposobnost načrtovanja ključni vidik znanosti o podatkih, saj obstajajo različni načini, kako se lotiti iste naloge, ki pa lahko imajo občutno drugačno porabo virov (Voulgaris, 2014, str. 27).

Sposobnosti in povezavi s področjem vodenja projektov in vodenja na splošno so sposobnosti, ki bi jih posameznik moral imeti zaradi narave dela (predlaganje izboljšav, razvijanje strategij, komuniciranje z naročniki, vodenje projektov ipd.) in dejstva, da podatkovni znanstvenik ni le operativni izvajalec, temveč se pojavlja tudi na različnih vodstvenih položajih ali pa kot vodja oddelka, tima (OECD, 2015, str. 255). Podatkovni znanstvenik po navadi deluje v sklopu raznolike ekipe strokovnjakov z različnih področij (odvisno od dejavnosti). Zelo redko podatkovni znanstvenik dela popolnoma ločeno in samostojno v daljšem časovnem obdobju, saj se za reševanje problema poveže s strokovnjaki, ki so na področju, s katerega izhaja problem, bolj izkušeni oziroma imajo več znanja. Zato je pomembno, da ima podatkovni znanstvenik sposobnost dela v timu. Podatkovni znanstvenik mora biti fleksibilen in imeti sposobnost hitrega prilagajanja novemu poslovnemu področju, novim članov ekipe ali novim programskim orodjem (Voulgaris, 2014, str. 27).

Ker imajo podatkovni znanstveniki po navadi poglobljena znanja iz vsaj enega znanstvenega področja (Lorica, Howard in Dumbill, 2012), kritično sposobnost podatkovnega znanstvenika predstavlja prevajanje med tehničnimi izrazi računalništva in statistike ter slovarjem domenskega znanja menedžmenta. Podatkovni znanstvenik mora zato imeti dobre komunikacijske sposobnosti. Pri tem pride do izraza predvsem sposobnost pripovedovanja zgodb (angl. storytelling), tj. sposobnost z uporabo podatkov predstaviti zgodbo in jo učinkovito prenesti različnim deležnikom (Lorica, Howard in Dumbill,

2012; Stanton, 2013, str. 5). Prednost za podatkovnega znanstvenika je, če ima poleg odličnih komunikacijskih sposobnosti tudi občutek za umetnost in prakso vizualizacije, kar pomeni, da je sposoben smiselno premostiti prepad med človekom in računalnikom s posredovanjem analitičnih dognanj (Lorica, Howard in Dumbill, 2012; Stanton, 2013, str. 6).

Stanton (2013, str. 6) kot pomembno sposobnost podatkovnega znanstvenika omenja tudi sposobnost biti etičen oziroma razmišljati etično. Če so podatki dovolj pomembni, da se jih odločimo zbirati, so po navadi dovolj pomembni, da lahko vplivajo na človeška življenja. Podatkovni znanstveniki morajo razumeti etično odgovornost, povezano z zasebnostjo, in morajo biti sposobni ustrezno predstaviti omejitve z namenom preprečiti zlorabo podatkov ali rezultatov analiz.

Za podatkovnega znanstvenika je ne nazadnje pomembno tudi, da goji strast učenja novih stvari in do dela, ki ga opravlja, ter da ima sposobnost »zaznavanja« podatkov (Granville, 2014, str. 4). Zaradi hitrega napredka tehnologij na področju masovnih podatkov in znanosti o podatkih mora biti podatkovni znanstvenik sposoben hitrega učenja ter hitrega sprejetja novih metod in orodij (Voulgaris, 2014, str. 27). Radovednost, želja po raziskovanju, učenju, strast in vztrajnost se zrcalijo na vseh vidikih življenja podatkovnega znanstvenika (Granville, 2014, str. 4; Lorica, Howard in Dumbill, 2012).

4 RAZISKAVA O ZNANJH IN SPOSOBNOSTIH PODATKOVNIH ZNANSTVENIKOV V SLOVENIJI

4.1 Metodologija

V raziskavi je sodelovalo 92 posameznikov iz Slovenije, ki se večino svojega časa ukvarjajo s podatki. Vprašalnik, pripravljen na podlagi pregleda teoretičnih izhodišč ter že izvedenih raziskav (Harris, Murphy in Vaisman, 2013; Hayes, 2015a; Swan, 2008), je bil sestavljen iz treh skupin vprašanj, ki se nanašajo na ugotovitve iz prejšnjih razdelkov. Prva skupina vprašanj se je nanašala na posamezne dimenzije masovnih podatkov: volumen, raznolikost, hitrost in vrednost (Piatetsky, 2015; The Emerging Big Returns on Big Data, 2013; Russom, 2011; Big Data Executive Survey, 2012). Druga skupina vprašanj se je nanašala na samooceno znanj ter oceno pomembnosti znanj in sposobnosti, predstavljenih v prejšnjem razdelku, tretja skupina vprašanj pa na pretekle izkušnje

in pridobivanje znanj in sposobnosti prek različnih načinov izobraževanja (Harris, Murphy in Vaisman, 2013). Zadnji sklop vprašanj je vključeval demografska vprašanja – spol, starost, stopnja in smer izobrazbe. Za zbiranje podatkov je bila uporabljena metoda spletnega anketiranja s pomočjo strukturirane vprašalnika. V vprašalniku so bile uporabljene različne merske lestvice: nominalna, ordinalna in intervalna. Pri vprašanjih v drugem sklopu je bila uporabljena petstopenjska Likertova lestvica. Pri določenih vprašanjih je bila dodatno omogočena možnost »ne vem«.

Vzorčni okvir v tem primeru ne obstaja, saj ni popolnega seznama posameznikov, ki se v Sloveniji večino svojega časa ukvarjajo s podatki oziroma s katerim od naslednjih področij: analitika, statistika, matematika, programiranje, menedžment podatkov, raziskovanje ali pa so vodje takšnih ekip. K izpolnitvi vprašalnika so na podlagi dostopnih informacij, preteklih izkušenj, sodelovanj in poznanstev bili povabljeni posamezniki iz različnih organizacij: In516ht, d. o. o., Petrol, d. d., Institut Jožef Stefan, Studio Moderna, d. o. o., Spar Slovenija, d. o. o., Si.Mobil, d. d., Zavarovalnica Triglav, d. d., ADD, d. o. o., Zavarovalnica Maribor, d. d., Mercator, d. d., Kendu, d. o. o., Ekipa2, d. o. o., Javni holding Ljubljana, d. o. o., D.Labs, d. o. o., Adriatic Slovenica, d. d., Nova ljubljanska banka, d. d., Zavod za pokojninsko in invalidsko zavarovanje Slovenije, IBM Slovenija, d. o. o., Ektimo, d. o. o., Revelo, d. o. o., Hekovnik, Arhea Solutio, d. o. o., Valicon, d. o. o., Inštitut za raziskovanje trga in medijev Mediana, d. o. o., Droga Kolinska, d. d., itd. Vprašalnik je bil objavljen tudi na skupini Big Data Developers in Data Science Slovenia ter na Facebook strani Udomačena Statistika. Povabilu k raziskavi je bila dodana tudi prošnja za posredovanje vprašalnika drugim primernim posameznikom. Metoda vzorčenja je bilo namensko priložnostno vzorčenje, saj so bili k izpolnjevanju vprašalnika povabljeni le posamezniki iz Slovenije, ki so ustrezali predhodno določenim kriterijem (ukvarjanje s podatki oziroma ukvarjanje z vnaprej določenim področjem dela). Izpolnjevanje vprašalnika je potekalo od 26. 4. do 21. 5. 2016. Vprašalnik je v celoti izpolnilo 94 oseb; 47 oseb pa je vprašalnik izpolnilo le delno, zato so bili izločeni iz analize. Rezultati ankete so bili ustrezno zakodirani v podatkovno bazo s 94 enotami in 126 spremenljivkami. Pri pregledu podatkov se je izkazalo, da dve enoti nista bili primerni

za analizo, saj sta vsebovali preveliko število neodgovorjenih vprašanj. Končni nabor enot za analizo je zajemal 92 enot. Podatki so bili zbrani s pomočjo spletnega orodja 1ka.si. Za analizo podatkov in vizualizacijo rezultatov sta bila uporabljena SPSS Statistics, verzija 21, in MS Excel 2010. Pri analizi podatkov so bile uporabljene metode opisnih statistik za prikaz rezultatov in značilnosti vzorca, parametrični in neparametrični testi za preverjanje domnev ter metode multivariatne analize (razvrščanje v skupine, algoritem K-means) za identifikacijo skupin podatkovnih znanstvenikov glede na samooceno znanj.

V raziskavi je sodelovalo 92 anketirancev; 59,8 odstotka jih je bilo moškega, 40,2 odstotka pa ženskega spola. Prevladovali so anketiranci moškega spola. Največji delež anketirancev je pripadalo starostni skupini od 26 do 35 let (51,1 odstotka). Mlajši od 18 let ni bil nihče, 1,1 odstotka anketirancev je bilo v starostni skupini od 18 do 25 let, 51,1 odstotka v starostni skupini od 26 do 35 let, 35,9 odstotka v starostni skupini od 36 do 45 let, 10,9 odstotka v starostni skupini od 46 do 55 let in 1,1 odstotka v starostni skupini 56 let in več. Največji delež anketirancev je imel univerzitetno izobrazbo (50 odstotkov). Sledili so anketiranci z magisterijem, doktoratom ali specializacijo (28,3 odstotka) ter anketiranci s poklicno ali štiriletno srednjo šolo (13 odstotkov). Najmanj je bilo anketirancev z višjo ali visoko šolo (8,7 odstotka). V vzorec niso bili zajeti anketiranci s stopnjo izobrazbe osnovna šola ali manj. Največji delež anketirancev je kot svojo prevladujočo smer izobrazbe navedlo računalništvo (26,1 odstotka), sledita ekonomija in poslovne vede (19,6 odstotka), splošno družboslovje (15,2 odstotka), druge naravoslovne ali tehnične vede (14,1 odstotka), statistika (9,8 odstotka) in matematika (8,7 odstotka). En anketiranec je kot smer izobrazbe navedel fiziko (1,1 odstotka).

Pri razvrščanju v skupine so bile kot relevantne spremenljivke vzeti odgovori na vprašanja, povezana s samooceno znanj. Udeleženci so pri petem vprašanju ocenjevali svojo stopnjo znanja z različnih področij znanj. Pri tem je bila uporabljena intervalna lestvica od 1 – Ne poznam (ne uporabljam/ne ustreza mojemu področju dela), 2 – Osnovno poznavanje (osnovno znanje, fokus je na izobraževanju), 3 – Začetnik (znanje pripravnika, fokus je na pridobivanju izkušenj na praktičnih primerih), 4 – Srednji nivo (samostojna kompetentna uporaba, fokus je na izboljšanju znanja) do 5 – Napredni nivo (poglobljena

znanja in kompetence, fokus je na reševanju strokovnih problemov). Pri preverjanju korelacij med spremenljivkami je bilo ugotovljeno, da sta edini spremenljivki, ki visoko korelirata med seboj, nadzorovano strojno učenje in nenadzorovano strojno učenje (Pearsonov korelacijski koeficient = 0,924). Korelacija med vsemi ostalimi spremenljivkami je bila manjša kot 0,9. Zaradi navedenega je bila iz nadaljnje analize izpuščena spremenljivka nenadzorovano učenje. Preostale spremenljivke (30) so bile še vedno dobra podlaga za razvrščanje v skupine. Cronbach alpha za preverbo notranje konsistentnosti za 30 spremenljivk in velikost vzorca $n = 83$ (toliko udeležencev je v celoti odgovorilo na vsa vprašanja, povezana s samooceno znanj) je znašal 0,931, kar pomeni visoko stopnjo notranje konsistentnosti za spremenljivke, merjene na tej lestvici, in za ta specifični vzorec.

Ker je šlo za majhen vzorec in ker končno število skupin ni bilo poznano vnaprej, je bilo najprej izvedeno hierarhično razvrščanje v skupine. Kot mero podobnosti oziroma različnosti med skupinami je bila uporabljena kvadratna evklidska razdalja s standardiziranimi spremenljivkami. Za metodo razvrščanja v skupine je bila izbrana Wardova metoda. Na podlagi hierarhičnega razvrščanja v skupine, pregleda dendograma in izračuna VRC Calinski in Harabasz sta bili za nadaljevanje analize upoštevani razvrstitvi v štiri ali pet skupin. V naslednjem koraku je bilo izvedeno nehierarhično razvrščanje v štiri in pet skupin s K-means algoritmom. Pri uporabi K-means algoritma so bile kot izhodiščne vrednosti upoštewane povprečne ocene, pridobljene iz hierarhičnega razvrščanja v štiri in pet skupin. Na podlagi kvalitativne primerjave razvrščanja v štiri in pet skupin je bilo ugotovljeno, da je razvrščanje v pet skupin identificiralo določeno podmnožico enot znotraj skupine C1 pri razvrščanju v štiri skupine, ki predstavlja drugačen in zanimiv nabor znanj, kot skupina C2 pri razvrščanju v pet skupin. Z razvrstitvijo v pet skupin se je tudi zmanjšala variabilnost znotraj skupin, vendar se je na ta račun povečalo število osamelcev v skupini C2. Zaradi vsega navedenega je bila kot najbolj primerna izbrana razvrstitev v pet skupin.

4.2 Diskusija

Na podlagi vzorčnih podatkov in rezultatov analize podatkov v nadaljevanju predstavljamo ugotovitve po posameznih raziskovalnih vprašanjih.

1. Ali se podatkovni znanstveniki v Sloveniji ukvarjajo z masovnimi podatki in kako se to zrcali skozi različne dimenzije masovnih podatkov?

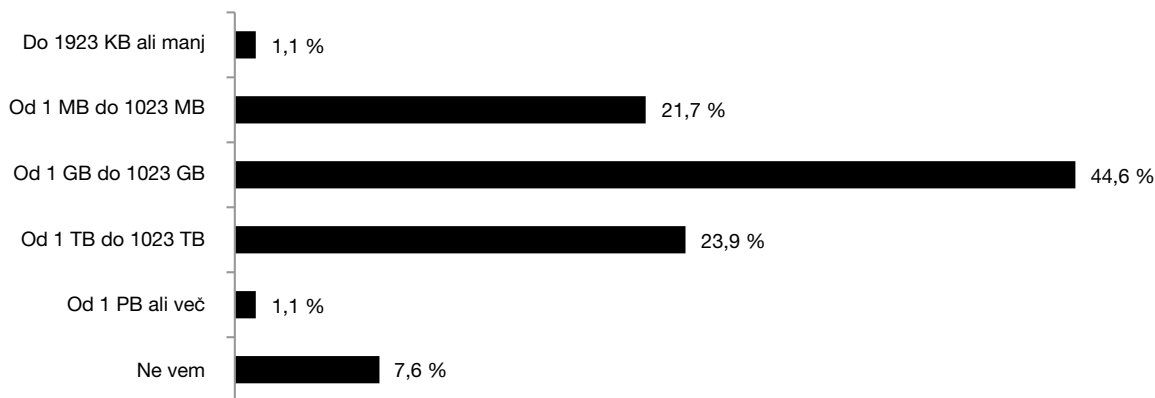
Analiza rezultatov v Sloveniji je interpretirana in analizirana prek primerjave z rezultati več raziskav, ki so se nanašale na različne dimenzije masovnih podatkov. Dimenzija količine/volumna podatkov je primerjana z rezultati raziskave Kdnuggets.com (Piatetsky, 2015), dimenzija raznolikosti z rezultati raziskave The Emerging Big Returns on Big Data (2013), dimenzija hitrosti z rezultati raziskave TDWI Big Data Analytics (Russom, 2011, str. 19) ter dimenzija vrednosti z rezultati raziskave Big Data Executive Survey (2012, str. 5).

Na podlagi rezultatov raziskave KDNuggets.com (Piatetsky, 2015) glede največjega obdelanega nabora podatkov po geografskih področjih največji delež posameznikov, ki obdeluje podatke v TB ali PB, prihaja iz Združenih držav Amerike in Kanade

(26,6 odstotka). Evropa je na četrtem mestu z 20,7 odstotka posameznikov, ki so obdelali TB podatkov ali več. V Evropi so drugače kot največji obdelan nabor podatkov največkrat (60 odstotkov) izbrali podatke v velikosti GB, manj kot 20 odstotkov pa jih obdeluje podatke velikosti MB.

Rezultati iz raziskave v Sloveniji (slika 5) so pokazali, da je približno 25 odstotkov udeležencev kot največji volumen podatkov, s katerim so se ukvarjali, izbralo TB ali PB, kar je v primerjavi z raziskavo KDNuggets.com (Piatetsky, 2015) bolj na ravni Združenih držav Amerike in Kanade. Udeleženci raziskave v Sloveniji so kot največji volumen podatkov največkrat (44,6 odstotka) izbrali podatke od 1 do 2023 GB. Rezultat je sicer nižji kot delež v raziskavi Kdnuggets.com, vendar še vedno lahko sprejmemo sklep, da se udeleženci raziskave v Sloveniji največ ukvarjajo z obdelavo podatkov velikosti GB.

Prosim, označite, kakšna je bila največja količina/volumen podatkov, s katero ste se vi osebno do sedaj ukvarjali (pridobivanje, obdelava, analiza, odločanje). (n = 92)



Slika 5: Največja obdelana količina/volumen podatkov (n = 92)

V raziskavi The Emerging Big Returns on Big Data (2013, str. 19) so ugotovili, da je v organizacijah v Evropi povprečno ocenjeni delež strukturiranih podatkov 50 odstotkov, delno strukturiranih je 25 odstotkov, 25 odstotkov pa je nestrukturiranih podatkov. Povprečno ocenjeni delež nestrukturiranih podatkov je najvišji v azijsko-pacifiški regiji, kjer znaša 34 odstotkov. Pričakovalo se je, da bo raziskava v Sloveniji pokazala podobne rezultate, in sicer da se bo približno 25 odstotkov udeležencev že ukvarjalo z nestrukturiranimi podatki.

Rezultati raziskave v Sloveniji (tabela 1) so pokazali, da se je 100 odstotkov udeležencev že ukvarjalo s strukturiranimi podatki, kar je pričakovano, saj so v raziskavi sodelovali posamezniki, ki se ukvarjajo s podatki. Bolj zanimiv je podatek, da se je 50 odstotkov udeležencev raziskave že ukvarjalo z nestrukturiranimi podatki, kar je višje kot pričakovano. Kot nestrukturirani podatki so se upoštevali vsi nestrukturirani podatki (besedilo, avdio, video, slike), generirani s strani človeka.

Tabela 1: Raznolikost podatkov glede na dimenzijo (n = 92)

Dimenzija	Delež (v %)
Podatki, generirani s strani naprav	96,7
Podatki, generirani s strani človeka	79,3
Notranji viri	97,8
Zunanji viri	47,8
Strukturirani podatki	100,0
Nestrukturirani podatki	50,0

Na podlagi raziskave *The Emerging Big Returns on Big Data* (2013, str. 20) so ugotovili, da je v organizacijah v Evropi povprečno ocenjeni delež podatkov, pridobljenih iz notranjih virov, 68 odstotkov, iz zunanjih virov pa 32 odstotkov. Povprečno ocenjeni delež podatkov, pridobljenih iz zunanjih virov, je najvišji v azijsko-pacifiški regiji, kjer znaša 38 odstotkov. Rezultati raziskave v Sloveniji so pokazali, da se je 97,8 odstotka udeležencev že ukvarjalo s podatki iz notranjih virov, kar je pričakovano, saj so v raziskavi sodelovali posamezniki, ki se ukvarjajo s podatki, ki večinoma izvirajo iz notranjih virov organizacije. S podatki iz zunanjih virov pa se je srečalo že skoraj 48 odstotkov udeležencev raziskave, kar je višje kot 32 odstotkov iz raziskave *The Emerging Big Returns on Big Data* (2013, str. 20). Zanimivo je, da obstaja določen delež posameznikov (2,2 odstotka), ki se ukvarja s podatki izključno iz zunanjih virov. Kot podatke iz zunanjih virov smo upoštevali vse strukturirane in nestrukturirane podatke (besedilo, avdio, video, slike), generirane s strani človeka ali naprave, ki so pridobljeni iz zunanjih virov organizacije.

Rezultati obeh raziskav sicer niso v celoti primerljivi, saj so v raziskavi *The Emerging Big Returns on Big Data* (2013) spraševali po stanju tipov/virov podatkov v organizacijah, v raziskavi v Sloveniji pa smo spraševal posameznike o tem, ali so se že srečali z različnimi tipi, viri podatkov. Vseeno je bila raziskava *The Emerging Big Returns on Big Data* (2013) uporabljena kot možna primerjava stanja uporabe različnih tipov/virov podatkov. Razlog za razliko v primerjavi z raziskavo, izvedeno v tujini, je lahko tudi izbor (namensko priložnostno vzorčenje) in velikost vzorca (92 enot), vključenega v raziskavo v Sloveniji.

Na podlagi raziskave *TDWI Big Data Analytics* (Russom, 2011, str. 19) štiri odstotke analiz v orga-

nizacijah opravljajo, izvajajo ali ponovno izvajajo v realnem času, štiri odstotke na nivoju ure, pet odstotkov vsakih nekaj ur, 24 odstotkov dnevno, 14 odstotkov tedensko, 35 odstotkov mesečno in 15 odstotkov letno. Delež udeležencev, ki so v raziskavi v Sloveniji označili, da so se že ukvarjali s podatki v realnem času, je bil 47,8 odstotka. Vseeno je pred primerjavo podatkov med raziskavama treba upoštevati, da je šlo za drugačen vzorec in da obstaja možnost, da so udeleženci raziskave v Sloveniji neustrezno razumeli definicijo podatkov v realnem času, kar predstavlja tudi pomembno omejitev raziskave.

Na podlagi raziskave *Big Data Executive Survey* (2012, str. 5) organizacije uporabljajo masovne podatke za širok nabor namenov. Kot dve najbolj izpostavljeni prednosti uporabe masovnih podatkov so največkrat izbrali boljše odločanje na podlagi dejstev (22 odstotkov) in izboljšanje izkušnje potrošnika/uporabnika (22 odstotkov). Sledi povečanje prodaje/prihodkov (15 odstotkov), inovacije na področju izdelkov in storitev (11 odstotkov), zmanjšanje tveganja (11 odstotkov), boljša kakovost izdelkov in storitev (10 odstotkov) ter bolj učinkovito izvajanje procesov (10 odstotkov). Rezultati raziskave v Sloveniji so pokazali (slika 6), da so udeleženci raziskave kot glavno korist, ki jo dosegajo z delom s podatki, prav tako izbrali boljše odločanje na podlagi dejstev (82,6 odstotka). Sledi izboljšanje izkušnje potrošnika/uporabnika oziroma boljše razumevanje potrošnika (72,2 odstotka), kar se ujema z rezultati iz zgornje raziskave.

Razlike se pojavijo šele pri drugih koristih, saj so udeleženci v Sloveniji kot tretjo korist izbrali bolj učinkovito izvajanje procesov, načina dela, operacij (66,3 odstotka) in povečanje prodaje/prihodkov (66,3 odstotka), medtem ko se je možnost bolj učinkovito izvajanje procesov v zgornji raziskavi pojavilo šele na zadnjih mestih. V Sloveniji so najmanjkrat izbrali inovacije na področju izdelkov in storitev (39,1 odstotka), medtem ko je ta opcija v zgornji raziskavi bila med prvimi štirimi.

Rezultati raziskave so pokazali, da se določeni posamezniki v Sloveniji z vidika dimenzije volumna (TB ali več), raznolikosti (vse dimenzije podatkov) in vrednosti podatkov (boljše odločitve na podlagi dejstev) dejansko ukvarjajo z masovnimi podatki v ožjem smislu. Hkrati pa je bilo z analizo samoocen znanj (v nadaljevanju) ugotovljeno, da so ravno znanja s področja tehnologije masovnih podatkov (masovni in distribuirani podatki, sistemi baz podat-

Prosim, označite, s kakšnimi nameni se vi osebno ukvarjate z obdelavo podatkov oziroma katere otipljive koristi menite, da dosegate preko dela s podatki. (n = 92)



Slika 6: Vrednost podatkov (n = 92)

kov – baze podatkov NoSQL) v povprečju najslabše ocenjena (povprečna ocena je bila okrog 2 – osnove). Zanimivo je bilo, da so tudi z vidika pomembnosti znanj omenjena znanja iz tehnologije masovnih podatkov slabo ocenjena (povprečna ocena pomembnosti je bila okrog 2). Sklepamo lahko, da se v Sloveniji posamezniki ukvarjajo z masovnimi podatki v ožjem smislu, vendar pri tem ne uporabljajo tehnologij masovnih podatkov oziroma teh tehnologij še ne uporabljajo v tolikšni meri, kot so že sprejete v tujini.

2. Kakšna so dejanska znanja podatkovnih znanstvenikov v Sloveniji ter katera znanja in sposobnosti so pomembna pri njihovem delu?

Analiza rezultatov v Sloveniji je delno interpretirana in analizirana preko primerjave z rezultati raziskave Hayesa (2015a). Raziskava Hayesa (Hayes, 2015a, str. 2–4) o znanjih in sposobnostih podatkovnih znanstvenikov ter delovanju v timih je pokazala, da so sodelujoči v povprečju izrazili višjo stopnjo samoocene na naslednjih področjih: komunikacija, strukturirani podatki, podatkovno rudarjenje, znanost/znanstvena metoda, matematika, menedžment projektov, menedžment podatkov ter statistika in statistično modeliranje. V povprečju pa so nižjo stopnjo samoocene dodelili področjem: sistemska administracija, čelno in zaledno programiranje, procesiranje naravnega jezika (NLP), masovni in distribuirani podatki ter menedžment podatkov v oblaku (Hayes, 2015a, str. 2).

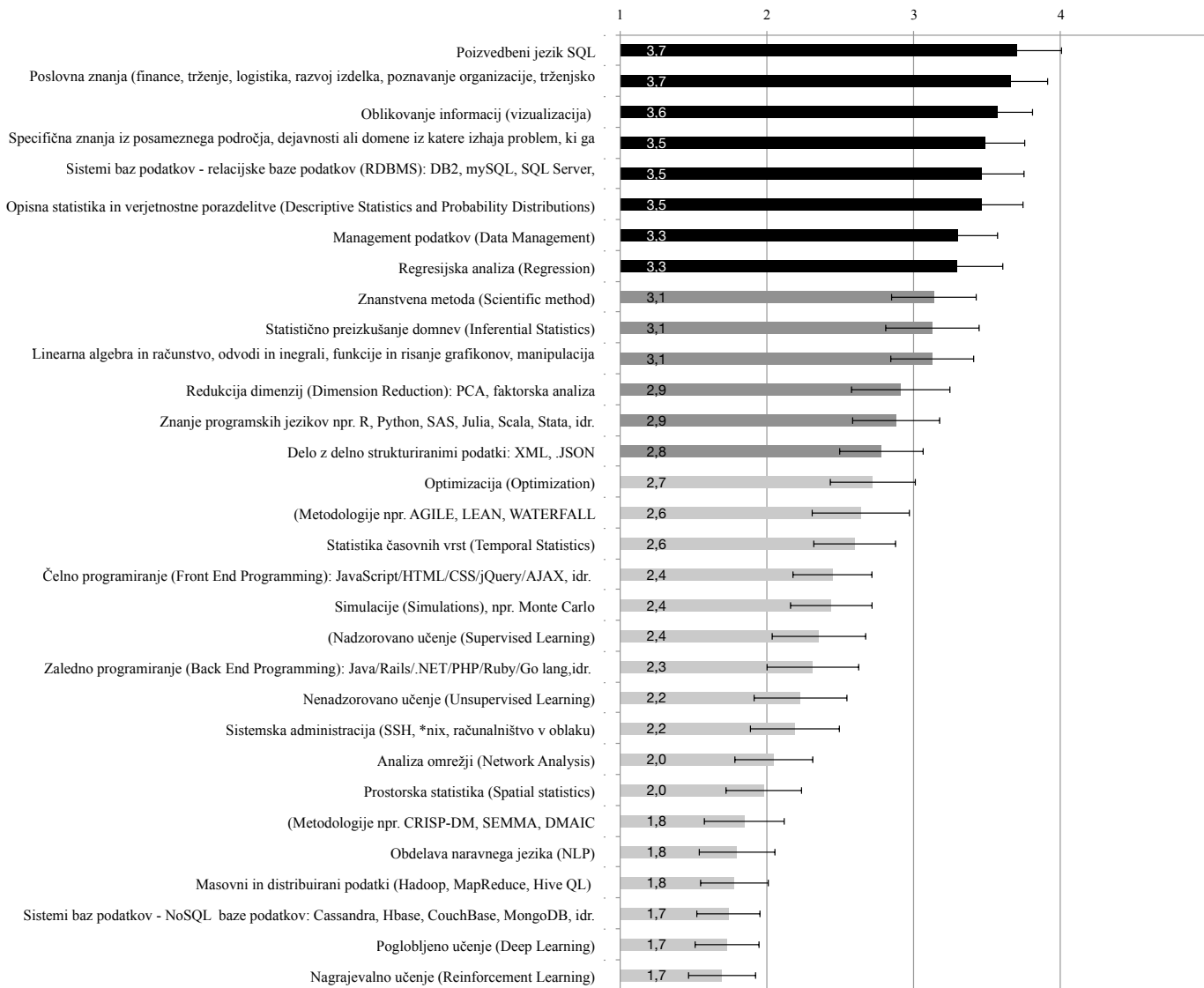
Na podlagi podatkov iz raziskave v Sloveniji je bilo ugotovljeno, da so posamezniki v Sloveniji v povprečju najvišjo samooceno znanj (slika 7) dodelili znanjem s področij baz podatkov (SQL, relacijske baze podatkov, menedžment podatkov), statistike (opisna statistika in verjetnostne porazdelitve ter regresija), domenskih znanj (poslovna znanja, specifična znanja s področja, s katerega izhaja problem) ter oblikovanja informacij. Zanimivo je, da so v sklopu posameznikov, ki se večino svojega časa ukvarjajo s podatki, med najbolj pomembnimi (slika 8) poslovna znanja, oblikovanje informacij in specifična znanja s posameznega področja. Pričakovati bi bilo, da bodo pomembna predvsem znanja iz statistike, baz podatkov ali druga znanja. Razlog verjetno leži v tem, da je rezultate analiz obdelave podatkov treba predstaviti jasno, enostavno in vizualno privlačno, da bodo razumljivi tudi ostalim deležnikom, in seveda z namenom, da prinašajo poslovno vrednost, za kar pa so potrebna poslovna znanja.

V povprečju pa slabo poznajo (slika 7) področja: metodologije strojnega učenja, nagrajevalno in poglobljeno učenje, obdelava naravnega jezika ter tehnologije masovnih podatkov. Razlog je verjetno v tem, da ta znanja pri njihovem delu trenutno niso pomembna, saj so ta področja znanj dobila tudi najnižjo povprečno oceno pomembnosti (slika 8). Z namenom približati tehnologijo masovnih podatkov (Hadoop, MapReduce, baze podatkov NoSQL) ter napredna področja strojnega učenja posamez-

nikom in organizacijam v Sloveniji bi bilo priporočljivo vključiti več primerov dobrih praks s tega področja v sklopu predstavitev na konferencah, povabiti strokovnjake iz tujine, več razširjanja znanja v sklopu družabnih skupin in na srečanjih ter v sklopu formalnega izobraževanja spodbujati uporabo

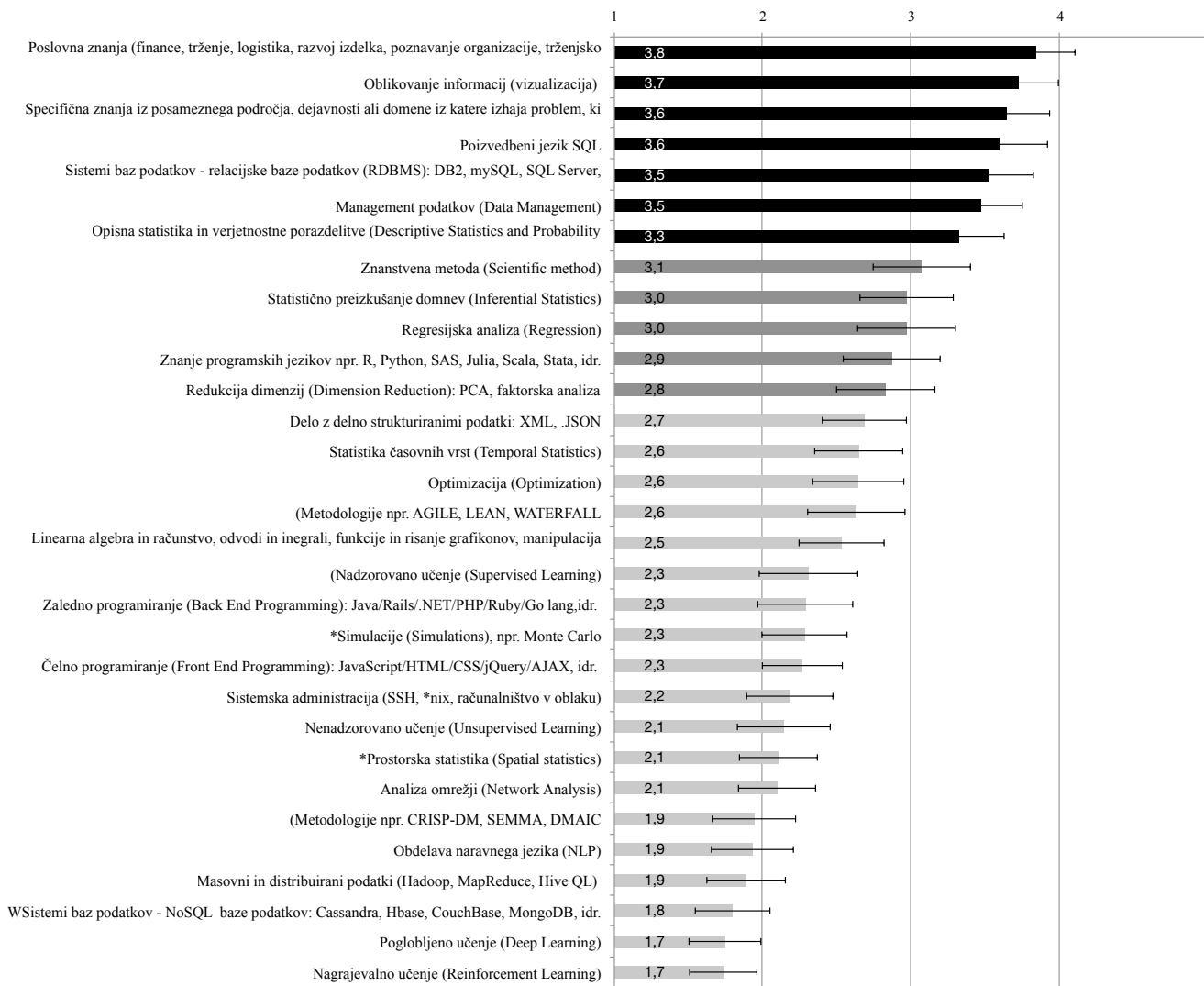
tehnologij masovnih podatkov na odprtih podatkih (angl. open data) ali prek vzajemnega sodelovanja z organizacijami. Primeri dobre prakse in prikazani dejanski učinki uporabe tehnologij masovnih podatkov bi tako spodbudili organizacije, da bodo začele razmišljati o uvajanju teh tehnologij.

1 - Ne poznam 2 - Osnove 3 - Začetnik 4 - Srednji nivo 5 - Napredni nivo



Slika 7: Povprečne ocene samoocene znanj (n = 87-91)

1 - Sploh ni pomembno 2 3 4 5 - Zelo je pomembno



Slika 8: Povprečne ocene pomembnosti znanj (n = 84-91)

Da bi ugotovili, ali obstaja statistično značilna razlika med samooceno znanj in pomembnostjo znanj, smo primerjali povprečne samoocene znanja in ocene pomembnosti znanja. Na podlagi statističnega preizkusa je bila identificirana vrzel med povprečno samooceno znanj in povprečno oceno pomembnosti le pri področjih znanj: regresijska analiza (povprečna samoocena = 3,3; povprečna pomembnost: 3,0; $P = 0,02$) ter linearna algebra in računstvo, odvodi

in integrali, funkcije in risanje grafikonov, manipulacija matrik (povprečna samoocena = 3,1; povprečna pomembnost: 2,5; $P = 0,000$). Pri navedenih znanjih lahko torej trdimo, da obstaja vrzel med obstoječimi znanji (na podlagi samoocene) in pomembnostjo pri njihovem delu. Pri obeh se je izkazalo, da anketiranci menijo, da imajo več znanj, kot je pomembno pri njihovem delu. Rezultat verjetno izhaja iz tega, da se matematike in delno statistike podrobno učimo

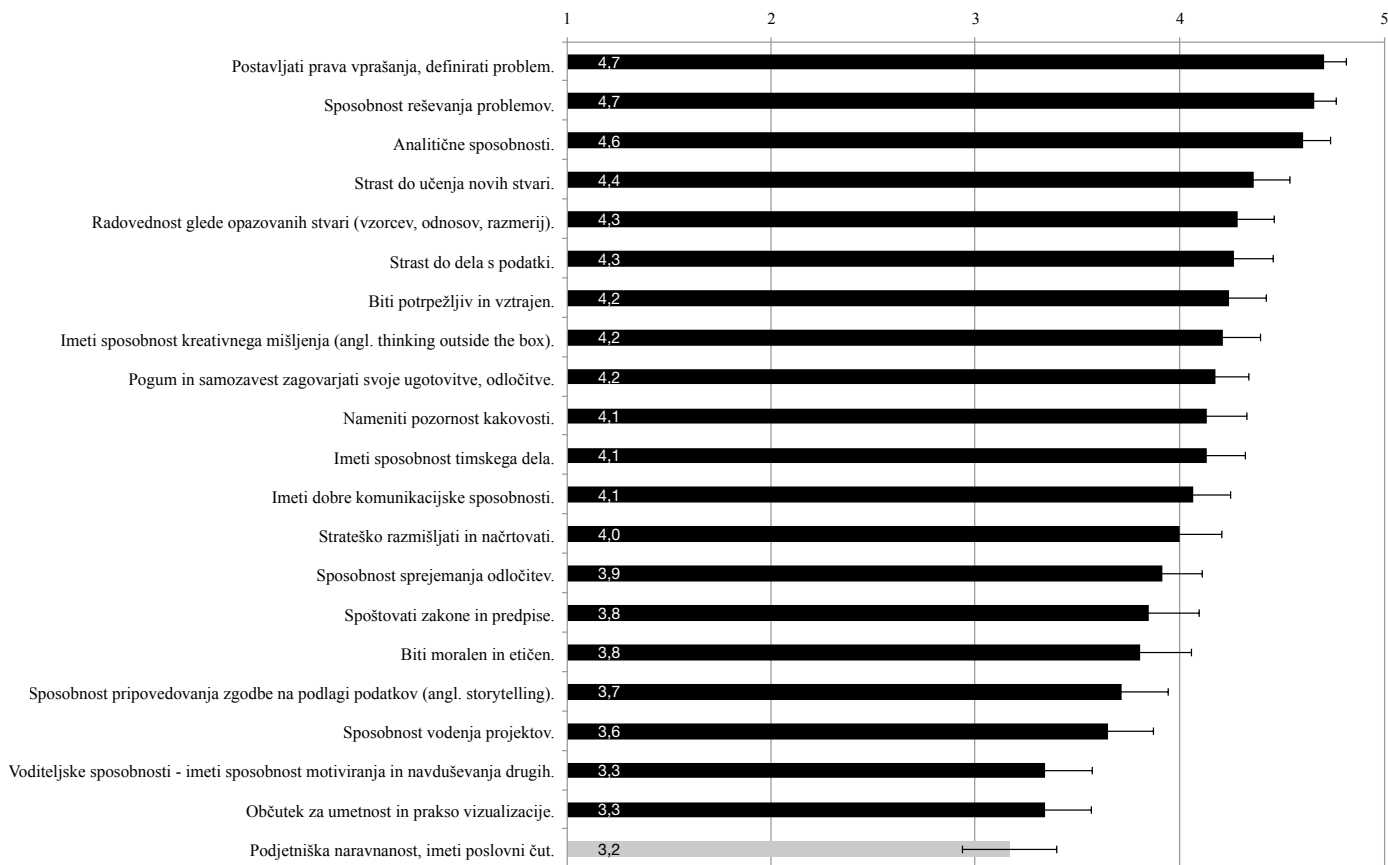
v sklopu formalne izobrazbe (osnovna šola, srednja šola itd.), v praksi pa s teh področij uporabljamo le znanja, ki so pri delu pomembna. Anketiranci namreč opravljajo različne funkcije, pri katerih uporabljajo različna znanja glede na njihovo področje dela, prav vsi pa imajo podobna osnovna izhodišča, npr. iz matematike.

Pri primerjavi rezultatov med raziskavo v Sloveniji in raziskavo Hayesa (2015a) je razvidno, da so skupna področja višje samoocenjenih znanj: strukturirani podatki/relacijske baze podatkov (SQL), menedžment podatkov ter do določene mere statistika (v Sloveniji področje regresijske analize, opisne statistike in verjetnostne porazdelitve). V Sloveniji so visoko povprečno samooceno dobila še druga, zgoraj omenjena področja znanj, ki pa v raziskavi Hayesa niso bila zajeta v vprašalnik v takšni obliki. V raziskavi Hayesa so bila v povprečju višje ocenjena znanja iz matematike in znanosti/znanstvene metode, ki

sta v Sloveniji dobili v povprečju oceno 3 – Začetnik. Sklepamo lahko, da imajo posamezniki v tujini bolj močno formalno izobrazbo na teh dveh področjih oziroma se več posameznikov s teh dveh področij ukvarja z znanostjo o podatkih ali pa omenjena razlika izhaja le iz drugačne sestave in velikosti vzorca.

Pri pregledu rezultatov raziskave o pomembnosti sposobnosti (slika 9) lahko sklepamo, da so vse sposobnosti, razen »podjetniške naravnosti«, anketiranci ocenili kot pomembne pri njihovem delu in da pomembno vplivajo na uspešnost njihovega dela. V sklopu pomembnosti sposobnosti so bile kot najbolj pomembne ocenjene postavljanje pravih vprašanj, sposobnost reševanja problemov ter analitične sposobnosti. To dopolnjuje prejšnjo ugotovitev, da v osnovi podatkovni znanstveniki rešujejo poslovne probleme, za kar potrebujejo ustrezne sposobnosti, da znajo pravilno opredeliti problem, se ga lotiti na pravi način in pri tem ustrezno uporabiti vsa svoja znanja.

1 - Sploh ni pomembno 2 3 4 5 - Zelo je pomembno



Slika 9: Povprečne ocene pomembnosti sposobnosti (n = 90–92)

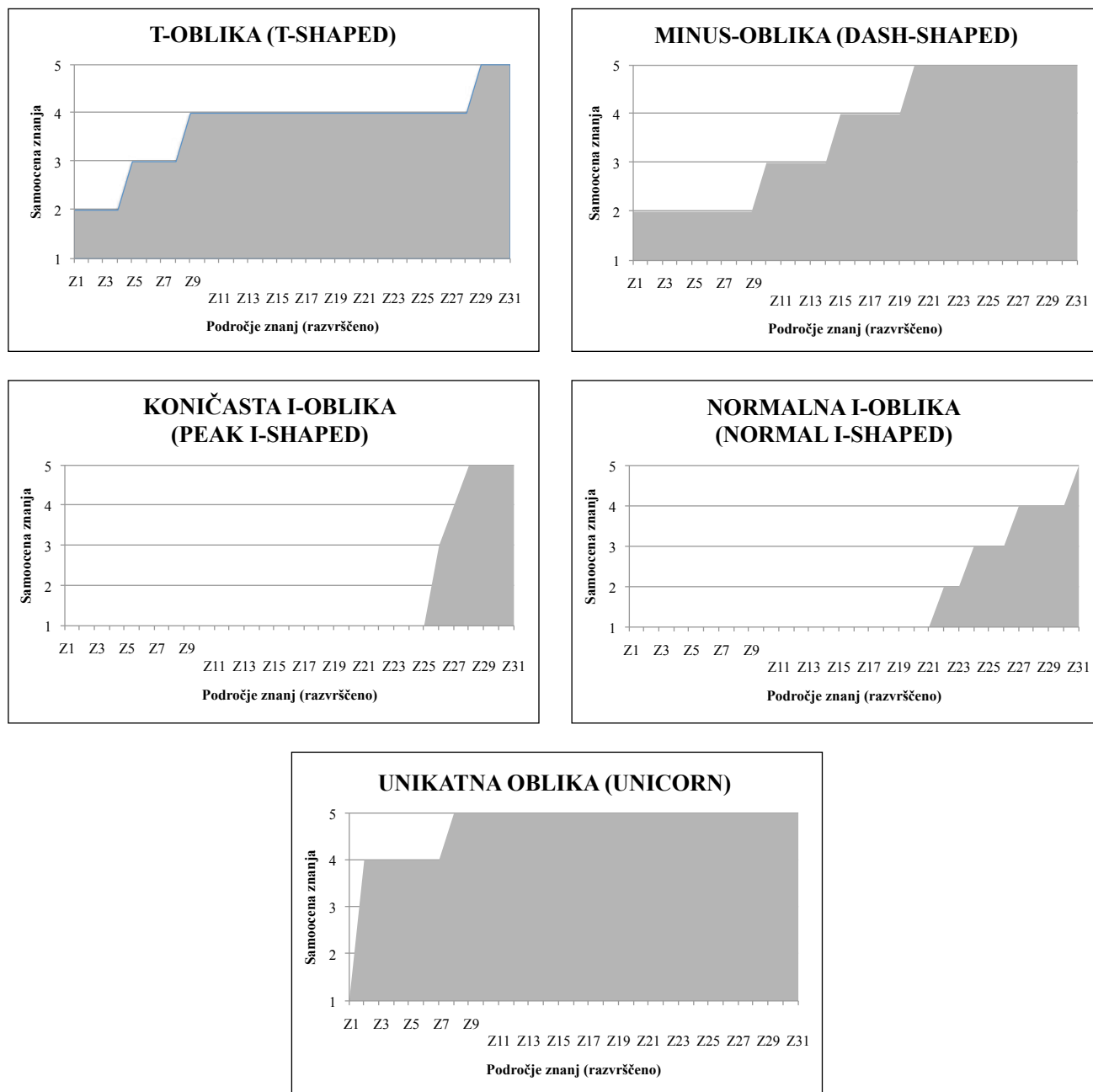
3. Ali je mogoče identificirati porazdelitve oziroma vzorce znanj po posameznih področjih med identificiranimi skupinami podatkovnih znanstvenikov na podlagi samoocene znanj?

Zaradi širokega področja znanj in sposobnosti, ki naj bi jih imel posameznik za delo podatkovnega znanstvenika, se v literaturi (Granville, 2014, str. 75) in v raziskavah (Harris, Murphy in Vaisman, 2013) omenja, da naj bi posameznik imel zelo poglobljena znanja z vsaj enega področja (statistike, matematike, programiranja, baz podatkov, strojnega učenja) ter vsaj osnovna znanja z drugih področij. Granville (2014) govori o t. i. vertikalnih podatkovnih znanstvenikih, v raziskavi *Analyzing the analyzers* (Harris, Murphy in Vaisman, 2013) pa so identificirali t. i. T-obliko znanj podatkovnih znanstvenikov. V sklopu raziskave smo želeli ugotoviti, ali je mogoče identificirati porazdelitve oziroma vzorce znanj po posameznih področjih znanj med skupinami podatkovnih znanstvenikov. Na podlagi tega smo želeli ugotoviti, ali obstajajo podatkovni znanstveniki s t. i. T-obliko znanj oz. ali obstajajo posamezniki, ki imajo visoko stopnjo znanj na vseh področjih.

Pri analizi sta bila uporabljena koeficient asimetrije (angl. skewness) in koeficient sploščenosti (angl. kurtosis). Če je koeficient asimetrije večji od 0, je porazdelitev asimetrična v desno; če je enak 0, je porazdelitev simetrična; če pa je manjši od 0, je porazdelitev asimetrična v levo. Če je koeficient sploščenosti večji od 0, je porazdelitev koničasta; če je enak 0, je porazdelitev normalna (angl. bell-shaped); če pa je manjši od 0, je porazdelitev sploščena. Z namenom identifikacije porazdelitve oziroma vzorcev znanj smo rezultate glede samoocene znanj vseh posameznikov, ki so sodelovali v raziskavi, izvozili v Excel, njihove samoocene znanj pa smo razvrstili od najmanjše do najvišje ter jih vizualizirali z orodjem Sparkline v Excelu. Za vsakega udeleženca smo izračunali koeficient asimetrije KA ter koeficient sploščenosti KS. Ugotovili smo, da je mogoče identificirati naslednje porazdelitve oziroma vzorce znanj posameznikov ($n = 92$) (slika 10):

- T-oblika (angl. T-shaped), če sta KA in KS med -1 in 1 . To so posamezniki, ki imajo visoko samooceno znanj na določenem področju znanj ali parih področij znanj, na drugih področjih pa imajo začetna ali osnovna znanja. Njihova porazdelitev znanj je simetrična ter približno podobna normalni porazdelitvi. Takšnih anketirancev je bilo v vzorcu 31,5 odstotka.

- Minus-oblika (angl. Dash-shaped), če je KA med -1 in 1 ter KS manjši od -1 . To so posamezniki, ki imajo simetrično porazdelitev znanj po področjih, vendar je njihova porazdelitev bolj sploščena (KS je manjši od -1). To pomeni, da imajo z veliko področij znanj določeno stopnjo znanja, na nobenem področju pa ne izstopajo ali pa hkrati izstopajo na več področjih. Takšnih anketirancev je bilo v vzorcu 34,8 odstotka.
- Normalna I-oblika (angl. Normal I-shaped), če je KA večji od 1 ter KS med -1 in 1 . To so posamezniki, ki imajo porazdelitev znanj asimetrično v desno. To pomeni, da imajo določeno področje znanja, ki ima visoko oceno, pri ostalih pa imajo zelo nizke samoocene znanja ali pa jih sploh ne poznajo. Njihova značilnost je še, da je njihova porazdelitev precej podobna normalni (KS je med -1 in 1). To pomeni, da imajo določeno poznavanje ostalih znanj, čeprav ne tako visoko kot posamezniki pri T-obliki. Takšnih anketirancev je bilo v vzorcu 21,7 odstotka.
- Koničasta I-oblika (angl. Peak I-shaped), če sta KA in KS večja od 1 . To so posamezniki, ki so podobni »normalni I-obliki«, vendar je njihova »koničnost« še bolj izrazita (KS je večji od 1), kar še dodatno poudarja višjo oceno znanj iz samo določenega področja. Takšnih anketirancev je bilo v vzorcu 5,4 odstotka.
- Unikatna oblika (angl. Unicorn), če je KA manjši od -1 ter KS večji od 1 . Na podlagi podatkov so bili identificirani tudi posamezniki, ki imajo porazdelitve znanj zelo asimetrične v levo (KA je manjši od -1) ter zelo »koničasto« porazdelitev. To pomeni, da imajo visoko samooceno iz vseh znanj, kar jih naredi zelo unikatne. Takšnih anketirancev je bilo v vzorcu 6,5 odstotka.



Slika 10: Porazdelitve znanj po izbranih anketirancih – samoocena znanj

4. Katere skupine podatkovnih znanstvenikov v Sloveniji lahko identificiramo na podlagi samoocene znanj?

Na podlagi razvrščanja v skupine z algoritmom K-means je bilo identificiranih pet skupin posameznikov v Sloveniji, ki se med seboj razlikujejo glede samoocene svojih znanj. Centroidi skupin so prikazani na sliki 11.

Interpretacija skupine C1: posamezniki v skupini C1 imajo v povprečju osnovna znanja (2,1) iz pro-

gramskih jezikov, npr. R, Python ter čelnega programiranja. Zaledno programiranje pa so v povprečju ocenili z Ne poznam ali uporabljam. V sklopu znanj iz menedžmenta in baz podatkov imajo povprečno oceno 3 pri oblikovanju informacij (vizualizacija) in menedžmentu podatkov. Osnovna znanja imajo v povprečju iz relacijskih baz podatkov, delno strukturiranih podatkov in poizvedbenega jezika SQL. Masovnih in distribuiranih podatkov, baz podatkov

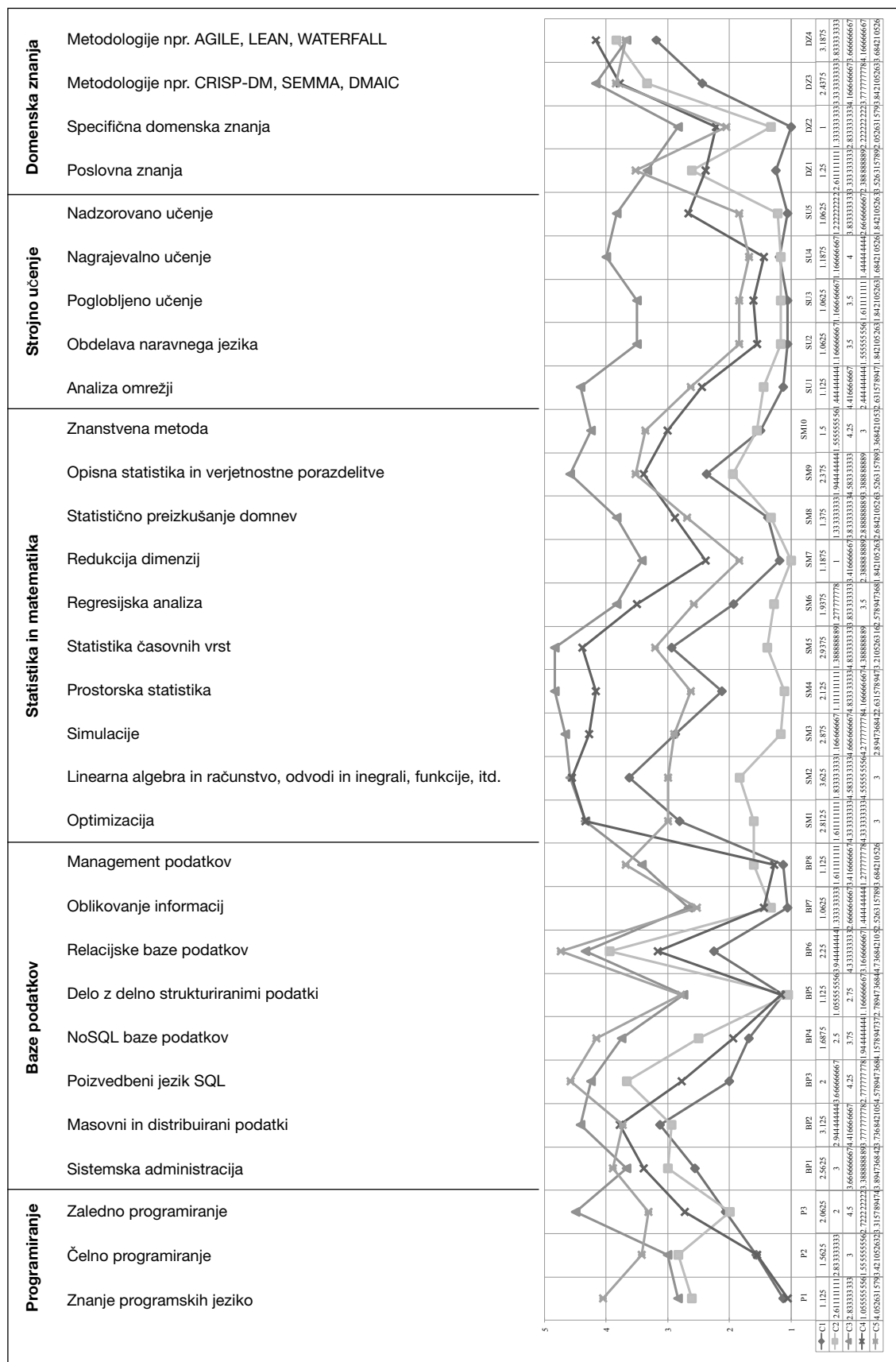
NoSQL ne poznajo, prav tako se ne ukvarjajo s sistemsko administracijo. V sklopu statistike in matematike imajo najvišjo povprečno oceno iz opisnih statistik in verjetnostnih porazdelitev (3,6), statističnega preizkušanja domnev (2,9), regresijske analize (2,9) ter znanstvene metode (2,8). Najmanj poznajo ali uporabljajo prostorsko statistiko (1,2), optimizacije (1,5) in simulacije (1,4). Celoten sklop strojnega učenja v povprečju ne poznajo ali ne uporabljajo. Prav tako v povprečju ne poznajo metodologij AGILE, LEAN, WATERFALL in CRISP-DM, SEMMA, DMAIC. Osnovna znanja imajo iz specifičnih znanj s posameznega področja ali domene, iz katere izhaja problem (2,4) ter začetna znanja s področja poslovnih znanj (3,2). Na podlagi navedenih značilnosti je bila ta skupina poimenovana trženjski raziskovalci – analitiki in zavzema 19 odstotkov posameznikov iz vzorca.

Trženjski raziskovalci – analitiki imajo torej začetni nivo znanj iz statistike in matematike ter poslovnih znanj. Dodatna analiza te skupine je pokazala, da jih ima polovica univerzitetno izobrazbo, smer izobrazbe pa je ekonomska (31 odstotkov) in družboslovna (38 odstotkov). Oba spola sta enako zastopana (50 odstotkov). Menijo, da imajo manj znanj, kot pa so pomembna, s področij oblikovanja informacij, relacijskih baz podatkov, poglobljenega učenja in analize omrežij. Z vidika dimenzije raznolikosti podatkov se jih največji delež ukvarja s MB (38 odstotkov) in GB (38 odstotkov) podatki ter z notranjimi, strukturiranimi podatki, generiranimi s strani naprav ali človeka (31 odstotkov). V tej skupini je najnižji delež takšnih, ki so se srečali z vsemi dimenzijami raznolikosti podatkov (13 odstotkov). Z vidika porazdelitve samoocene znanj v tej skupini prevladujejo posamezniki (50 odstotkov), ki imajo normalno I-obliko porazdelitve znanj. To pomeni, da imajo določeno področje znanja visoko ocenjeno, pri ostalih pa imajo nizke samoocene znanja ali pa jih sploh ne poznajo. Ta skupina naj v prihodnosti predvsem razvija naprej znanja iz statistike in matematike, ki sta njeni najmočnejši področji. Manjka jim predvsem razširitev njihovih znanj (vsaj na osnove) s področij znanj programiranja, baz podatkov, menedžmenta podatkov, strojnega učenja in domenskih znanj, da bi postali podatkovni znanstveniki. S programiranjem bi se lahko seznanili ob uporabi programskega jezika R, v katerem bi se lahko hitro naučili izvedbo ukazov s področja statistike, ki bi jih

lahko takoj uporabili pri svojem delu. Ker se pogosto udeležujejo delavnic ali tečajev, bi lahko na kateri od njih predstavili nove možnosti vizualizacije, osnove baz podatkov ter metode strojnega učenja, ki bi jih lahko kar najhitreje praktično uporabili pri svojem delu. Pozitivni učinki uporabe pri delu bi jih spodbudili k nadaljnji uporabi in raziskovanju teh področij tudi v prihodnje, saj določen del te skupine meni, da sta pri delu s podatki pomembni radovednost in strast.

Interpretacija skupine C2: Posamezniki iz skupine C2 imajo v povprečju osnovna znanja iz programskih jezikov (2,0) ter začetna znanja iz zalednega programiranja (2,6) in čelnega programiranja (2,8). V sklopu znanj iz menedžmenta in baz podatkov imajo znanja na srednjem nivoju poizvedbenega jezika SQL (3,9) ter iz relacijskih baz podatkov (3,7). Začetna znanja imajo na področju menedžmenta podatkov (3,0), oblikovanja informacij (2,9) ter dela z delno strukturiranimi podatki (2,5). Masovnih in distribuiranih podatkov, baz podatkov NoSQL ne poznajo, pri sistemski administraciji poznajo osnove. Na področju statistike in matematike v povprečju nimajo znanj ali jih ne uporabljajo, ali pa poznajo le osnove (znanstvena metoda, opisna statistika, linearna algebra in računstvo, odvodi in integrali, funkcije in risanje grafikonov, manipulacija matrik, optimizacija). Celotnega sklopa strojnega učenja v povprečju ne poznajo ali ga ne uporabljajo. Prav tako ne poznajo metodologij CRISP-DM, SEMMA, DMAIC. Bolj so seznanjeni z metodologijami AGILE, LEAN, WATERFALL (2,6). Začetna znanja imajo iz specifičnih znanj s posameznega področja ali domene, iz katere izhaja problem (3,3), ter srednji nivo znanj iz področja poslovnih znanj (3,8). Na podlagi navedenih značilnosti je bila ta skupina poimenovana podatkovni analitiki in zavzema 22 odstotkov posameznikov iz vzorca.

Podatkovni analitiki imajo torej srednji nivo znanj s področij baz podatkov ter poslovnih znanj ter začetna znanja s področja programiranja. Dodatna analiza te skupine je pokazala, da v tej skupini prevladujejo moški (61 odstotkov), univerzitetna izobrazba (56 odstotkov), smer izobrazbe računalništvo (28 odstotkov) ter druge naravoslovne in tehnične vede (22 odstotkov). Menijo, da imajo več znanj, kot so pomembna, s področij: čelno programiranje, regresija, nadzorovano učenje, linearna algebra in računstvo, odvodi in integrali, funkcije in risanje grafikonov, manipulacija matrik. Imajo potencial, da



postanejo podatkovni znanstveniki, saj se jih že sedaj 44 odstotkov ukvarja s podatki v GB ter skoraj 40 odstotkov z vsemi dimenzijami raznolikosti podatkov, prav tako pa očitno delajo na področjih, na katerih so pomembna znanja programiranja (čelno) in strojnega učenja. Z vidika porazdelitve samoocene znanj v tej skupini prevladujejo posamezniki (61 odstotkov), ki imajo normalno I-obliko porazdelitve znanj. To pomeni, da imajo določeno področje znanja visoko ocenjeno, pri ostalih pa imajo nizke samoocene znanja ali pa teh znanj sploh nimajo. Ta skupina naj v nadaljevanju razvija znanja s področij znanosti/znanstvena metoda, programiranja, statistike, strojnega učenja in domenskih znanj. Njihova prednost je v dobri osnovi na področju baz podatkov in poslovnih znanjih.

Interpretacija skupine C3: Posamezniki iz skupine C3 imajo v povprečju napredna znanja iz programskih jezikov (4,5) ter srednji nivo znanj iz zalednega programiranja (2,8) in čelnega programiranja (3). V sklopu znanj iz menedžmenta in baz podatkov imajo srednji nivo znanj iz oblikovanja informacij (4,4), poizvedbenega jezika SQL (4,3), relacijskih baz podatkov (4,3), delno strukturiranih podatkov (3,8) in menedžmenta podatkov (3,7). Začetna znanja imajo na področju systemske administracije (3,4), masovnih in distribuiranih podatkov (2,7) in baz podatkov NoSQL (2,8). V sklopu statistike in matematike imajo pri vseh področjih znanj v povprečju srednja ali napredna znanja, prav tako na področju strojnega učenja. Na področju domenskih znanj pa imajo začetni nivo znanj iz metodologij AGILE, LEAN, WATERFALL ter CRISP-DM, SEMMA, DMAIC, pri ostalih domenskih znanjih pa srednji nivo. Posamezniki iz te skupine so v primerjavi z ostalimi skupinami edini, ki imajo največje število področij znanj ocenjeno s povprečno oceno 3 ali več. Na podlagi navedenih značilnosti je bila ta skupina poimenovana podatkovni znanstveniki in zavzema 14 odstotkov posameznikov iz vzorca.

Podatkovni znanstveniki izstopajo predvsem po naprednem znanju programskih jezikov in so edini od skupin, ki imajo največje število področij znanj ocenjeno s povprečno oceno 3 (začetnik) ali več. Dodatna analiza te skupine je pokazala, da prevladujejo moški (83 odstotkov), univerzitetna izobrazba (56 odstotkov) ter računalniška smer izobrazbe (33 odstotkov). Menijo, da imajo več znanj, kot pa so pomembna, na področjih opisna statistika in verjetnostne porazdelitve ter redukcija dimenzij. Ukvarjajo

se s podatki v GB (42 odstotkov) in TB (41 odstotkov). Polovica se je že srečala z vsemi dimenzijami raznolikosti podatkov. Kakovost izdelkov in storitev jim je enako pomembna kot odločanje na podlagi dejstev. Z vidika porazdelitve samoocene znanj v tej skupini prevladujejo posamezniki, ki imajo unikatno obliko (42 odstotkov) in T-obliko (42 odstotkov) porazdelitve znanj. V primerjavi z drugimi skupinami so tudi edina skupina, ki sploh vsebuje unikatno obliko porazdelitve znanj. Za to skupino je predvsem pomembno to, da lahko svoje bogato znanje uporabijo v praksi na zanimivih projektih, ki jim bodo predstavljali izziv. Svoje znanje že sedaj izpopolnjujejo in ga bodo tudi v prihodnje, zato je pomembno, da so obveščeni o aktualnih konferencah v Sloveniji in tujini ter aktualnih natečajih in tekmovanjih. Ker so jim verjetno najbolj pomembne praktične izkušnje iz izvedenih projektov, bi lahko znanje medsebojno delili prek srečanj v družabnih skupinah.

Interpretacija skupine C4: Posamezniki iz skupine C4 v povprečju ne uporabljajo zalednega programiranja ali ga ne poznajo. Začetna znanja imajo iz programskih jezikov (2,7) in čelnega programiranja (1,6). V sklopu znanj iz menedžmenta in baz podatkov imajo srednji nivo znanj iz oblikovanja informacij (3,8), začetni nivo iz menedžmenta podatkov (3,4), poizvedbenega jezika SQL (3,2), relacijskih baz podatkov (2,8). V povprečju imajo osnove iz dela z delno strukturiranimi podatki (1,9). Masovnih in distribuiranih podatkov, baz podatkov NoSQL v povprečju ne poznajo, prav tako se ne ukvarjajo s systemsko administracijo. V sklopu statistike in matematike imajo pri vseh področjih znanj v povprečju srednji nivo znanj, razen pri prostorski statistiki (2,4), simulacijah (2,9), optimizaciji (3). S področja strojnega učenja imajo v povprečju osnovna znanja, razen pri obdelavi naravnega jezika (1,4), ki ga ne poznajo ali uporabljajo, ter analizi omrežij (2,7), za katero imajo začetna znanja. Na področju domenskih znanj pa imajo osnovni nivo znanj iz obeh metodologij. Srednji nivo znanj imajo iz specifičnih znanj (3,8) in poslovnih znanj (4,2). Na podlagi navedenih značilnosti je bila ta skupina poimenovana raziskovalci in zavzema 22 odstotkov posameznikov iz vzorca.

Raziskovalci zelo dobro kombinirajo znanja s področja oblikovanja informacij, poslovnih znanj, baz podatkov ter statistike. Dodatna analiza te skupine je pokazala, da v tem segmentu izjemoma prevladujejo ženske (73 odstotkov), univerzitetna izobrazba

(50 odstotkov) ter smer izobrazbe statistika in ekonomija. Predstavljajo zelo dober potencial, da postanejo podatkovni znanstveniki, saj iz te skupine prihaja najvišji delež posameznikov, ki je kot največjo obdelano količino podatkov izbralo PB (6 odstotkov), drugače pa se ukvarjajo s podatki v MB (33 odstotkov) in GB (39 odstotkov). V tej skupini se jih največ (33 odstotkov) ukvarja z vsemi dimenzijami raznolikosti podatkov. Z vidika porazdelitve samoocene znanj v tej skupini prevladujejo posamezniki, ki imajo minus obliko (72,2 odstotkov) porazdelitve znanj. To pomeni, da imajo določeno stopnjo znanja z veliko področij, na nobenem pa ne izstopajo ali pa izstopajo na več področjih. Ta skupina naj v nadaljevanju razvija znanja s področja programiranja in strojnega učenja ter domenska znanja. Ker imajo dobra znanja s področja vizualizacije, bi jim lahko približali programske jezike in programiranje prek različnih načinov vizualizacij podatkov s pomočjo programskih jezikov. Z uporabo strojnega učenja pa bi lahko izboljšali rezultate, kjer si želijo doseči boljše odločanje na podlagi dejstev. V primerjavi z drugimi skupinami se veliko udeležujejo masovnih odprtih spletnih tečajev, na katerih lahko pridobijo omenjena znanja.

Interpretacija skupine C5: Posamezniki iz skupine C5 imajo v povprečju srednji nivo znanj iz zalednega programiranja (4,1) ter osnovni nivo iz čelnega programiranja (3,4) ter programskih jezikov (3,3). Znanja programiranja najbolj izstopajo v tej skupini od vseh naštetih skupin. V sklopu znanj iz menedžmenta in baz podatkov imajo napredni nivo znanj iz relacijskih baz podatkov (4,6) in poizvedbenega jezika SQL (4,7). Srednji nivo znanj imajo iz menedžmenta podatkov, oblikovanja informacij, dela z delno strukturiranimi podatki in sistemsko administracijo. So edina skupina, ki ima začetni nivo znanj na področju baz podatkov NoSQL (2,8) ter masovnih in distribuiranih podatkov (2,5). V sklopu statistike in matematike imajo pri vseh področjih znanj v povprečju začetni nivo znanj, razen pri linearni algebr in računstvu (3,5) in prostorski statistiki (1,8). Na področju strojnega učenja imajo v povprečju osnovna znanja, razen pri nadzorovanem učenju (2,6), pri katerem imajo začetna znanja. Na področju domenskih znanj imajo srednji nivo znanj metodologij AGILE, LEAN, WATERFALL ter osnove iz metodologij CRISP-DM, SEMMA, DMAIC. Srednji nivo znanj imajo iz specifičnih znanj (3,8) in poslovnih znanj (3,7). Na podlagi navedenih značilnosti je bila ta skupina poimenova-

na programerji in zavzema 23 odstotkov posameznikov iz vzorca.

Programerji izstopajo z najvišjo povprečno samooceno znanj iz programiranja, baz podatkov in domenskih znanj. Prevladujejo moški (84 odstotkov), univerzitetna izobrazba (37 odstotkov) in smer izobrazbe računalništvo (53 odstotkov). So edina skupina, ki ima začetni nivo znanj na področju baz podatkov NoSQL ter osnovni nivo znanj iz masovnih in distribuiranih podatkov. To potrjuje tudi dejstvo, da se jih največ ukvarja s podatki v GB (53 odstotkov) in TB (32 odstotkov). Ker menijo, da imajo manj poslovnih znanj, kot pa so pomembna pri njihovem delu, naj razvijajo znanja s področja poslovnih ved. Da bi postali podatkovni znanstveniki, jim manjkajo še znanja s področja znanosti/znanstvene metode in statistike. Z vidika porazdelitve samoocene znanj v tej skupini prevladujejo posamezniki, ki imajo T-obliko (53 odstotkov) in minus obliko porazdelitve znanj. Priporočljivo je, da ta skupina tesno sodeluje s skupino podatkovnih znanstvenikov pri različnih projektih. Na podlagi skupnega sodelovanja bodo lahko programerji pridobili vpogled v znanstveni pristop k podatkom. Znanja s področij statistike, znanstvene metode in poslovnih znanj lahko pridobijo s formalno izobrazbo ali vsaj z udeležbo na masovnem odprtem spletnem tečaju.

Čeprav smo pri oblikovanju raziskovalnega načrta, metodologije ter pri sami izvedbi raziskave in analizi podatkov kar najbolj upoštevali raziskovalne standarde (Malhotra, 2012), ima raziskava tudi določene omejitve. Prva omejitev izhaja iz velikosti vzorca – če bi bila velikost vzorca večja, bi lahko bili rezultati bolj zanesljivi. Poleg tega je bilo uporabljeno priložnostno namensko vzorčenje (neverjetnostno vzorčenje), kar pomeni, da vzorec ni reprezentativen in rezultatov raziskave ni mogoče posplošiti na populacijo. Kljub temu so bili k raziskavi povabljeni posamezniki, ki s svojim področjem dela pokrivajo širok nabor strokovnjakov, ki bi jih lahko uvrščali med podatkovne znanstvenike, zato menimo, da rezultati raziskave ponujajo dober vpogled v stanje na tem področju in dajejo podlage za ukrepanje.

5 SKLEPNE UGOTOVITVE

Konvergenca različnih znanstvenih disciplin je omogočila pojav novega razreda strokovnjaka, podatkovnega znanstvenika. Podatkovni znanstvenik naj bi imel znanja s področij programiranja, menedžmenta

podatkov, baz podatkov, znanosti (znanstvena metoda), statistike, matematike, strojnega učenja in domenskih znanj. V sklopu sposobnosti pa so pomembni: sposobnost definiranja in reševanja problemov, analitične sposobnosti, strast do učenja novih stvari, radovednost, strast do dela s podatki, potrpežljivost, vztrajnost, kreativno mišljenje, pogum in samozavest zagovarjati svoje odločitve, pozornost nameniti kakovosti, sposobnost timskega dela, sposobnost komunikacije, strateško razmišljanje, sposobnost sprejemanja odločitev, spoštovanje zakonov in predpisov, moralnost in etičnost, sposobnost pripovedovanja zgodbe, sposobnost vodenja projektov, sposobnost motiviranja in navduševanja drugih ter občutek za umetnost in prakso vizualizacije. Na ta način ima namreč vse potrebno, da lahko samostojno izvede celoten proces znanosti o podatkih. Na podlagi identificiranih segmentov in njihovih značilnosti lahko sklepamo, da v Sloveniji obstajajo posamezniki, ki bi jim lahko podelili naziv podatkovni znanstvenik, saj imajo znanja in sposobnosti z vseh identificiranih področij znanj, s katerimi lahko pokrijejo celoten proces izvajanja znanosti o podatkih. Glede na podatke o rastočem povpraševanju po takšnih posameznikih v svetu bo v prihodnosti predvsem pomembno ustvariti okolje in razmere, da bodo takšni posamezniki našli ustrezne izzive za izpopolnitev svojega zmožnosti v Sloveniji, hkrati pa razviti oziroma dopolniti zmožnosti preostalih posameznikov iz identificiranih skupin. Pri tem bo zelo pomembna podpora v sklopu formalnega izobraževanja na navedenih področjih, stalno izpopolnjevanje, prenos znanja med posamezniki in skupinami ter pridobivanje izkušenj na praktičnih primerih. V ta namen bi bilo treba še bolj spodbujati srečanja v družabnih skupinah, omogočiti delo na »odprtih« podatkih ter ustrezna znanja za opravljanje takšne pozicije vključiti v del redne formalne izobrazbe.

6 LITERATURA IN VIRI

- [1] *Big Data Executive Survey* (2012). Najdeno 10. 1. 2015 na <http://newvantage.com/wp-content/uploads/2012/12/NVP-Big-Data-Survey-Themes-Trends.pdf>.
- [2] Boyd, D. in Crawford, K. (2012). Critical questions for big data. *Information, Communication & Society*, 15(5), 662–679.
- [3] Chordas, L. (2014). Data driven. *Best's Review*, 115(1), 22–26.
- [4] DAMA (2014, 6. marec). DAMA-DMBOK2 Framework Guide. *Dama*. Najdeno 21. 3. 2016 na <https://www.dama.org/sites/default/files/download/DAMA-DMBOK2-Framework-V2-20140317-FINAL.pdf>.
- [5] Davenport, T. T., in Patil, D. J. (2012). Data scientists: the sexiest job of the 21st century. *Harvard Business Review*, oktober 2012, 70–76.
- [6] Dhar, V. (2013). Data Science and Prediction. *Communications of the ACM*, 56(12), 64–73.
- [7] *The Emerging Big Returns on Big Data* (2013). Najdeno 16. 1. 2015 na http://www.tcs.com/SiteCollectionDocuments/Trends_Study/TCS-Big-Data-Global-Trend-Study-2013.pdf.
- [8] *The field guide to data science*. Najdeno 10. 1. 2015 na <https://www.boozallen.com/content/dam/boozallen/documents/2015/12/2015-Field-Guide-To-Data-Science.pdf>.
- [9] Granville, V. (2013). Job titles for data scientists. *Datasciencecentral*. Najdeno 5. 12. 2015 na <http://www.datasciencecentral.com/profiles/blogs/job-titles-for-data-scientists>.
- [10] Granville, V. (2014). *Developing analytic talent: becoming a data scientist*. United States: Wiley.
- [11] Harris, H., Murphy, S. in Vaisman, M. (2013). *Analyzing the analyzers: an introspective survey of data scientists and their work*. United States: O'Reilly Media.
- [12] Hayes, B. E. (2014a). The One hidden skill you need to unlock the value of your data. *Businessoverbroadway*. Najdeno 5. 12. 2015 na <http://businessoverbroadway.com/the-one-hidden-skill-you-need-to-unlock-the-value-of-your-data>.
- [13] Hayes, B. E. (2014b). The what and where of big data: a data definition framework. *Customerthink*. Najdeno 5. 12. 2015 na <http://customerthink.com/the-what-and-where-of-big-data-a-data-definition-framework/>.
- [14] Hayes, B. E. (2015a). Optimizing your data science team, a survey of data professionals. *Analytics Week*. Najdeno 5. 12. 2015 na <https://analyticsweek.com/docs/research/open/OptimizingYourDataScienceTeamsV2.0.pdf>.
- [15] Kuhn, M. (2015). CRAN task view: Reproducible research. CRAN. Najdeno 5. 12. 2015 na <https://cran.r-project.org/web/views/ReproducibleResearch.html>.
- [16] Leban, G. (2007). *Vizualizacija podatkov s strojnim učenjem*. Doktorska disertacija. Ljubljana: Fakulteta za računalništvo in informatiko.
- [17] Lorica, B., Howard, J., Dumbill, E. (2012, 11. januar). What is big data. *O'Reilly*. Najdeno 21. 11. 2015 na <https://beta.oreilly.com/ideas/what-is-big-data>.
- [18] Malhotra, N. K. (2012). *Basic marketing research: integration of social media* (4th ed.). New Jersey: Prentice Hall.
- [19] Manyika, J., idr. (2011). Big data: The next frontier for innovation, competition, and productivity. *McKinsey Global Institute*. Najdeno 5. 12. 2015 na http://www.mckinsey.com/insights/business_technology/big_data_the_next_frontier_for_innovation.
- [20] O'Neill, C. in Schutt, R. (2013). *Doing data science*. United States: O'Reilly Media.
- [21] OECD (2015). *Data driven innovation: Big Data for growth and well-being*. Paris: OECD Publishing.
- [22] Olofson, C. W. in Vesset, D. (2012). Big Data: Trends, Strategies, and SAP Technology. *SAP*. Najdeno 16. 1. 2015 na https://www.sap.com/bin/sapcom/en_ae/downloadasset.2012-09-sep-26-13.idc-report--big-data-trends-strategies-and-sap-technology-pdf.html.
- [23] Piatetsky, G. (2014). Four main languages for Analytics, data mining, data science. *Kdnuggets*. Najdeno 21. 11. 2015 na <http://www.kdnuggets.com/2014/08/four-main-languages-analytics-data-mining-data-science.html>.
- [24] Piatetsky, G. (2015). Poll results: Where is big data?. *Kdnuggets*. Najdeno 15. 8. 2015 na <http://www.kdnuggets.com/2015/08/largest-dataset-analyzed-more-gigabytes-petabytes.html>.
- [25] Rivera, R. in Haverson, A. (2014). Data Scientist vs Data Analyst. *Captechconsulting*. Najdeno 15. 12. 2015 na <https://www.captechconsulting.com/blogs/data-scientist-vs-data-analyst>.

- [26] Russom, P. (2011). Big Data Analytics. *Tableau*. Najdeno 21. 3. 2016 na http://www.tableau.com/sites/default/files/whitepapers/tdwi_bpreport_q411_big_data_analytics_tableau.pdf.
- [27] Somohano, C. (2013). Big data & data science: what does a data scientist do?. *Data Science London*. Najdeno 21. 11. 2015 na <https://www.slideshare.net/datasciencelondon/big-data-sorry-data-science-what-does-a-data-scientist-do>.
- [28] Stanton, J. M. (2013). Introduction to data science. *iTunes*. Najdeno 21. 1. 2016 na <https://itunes.apple.com/us/book/introduction-to-data-science/id529088127?mt=11>.
- [29] Swan, A. (2008). The skills, role and career structure of data scientists and curators: an assessment of current practice and future needs. *Key Perspectives*. Najdeno 17. 1. 2015 na <http://beta.jisc.ac.uk/media/documents/programmes/digital-repositories/data>.
- [30] Toš, N. in Hafner-Fink, M. (1998). *Metode družboslovnega raziskovanja*. Ljubljana: Fakulteta za družbene vede.
- [31] *Ultimate skills checklist for your first data analyst job*. Najdeno 21. 11. 2015 na <http://static.cdn.responsesys.net/i2/responsesysimages/content/udacity/Ultimate%20Skills%20Checklist%20For%20Your%20First%20Data%20Analyst%20Job.pdf>.
- [32] Vesset, D., idr. (2012). Worldwide big data technology and services 2012 - 2016 Forecast. *IDC*. Najdeno 21. 11. 2015 na <http://laser.inf.ethz.ch/2013/material/breitman/additional%20reading/Worldwide%20Big%20Data%20Technology%20and%20Services%202012-2016%20Forecast.pdf>.
- [33] Voulgaris, Z. (2014). *Data scientist: The definitive guide to becoming a data scientist*. United States: Technics Publications.

■

Mateja Grobelnik je zaposlena v podjetju Petrol, d. d., kot analitik. Leta 2016 je dokončala znanstveni magistririj na Ekonomski fakulteti Univerze v Ljubljani, smer informacijsko-upravljalne vede. Področja, ki jo zanimajo in s katerimi se ukvarja, so poslovna inteligenca, baze podatkov, strojno učenje in statistika.

■

Jurij Jaklič je redni profesor na Katedri za poslovno informatiko in logistiko na Ekonomski fakulteti Univerze v Ljubljani. Raziskovalno, pedagoško in svetovalno se ukvarja predvsem s področjem podatkovne analitike oz. poslovne inteligence, pa tudi z menedžmentom poslovnih procesov. Je (so)avtor več kot sto znanstvenih člankov in poročil projektov, številni so bili objavljeni v mednarodnih znanstvenih revijah. Sodeloval je v več raziskovalnih in svetovalnih projektih s področja poslovne inteligence, prenove poslovnih procesov in strateškega načrtovanja informatike.

▣ Razvoj modela sistemske dinamike investiranja in donosnosti kapitala v organizaciji in realizacija v obliki spletne aplikacije

¹Damijan Kozina, ²Davorin Kofjač, ³Andrej Škraba

¹UniCredit banka Slovenije, d. d., Tržaška c. 19, 1000 Ljubljana

^{2,3}Univerza v Mariboru, Fakulteta za organizacijske vede, Kidričeva cesta 55a, 4000 Kranj

Izvleček

Prispevek obravnava vpliv stopnje investiranja in zadolževanja posamezne organizacije na njeno finančno stanje s pomočjo simulacijskega modela, razvitega v jeziku javascript. Ustrezno razmerje med investiranjem in zadolženostjo pomembno prispeva k trajnostnemu razvoju organizacije in poslovnega okolja. Obravnavanih je več možnih simulacijskih scenarijev. Prevelika zadolženost zavira gospodarsko rast, po drugi strani pa premajhna stopnja investiranja tudi ne zagotavlja ustreznega napredka. Z razvitim modelom lahko kvantitativno podpremo strategijo organizacije za uspešno delovanje na dolgi rok. Razvit je bil delujoč simulacijski model, ki upošteva stopnjo investiranja in zadolževanja v obravnavani organizaciji in prikaže rezultate simulacije. Opredeljeni so ustrezni vhodni parametri modela, ki zagotavljajo kar najboljši rezultat za posamezno organizacijo. Uporabljena je metodologija sistemske dinamike z obravnavo ključnih povratnih zank. Izdelan je simulacijski model, podprt z aplikacijo, ki je testirana z uporabo testnih primerov in vključenimi različnimi scenariji. Simulacijski model je izdelan z orodjem Powersim ter realiziran s programskim jezikom javascript/ECMAScript; dostopen je na svetovnem spletu.

Ključne besede: sistemska dinamika, model dinamike razvoja, JavaScript, ECMAScript, spletna simulacija.

Abstract

Development of System Dynamics Model of Investment and Return on Equity in Organizations and Realization of Web Simulation Application

This paper describes the relation between investments and debt and its influence on the financial state of an organization. The simulation model was developed and realized in JavaScript. The suitable ratio between the investment and debt contributes to the sustainable development of an organization and the business environment. Several simulation scenarios are considered. High debt hinders company growth; however, insufficient investments cannot guarantee proper progress. Via the developed model, a particular strategy could be quantitatively determined for long-term sustainable organizational progress. A simulation model was developed, which considers the investments and debt and provides for the analysis of results. Input parameters of the model were determined, which provide the best result for individual organizations. The System Dynamics method was applied, taking into account major feedback loops. The simulation model was developed using a web user interface which was tested by stating several simulation scenarios. The simulation model was first developed in Powersim and then migrated to JavaScript/ECMAScript. The model can be accessed via World Wide Web.

Keywords: System Dynamics, JavaScript, ECMAScript, web simulation.

1 UVOD

Pomen ustreznega razmerja med zadolženostjo in kapitalom je v zadnjem času precej pomemben ne le na ravni organizacij, temveč tudi na ravni nacionalnih ekonomij, v katerih se srečamo z dolžniškimi krizami (Islandija, Grčija idr.).

Prispevek obravnava vpliv stopnje investiranja in zadolževanja za obravnavani poslovni subjekt, ki je lahko podjetje, posameznik ali država (narodno gospodarstvo).

S pomočjo sistemske dinamike želimo razviti model, ki bo omogočal analizo interakcij ključnih pozitivnih povratnih zank in vpliv neustreznega vlaganja v kapitalne investicije, ki ima lahko vzrok v korupciji, na učinkovitost celotnega sistema. Poleg tega želimo model razviti s pomočjo spletnih tehnologij, da bi bilo moč zagotoviti dostopnost za potrebe informiranosti in izobraževanja.

Pomembno je, da določimo ustrezno kombinacijo med investiranjem v nove projekte in zadolževanjem. Pri tem predpostavljamo, da je z ustreznim modelom mogoče ugotoviti tako razmerje med investiranjem in zadolževanjem v posamezni organizaciji, ki zagotavlja stabilno finančno stanje in rast. Z modelom podpremo strategijo podjetja pri investiranju v nove projekte in zagotavljamo večjo uspešnost začrtanega plana ter uspešno delovanje organizacije na dolgi rok. Razviti model je uporaben za vsak subjekt, tako za posameznika, malo ali veliko podjetje kot za državo. Pereč problem pri investicijah je korupcija, ki jo prav tako obravnava razviti model.

Z modelom in pristopom želimo predvsem opisati interakcijo med pozitivnima zankama investicij in zadolževanja ter izpostaviti pomembnost vzdrževanja občutljivega razmerja med količniki, ki vplivajo na odziv celotnega sistema.

Razviti model predstavlja poenostavljeno strukturo investicij in zadolževanja. Pri tem upoštevamo le ključne dejavnike, tako npr. ni vključen mehanizem inflacije. Model vsebuje količnik korupcije, s katerim bomo predpostavili, da se denar prek korupcije nikoli ne vrne v sistem v obliki sistemskega kapitala, čeprav se ta denar vseeno porabi morda kje drugje na svetu (Panama Papers, 2016). Pri tem lahko korupcija privede do kolapsa tudi globalni finančni sistem (ne le posamezno podjetje ali državo).

Pomembno je, da so tovrstni modeli splošno dostopni, kakor tudi razumevanje strukture in interakcije ključnih povratnih zank v sistemu, zato smo se odločili za realizacijo modela in uporabniškega vmesnika s programskim jezikom javascript in označevalnim jezikom HTML.

2 METODE DELA

Pri razvoju modela smo uporabili metodologijo sistemske dinamike (Forrester, 1958; Sterman, 2000; Rahmandad in Sterman, 2012). Obravnavani model vsebuje tako negativne kakor tudi pozitivne povratne zanke. Pozitivna povratna zanka povzroči eksponen-

tno rast tako na strani premoženja kot zadolženosti. Negativna povratna zanka stabilizira stanje, v našem primeru gre pri tem za odplačilo dolga in za odpis vrednosti premoženja. V krogu povratne zanke neparno število negativnih predznakov v posamezni zanki pomeni, da je zanka negativna. Parno število negativnih predznakov v posamezni zanki pomeni, da je zanka pozitivno povratna. Povratno zvezo, ki povečuje vpliv vhodne spremenljivke na izhodno spremenljivko sistema, imenujemo pozitivna, tisto, ki zmanjšuje ta vpliv, pa negativna. Pozitivno povratno zanko v ekonomiji lahko srečamo kot npr. obrestovanje bančne vloge. Negativno povratno zanko srečamo, kjer želimo ohraniti obstoječe stanje oziroma zmanjšati vpliv okolja na obravnavani sistem (Kljajić, 1994). Rahmandad in Sterman (2012) predlagata standardizacijo izvedbe simulacijskih študij v okviru družboslovnih raziskav z osnovnimi gradniki:

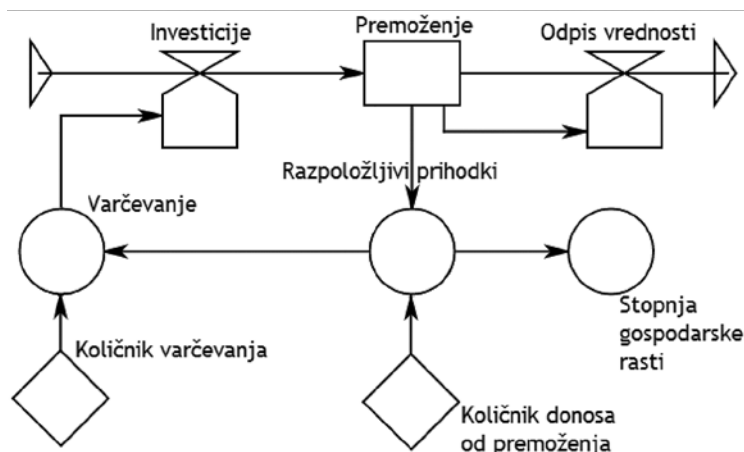
- model: matematični prikaz sistema, ki ga simuliramo, da bi pridobili številsko izražene rezultate;
- vhodni parametri oz. podatki;
- simulacijski tek: posamezna simulacija je sestavljena iz več računskih operacij, ki na podlagi modela generirajo številске rezultate. Različne simulacijske teke razlikujemo med seboj: a) glede na posamezne ponovitve simulacije in b) na simulacijske teke, ki uporabljajo iste parametre in vhodne vrednosti spremenljivk, vendar se razlikujejo zaradi upoštevanja verjetnostnih porazdelitev vrednosti parametrov;
- načrtovanje poizkusov: načrtovanje simulacijskih tekov glede na simulacijske scenarije in število ponovitev simulacijskih tekov, vezano na simulacijske in optimizacijske poizkuse;
- simulacijski poizkus: niz več simulacijskih tekov, ki so bili izvajani in na podlagi katerih smo pridobili rezultate;
- optimizacija poizkusov: kombiniranje rezultatov več simulacij, pri čemer z ustreznim algoritmom poiščemo optimalne vrednosti vhodnih parametrov, ki nam dajo najboljši rezultat. (Rahmandad, Sterman, 2012)

Razviti model smo testirali na različnih scenarijih, ki predstavljajo različne strategije, ki jih načrtuje posamezna organizacija. Vhodni parametri modela so podrobno analizirani. Simulacijski model je bil razvit z orodjem Powersim in nato sprogramiran v obliki spletne aplikacije. Spletna aplikacija je prosto dostopna javnosti (Odperta koda, 2015) na spletnem mestu <http://kozina.eu/fov/model/> (Kozina, 2013–2015).

2.1 Razvoj modela

Pomembnejši prispevek k sodobnim teorijam gospodarske rasti sta dodala R. Harrod ter E. Domar. Njuni teoriji o uravnoteženi in eksponentni rasti sta razloženi tako s strani povpraševanja (Harrod) kot ponudbe (Domar), zato večkrat govorimo kar o Har-

rod-Domarjevem modelu. Osnovni princip modela z vidika vrednosti premoženja je prikazan na sliki 1. Avtorja sta izhajala iz različnih domnev, vseeno pa sta ugotovila, da je stopnja rasti odvisna od razmerja med stopnjo varčevanja in učinkovitostjo gospodarstva (Sušjan, 2002, str. 5–6).



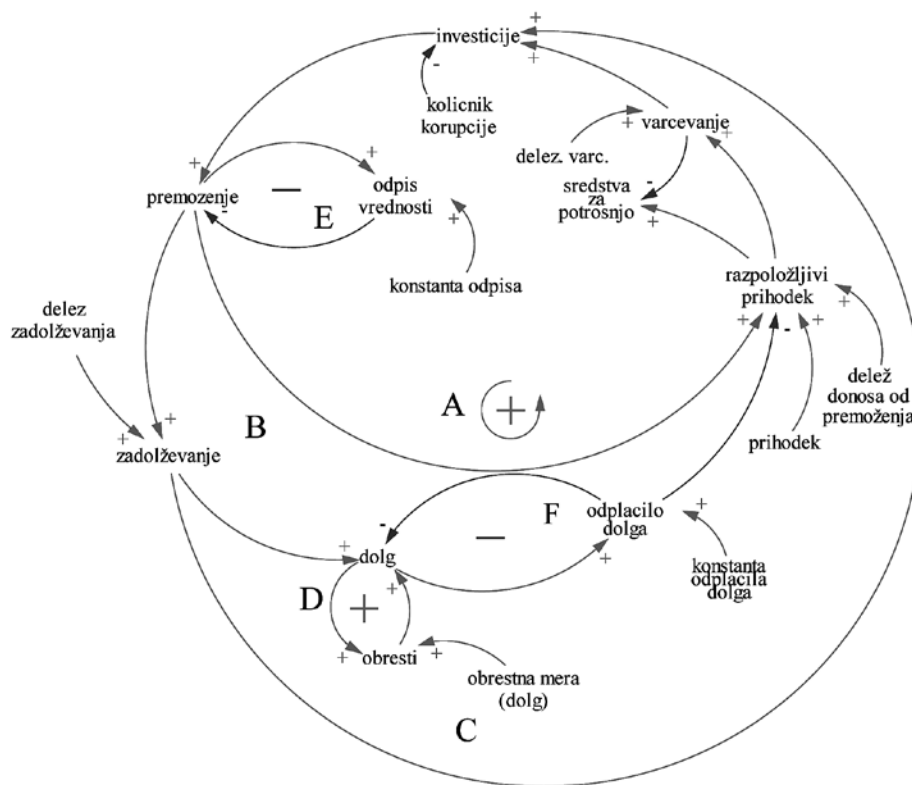
Slika 1: Harrod-Domarjev model ekonomske rasti na strani premoženja (Mojtahedzadeh, 2015)

Model omogoča, da s poizkušanjem najdemo ustrezne vhodne parametre, ki nam omogočajo dober finančni izid. Z modelom sistemske dinamike zmanjšujemo možna tveganja, ker se lahko odločimo na podlagi bolj točnih kvantitativnih meril ob predpostavki, da model vsebuje vse potrebne parametre in da so ocenjeni posamezni parametri točni. Zmanjševanje tveganja v segmentu kreditiranja, ki ga zajema model, mora biti v interesu kreditodajalca kot tudi kreditjemalca.

Celoten model upošteva premoženje, prihodek in zadolževanje. Premoženje povečamo z investicijami, npr. v nepremičnine, z varčevanjem in zadolževanjem, kar poveča obseg našega premoženja. Obseg varčevanja je določen z razpoložljivim prihodkom, ki je odvisen od prihodkov (npr. prihodki od prodaje), količnika donosa od premoženja, obsega premoženja ter odplačila dolga. Koliko sredstev bomo namenili varčevanju, je odvisno od količnika varčevanja. Vrednost premoženja se zmanjšuje z odpisom vrednosti, ki je odvisen od količnika odpisa vrednosti premoženja. Zadolževanje je odvisno od obsega našega premoženja in deleža, za katerega smo se pripravljene zadolžiti, tj. količnika zadolževanja. Zadolževanje prispeva k investicijam in poveča

vrednost našega premoženja. Dolg se povečuje glede na obrestno mero. Odplačilo dolga vpliva na razpoložljivi prihodek. Razlika med razpoložljivim prihodkom in varčevanjem predstavlja sredstva, ki jih lahko potrošimo oz. so na voljo za porabo.

Ključna elementa v vzročno-posledičnem diagramu na sliki 2 sta premoženje, ki ga razumemo kot kapital in dolg. Premoženje povečamo z investicijami, z varčevanjem in zadolževanjem. Obseg varčevanja je določen z razpoložljivim prihodkom, ki je odvisen od plače oz. prihodkov, količnika donosa od premoženja, obsega premoženja ter odplačila dolga. Koliko sredstev bomo namenili varčevanju, je odvisno od količnika varčevanja. Vrednost premoženja se zmanjšuje z odpisom vrednosti, ki je odvisen od količnika odpisa vrednosti premoženja. Zadolževanje je odvisno od obsega našega premoženja in količnika, za katerega smo se pripravljene zadolžiti. Dolg se povečuje glede na obrestno mero, po kateri smo najeli kredit. Odplačilo dolga vpliva na razpoložljivi prihodek. Razlika med razpoložljivim prihodkom in varčevanjem so sredstva, ki jih lahko v obravnavanem časovnem intervalu porabimo za potrošnjo. Količnik korupcije zmanjšuje maso denarja za investicije in s tem moč pozitivne povratne zanke kapitala.



Slika 2: **Vzročno-posledični diagram**

Ključni del sistema sta pozitivni povratni zanki D in A; pri tem moramo zagotoviti dominanco zanke A, sicer dominanca zanke D vodi v finančni zlom.

2.2 Primer Slovenije

V času gospodarske rasti (»obdobje debelih krav«) nismo zmanjšali zadolženosti. Pred uvedbo Družbe za upravljanje terjatev bank (DUTB), tako imenovane »slabe banke«, smo se precej zadolžili po visoki obrestni meri, vendar smo to potrebovali, da je obrestna mera padla.

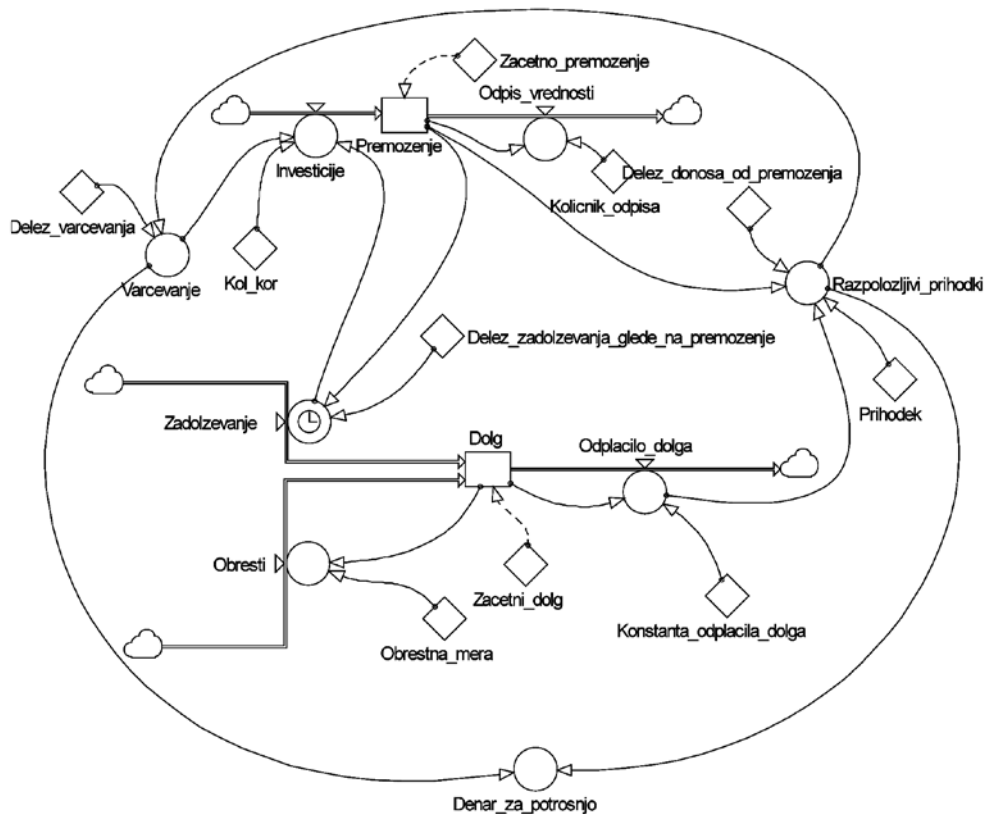
Slovenija se je v letih 2013 in 2014 veliko dodatno zadolževala. Izjave pristojnih leta 2014: »Sedaj se moramo dodatno zadolžiti, da lahko izvedemo reforme, ki nam bodo pomagale pri izhodu iz krize« (V pričakovanju nove zadolžitve, 2014), »Slovenski dolg je bil leta 2014 cca 30 milijard evrov« (Statistični urad, 2015), »Zahtevana donosnost desetletnih slovenskih državnih obveznic je novembra 2011 preseгла 7 odstotkov« (Ali so zahtevane donosnosti, 2015), »Če bi bila obrestna mera za izdane obveznice skozi daljše obdobje 7-odstotna, bi bila Slovenija v veliko težji situaciji, kot je bila nato leta 2015, ko se je stanje

izboljšalo. Uvedba DUTB je eden od razlogov, da je obrestna mera močno upadla. Zahtevana donosnost desetletnih slovenskih državnih obveznic je marca 2015 znašala 0,92 odstotkov« (Rekordno: zahtevani donos, 2015). Zaradi nihanja obrestne mere za najem kredita (izdaje obveznic) je priporočljivo uporabiti scenarij, po katerem se obrestna mera skozi čas spreminja. Tako dobimo bolj realen primer.

Razviti model tako omogoča analizo interakcij med ključnima povratnima zankama premoženja ter zadolženosti in izpostavi pomembnost zagotavljanja ustreznega obsega kapitala in nizkih obrestnih mer pri zadolževanju. Prav tako je moč analizirati vplive korupcije, ki imajo za posledico zmanjšanje obsega premoženja in s tem donosov.

3 REŠITEV, RAZVITA Z ORODJEM POWERSIM CONSTRUCTOR

Slika 3 prikazuje simulacijski model, izdelan z orodjem Powersim Constructor. Model vsebuje stanji, ki predstavljata vrednost premoženja in obseg zadolženosti.



Slika 3: Model, izdelan z orodjem Powersim

4 MODEL KOT SPLETNA APLIKACIJA

Model je implementiran kot spletna aplikacija, ki je prosto dostopna na svetovnem spletu in omogoča vizualizacijo rezultatov s pomočjo izrisanega modela, grafov in podatkov, izpisanih v obliki tabele. Delujoča spletna aplikacija je dostopna na spletnem mestu <http://kozina.eu/fov/model/>.

Tehnološko gledano je aplikacija zasnovana s tehnologijo HTML, predlogo CSS, skriptnim jezikom javascript (knjižnice jQuery, jQuery Mobile, jsCharts) ter vektorsko grafiko SVG za izris modela (knjižnica SVG.JS).

Slogovni jezik CSS je bil uporabljen za oblikovanje predstavitev spletnih strani (CSS, 2015), javascript za realizacijo simulacijskih modelov, SVG za prikaz modela v vektorski grafiki, jQuery in jQuery mobile za vnos parametrov oz. uporabniški vmesnik; uporabniški vmesnik je bil oblikovno prilagojen za uporabo na napravah z zaslonom, občutljivim na dotik (jQuery Mobile, 2015), jsCharts pa je bil uporabljen za izris grafov (JS Charts, 2015). Za prikaz strukture modela v vektorski grafiki smo uporabili programsko knjižnico Svg.js, ki omogoča izdelavo statičnih

in dinamičnih elementov SVG. Knjižnica omogoča, da na preprost in minimalističen način izrišemo na ekran vektorske grafične elemente SVG (A lightweight library, 2015). Aplikacijo je moč uporabiti na delovni postaji, tablici ali pametnem telefonu.

Celotna spletna aplikacija je sestavljena iz štirih datotek, v katerih je okvirno tisoč vrstic programske kode.

4.1 Javascript

Javascript je objektni skriptni programski jezik, ki ga je razvil Netscape, da bi spletnim programerjem pomagal pri ustvarjanju interaktivnih spletnih strani. Jezik je bil razvit neodvisno od jave, vendar si z njo deli številne lastnosti in strukture. Javascript lahko sodeluje s kodo HTML in s tem poživi stran z dinamičnim izvajanjem, v zadnjem času pa omogoča tudi upravljanje strojne opreme, tj. kiberfizičnih sistemov in interneta stvari (Škraba idr., 2015). Javascript podpirajo velika programska podjetja in kot odprti jezik ga lahko uporablja vsakdo, ne da bi za to potreboval licenco. Podpirajo ga vsi novejši spletni brskalniki. Sintaksa jezika javascript ohlapno sledi programskemu jeziku

C (javascript). V našem primeru se je jezik izkazal kot primeren za realizacijo simulacijskih modelov, razvitih po principu sistemske dinamike.

Uporabniški vmesnik omogoča vnos vhodnih parametrov v spletne aplikacije in izris rezultatov v treh oblikah:

- izris modela v vektorski grafiki SVG,
- izris grafov,
- izris podatkov v tabelarni obliki.

4.2 Vnos vhodnih parametrov v model

Model vsebuje enajst vhodnih parametrov, ki jih lahko razdelimo na:

- sedem vhodnih parametrov v obliki količnika,
- tri vhodne parametre v denarni enoti (privzeta denarna enota je evro),
- enoto časa za vnos trajanja simulacije (privzeta enota časa je leto).

Vhodni parametri modela so:

- Čas simulacije je število časovnih ponovitev/intervalov. V našem primeru je možen vnos časa simulacije do največ 600 let, ker je tako nastavljena zgornja omejitev glede možnosti vnosa parametrov preko naprav, občutljivih na dotik s pomočjo drsnikov.
- Količnik varčevanja pove, koliko želimo na leto privarčevati glede na razpoložljivi dohodek. Razpon vrednosti je od 0,000 do 1,000.
- Količnik odpisa nam pove delež denarja, ki ga ne moremo izterjati od dolžnika in ga moramo posledično odpisati. Razpon vrednosti je od 0,000 do 1,000.
- Količnik letne obrestne mere je obrestna mera, po kateri se zadolžimo z najemom kredita ali izdaje obveznic v primeru države. Razpon vrednosti je od 0,000 do 1,000.

- Začetni dolg je znesek kreditov ali izdanih obveznic pred začetkom opazovanega časovnega intervala. Razpon vrednosti je od 0 do 300.000 denarnih enot.
- Količnik korupcije ima negativen predznak – višja kot je korupcija, manjše so investicije. Razpon vrednosti je od 0,000 do 1,000.
- Količnik donosa od premoženja, ki ga vložimo v depozit ali drug produkt, v katerega vložimo denar in na podlagi vložka prejmemo obresti ali dobiček. Razpon vrednosti je od 0,000 do 1,000.
- Količnik zadolževanja glede na premoženje je delež, za koliko se zadolžimo glede na trenutno premoženje. Razpon vrednosti je od 0,000 do 1,000.
- Količnik odplačila dolga je delež vračila v eni enoti časa glede na celotni dolg obravnavanega subjekta. Razpon vrednosti je od 0,000 do 1,000.
- Plača (oz. prihodek) je letni prihodek glede na to, da v uporabljenih scenarijih za časovni interval privzeto uporabljamo enoto leto. Razpon vrednosti je od 0 do 60.000 denarnih enot.
- Začetno premoženje je znesek privarčevanega denarja na začetku opazovanega obdobja. Razpon vrednosti je od 0 do 300.000 denarnih enot.

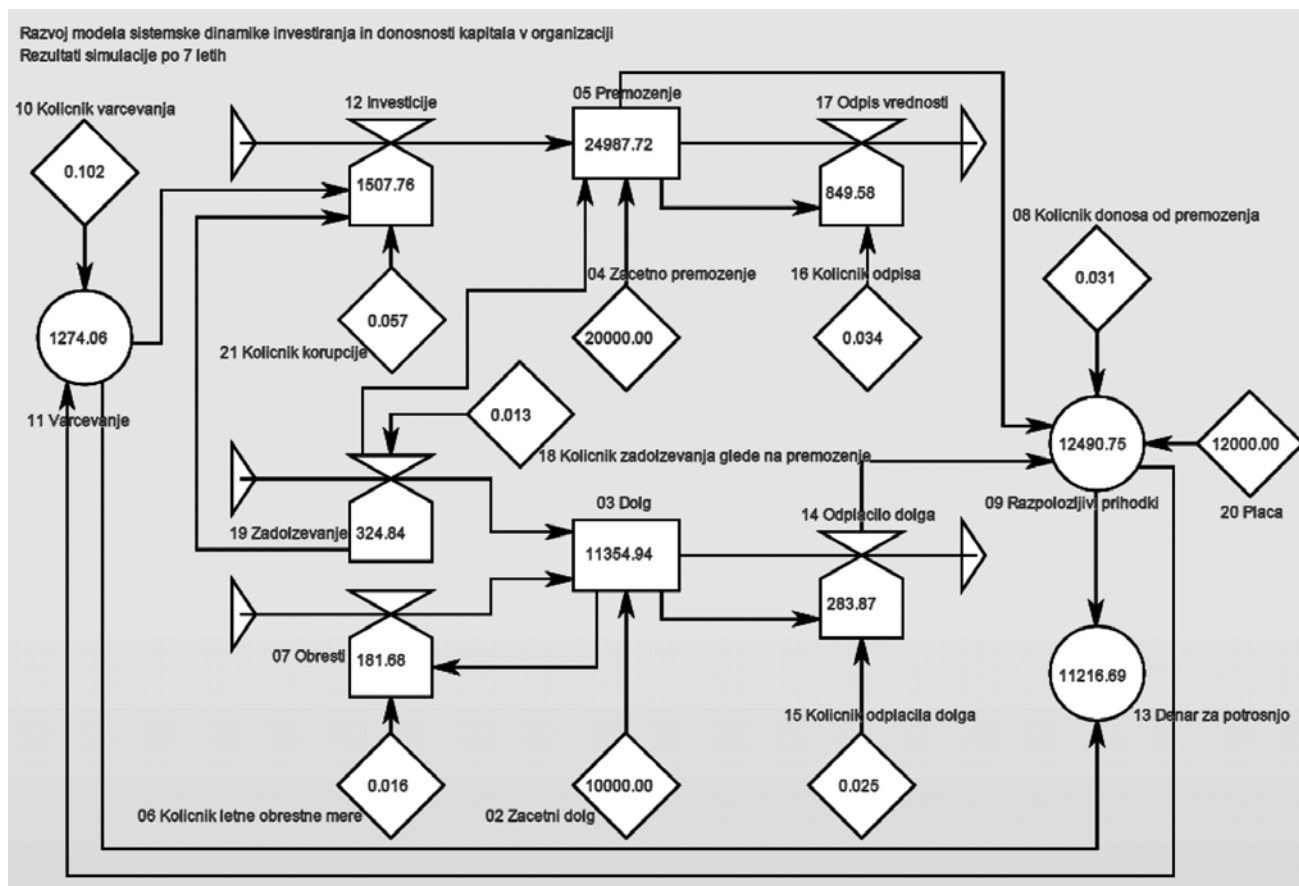
Podatke lahko vnesemo na tipkovnici z vpisom številke ali z drsnikom (z računalniško miško ali s prstom pri napravah, občutljivih na dotik). Uporabniški vmesnik za vnos vhodnih parametrov modela vidimo na sliki 4. Na sliki 5 je prikazan model v vektorski grafiki SVG.

4.3 Izris grafov

Za izris grafov smo uporabili knjižnico jsCharts, ki je razvita s pomočjo tehnologije JavaScript in omogoča preprost izris grafov. V knjižnici jsCharts podamo podatke za izris grafa v obliki XML, JSON ali polju

Parameter	Value
Cas simulacije (v letih)	7
Kolicnik varcevanja	0,102
Kolicnik odpisa	0,034
Kolicnik letne obrestne mere	0,016
Zacetni dolg (v EUR)	10000
Kolicnik korupcije	0,057
Kolicnik donosa od premozenja	0,031
Kolicnik zadolzevanja glede na premozenje	0,013
Kolicnik odplacila dolga	0,025
Letna neto placa (v EUR)	12000
Zacetno premozenje (v EUR)	20000

Slika 4: Uporabniški vmesnik za vnos vhodnih parametrov modela



Slika 5: Izris rezultatov v modelu SVG (Kozina, 2013–2015)

JavaScript (Array). Kreiranje grafov različnih oblik omogoča jsCharts (JS Charts, 2015).

Tabela 2 prikazuje vse vhodne parametre in vse izračunane vrednosti. Vsaka vrstica predstavlja časovni interval. V tabeli so prikazane izračunane vrednosti simulacije od prvega do N-tega časovnega koraka (glede na podani končni čas). Če npr. želimo narediti izračun za tri leta, se izpišejo štiri vrstice, ki predstavljajo začetno (podatki pred prvim letom), prvo, drugo in tretje leto.

4.4 Scenarij začasne spremembe obrestne mere s povrnitvijo v prvotno stanje

Scenarij prikaže vpliv spremembe obrestne mere za najem kredita oz. izdajo obveznic in kako to vpliva na dolgi rok na primeru Republike Slovenije. V času gospodarske rasti (»debelih krav«) v Sloveniji nismo zmanjšali zadolženosti. Pred uvedbo Družbe

za upravljanje terjatev bank (DUTB), t. i. slabe banke, smo se precej zadolžili po visoki obrestni meri, vendar smo to potrebovali, da je obrestna mera padla. Torej je bila odločitev pravilna.

Zaradi nihanja obrestne mere za najem kredita (izdaje obveznic) je priporočljivo uporabiti scenarij, po katerem se obrestna mera skozi čas spreminja. Tako dobimo bolj realen primer, ker se model prilagaja spremembi parametrov.

Obrestna mera za desetletne obveznice Republike Slovenije je za krajši čas zelo zrasla. Najvišja obrestna mera je bila več kot 7 odstotkov, sledil pa je padec pod 1 odstotek. Scenarij prikaže, kako sprememba obrestne mere vpliva na najem kredita/izdajo obveznic in kako se to odraža na dolgi rok. Količnik letne obrestne mere se skozi čas spreminja in na obravnavanem scenariju prikazujemo, kako ta sprememba vpliva na rezultate. Vsa nadaljnja leta po drugem letu (tj. tretje in naslednja leta) je čutiti vpliv začasnega dviga

Tabela 2: **Izpis rezultatov obravnavanega modela v tabelarni obliki**

Čas simulacije (leta)	Zacetni dolg	Dolg	Zacetno premoženje	Premoženje	Količnik letne obrestne mere	Obresti	Količnik donosa od premoženja	Razpoložljivi prihodki	Količnik varčevanja	Varčevanje	Investicije	Denar za potrosnjo	Odplacilo dolga	Količnik odplacila dolga	Količnik odpisa	Opis vrednosti	Količnik zadolževanja glede na premoženje	Zadolževanje	Plača	Količnik korupcije
0	10000	10000	20000	20000	0,016	160	0,031	12370	0,102	1262	1435	11108	250	0,025	0,034	680	0,013	260	12000	0,057
1	10000	10170	20000	20755	0,016	163	0,031	12389	0,102	1264	1446	11125	254	0,025	0,034	706	0,013	270	12000	0,057
2	10000	10348	20000	21495	0,016	166	0,031	12408	0,102	1266	1457	11142	259	0,025	0,034	731	0,013	279	12000	0,057
3	10000	10535	20000	22222	0,016	169	0,031	12426	0,102	1267	1468	11158	263	0,025	0,034	756	0,013	289	12000	0,057
4	10000	10729	20000	22934	0,016	172	0,031	12443	0,102	1269	1478	11174	268	0,025	0,034	780	0,013	298	12000	0,057
5	10000	10930	20000	23632	0,016	175	0,031	12459	0,102	1271	1488	11188	273	0,025	0,034	803	0,013	307	12000	0,057
6	10000	11139	20000	24316	0,016	178	0,031	12475	0,102	1272	1498	11203	278	0,025	0,034	827	0,013	316	12000	0,057
7	10000	11355	20000	24988	0,016	182	0,031	12491	0,102	1274	1508	11217	284	0,025	0,034	850	0,013	325	12000	0,057

Op.: Podatki v belih poljih so vhodni parametri, podatki v sivih poljih so izračunani rezultati.

obrestne mere v drugem letu. Primer velikega nihanja obrestne mere se je zgodil v Sloveniji leta 2011. Uporabimo spletno aplikacijo, v katero trikrat vnesemo parametre. Po koncu vsake stopnje/stopničke rezultat vnesemo kot vhodni parameter v naslednjo stopnjo.

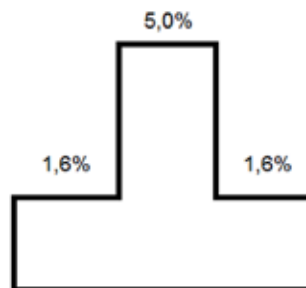
Slika 6 prikazuje primer scenarija, po katerem se skozi tri časovne enote spreminja obrestna mera (stopničasta obrestna mera). V tabeli 3 so podani rezultati scenarija z nespremenljivo kreditno obrestno mero skozi tri časovne intervale (tri leta) in scenarij s stopničasto obrestno mero, po katerem je bila drugo leto obrestna mera višja (5 odstotkov), tretje leto pa se izenači s prvim letom (1,6 odstotka). Iz rezultatov je razvidno, da ima začasno enoletno povišanje obrestne mere v drugem letu posledice tudi v tretjem letu.

4.5 Scenariji z upoštevanjem stopnje korupcije

V tem primeru želimo na modelu prikazati, kakšen je dolgoročni vpliv korupcije na finančno stanje države, podjetja ali posameznika. V primeru korupcije se zmanjša količina oz. obseg sredstev za investicije, kar negativno vpliva na razvoj. Na dolgi rok se učinek korupcije poveča zaradi učinka pozitivne povratne zanke. Izhajali smo iz več virov o stopnji korupcije v Sloveniji. Na podlagi teh ocen je izdelanih več scenarijev. Rezultat je izračun vpliva korupcije na narodno gospodarstvo Slovenije po desetih letih. V našem modelu bomo korupcijo merili s količnikom korupcije, ki zmanjšuje maso denarja, namenjenega za investicije. Denar za investicije pridobimo iz privarčevanih sredstev in z zadolževanjem (kredit).

Primer: če je količnik korupcije 0,1 in namenimo za investicije 100 denarnih enot, je investicija dejansko samo 90 denarnih enot ($100 * (1 - 0,10) = 90$).

Korupcija je še vedno med glavnimi težavami v državah Evropske unije (Panama Papers, 2016), v zadnjih treh letih pa naj bi se njena raven celo povišala, kaže raziskava Eurobarometra, ki jo je objavila Evropska komisija. Korupcija znižuje raven naložb, ovira delovanje notranjega trga in negativno vpliva na javne finance. Gospodarski stroški korupcije v EU so ocenjeni na približno 120 milijard evrov na leto. Kot resno težavo jo v svoji državi najpogosteje zaznavajo Grki (98 odstotkov), najmanj pa je po mnenju vprašanih na Danskem (19 odstotkov). V Sloveniji 95 odstotkov vprašanih meni, da korupcija v naši državi pomeni resno težavo, kar 74 odstotkov pa jih tudi meni, da se je v zadnjih treh letih še povečala (Strošek korupcije v EU je 120 milijard evrov, 2012). V Evropski uniji živi približno 500 milijonov ljudi. Preračunano je torej strošek korupcije v Sloveniji ocenjen na 500 milijonov evrov na leto. Gospodarstvenik Bine Kordež je leta 2015 ocenil, da je v zgodovini samostojne Slovenije izginilo skozi koruptivna dejanja



Slika 6: **Sprememba obrestne mere po letih**

Tabela 3: **Rezultati scenarijev obrestnih mer**

Parameter	Scenarij z nespremenljivo kreditno obrestno mero			Scenarij s stopničasto obrestno mero, po katerem je bila drugo leto obrestna mera višja in se tretje leto izenači s prvim letom		
	1	2	3	1	2	3
Čas simulacije (leta)						
Začetni dolg	10.000	10.170	10.348	10.000	10.170	10.694
Dolg	10.170	10.348	10.535	10.170	10.694	10.877
Začetno premoženje	20.000	20.755	21.495	20.000	20.755	21.495
Premoženje	20.755	21.495	22.222	20.755	21.495	22.220
Količnik letne obrestne mere	0,016	0,016	0,016	0,016	0,050	0,016
Obresti	163	166	169	163	535	174
Količnik donosa od premoženja	0,031	0,031	0,031	0,031	0,031	0,031
Razpoložljivi prihodki	12.389	12.408	12.426	12.389	12.399	12.417
Količnik varčevanja	0,102	0,102	0,102	0,102	0,102	0,102
Varčevanje	1.264	1.266	1.267	1.264	1.265	1.267
Investicije	1.446	1.457	1.468	1.446	1.456	1.467
Denar za potrošnjo	11.126	11.142	11.158	11.126	11.134	11.150
Odplačilo dolga	254	259	263	254	267	272
Količnik odplačila dolga	0,025	0,025	0,025	0,025	0,025	0,025
Količnik odpisa	0,034	0,034	0,034	0,034	0,034	0,034
Odpis vrednosti	706	731	756	706	731	755
Količnik zadolževanja glede na premoženje	0,013	0,013	0,013	0,013	0,013	0,013
Zadolževanje	270	279	289	270	279	289
Plača	12.000	12.000	12.000	12.000	12.000	12.000
Količnik korupcije	0,057	0,057	0,057	0,057	0,057	0,057

10 milijard evrov (Bine Kordež ve, kam je izginilo deset milijard, 2015). Marca 2015 Miha Mazzini navaja, da v Sloveniji po njegovi oceni v korupciji izgine vsaj 30 odstotkov denarja od investicij (Čisto preprosta odločitev, 2015). Vrednost investicij v osnovna sredstva v Sloveniji je letu 2014 znašala 4.990 milijonov evrov (Vrednost investicij, 2015).

Iz podatka o vrednosti investicij v osnovna sredstva v Sloveniji in podatka o znesku korupcije lahko izpeljemo te količnike korupcije:

- Eurobarometer (stopnja korupcije skladna s povprečjem EU), 500 milijonov evrov na leto v Sloveniji. Količnik korupcije v tem primeru je 0,10.
- Kordež, 10 milijard v 20 letih je 500 milijonov evrov na leto, vendar obseg ni enakomeren skozi leta, zato postavimo scenarij na 1 milijardo evrov letno. Količnik korupcije je 0,20.

- Mazzini, 30 odstotnih točk stopnje korupcije od investicij. Torej je po Mazzinijevem scenariju strošek korupcije 1,5 milijarde evrov letno. Količnik korupcije je 0,30.

Strošek zadolževanja, to je plačila obresti od najetih kreditov in izdanih obveznic države, znaša milijardo evrov na leto. Torej obstaja verjetnost, da je strošek korupcije v Sloveniji višji od stroška zadolževanja. Preko obrestnoobrestnega računa pa se skozi daljše obdobje zelo občuti nazadovanje gospodarstva, družbe in države zaradi zmanjšane gospodarske aktivnosti ter rasti in moči gospodarstva. Slovenija ustvari letno 37,303 milijarde evrov bruto družbenega proizvoda (Slovenija, 2015). Uporabimo štiri različne scenarije, s katerimi prikazemo vpliv korupcije na narodno gospodarstvo Slovenije.

V spletno aplikacijo tako lahko vnesemo naslednje parametre:

▪ Čas simulacije (v letih)	10 let
▪ Količnik donosa od premoženja	0,01
▪ Količnik varčevanja	0,05
▪ Količnik zadolževanja glede na premoženje	0,50
▪ Količnik odpisa	0,05
▪ Količnik odplačila dolga	0,20
▪ Količnik letne obrestne mere	0,02
▪ Letna neto plača oz. prihodki v SLO v letu 2014	17 milijard EUR (Stat. urad, 2015)
▪ Začetni dolg (v EUR)	30 milijard EUR (Stat. urad, 2015)
▪ Začetno premoženje (v EUR)	41 milijard EUR

Koliko ima država premoženja? $37.303 \text{ milijonov EUR (BDP)} \times 1,1$
 (bruto premoženje države = 110 % BDP) = 41.033 mio EUR (Kordež, 2014)

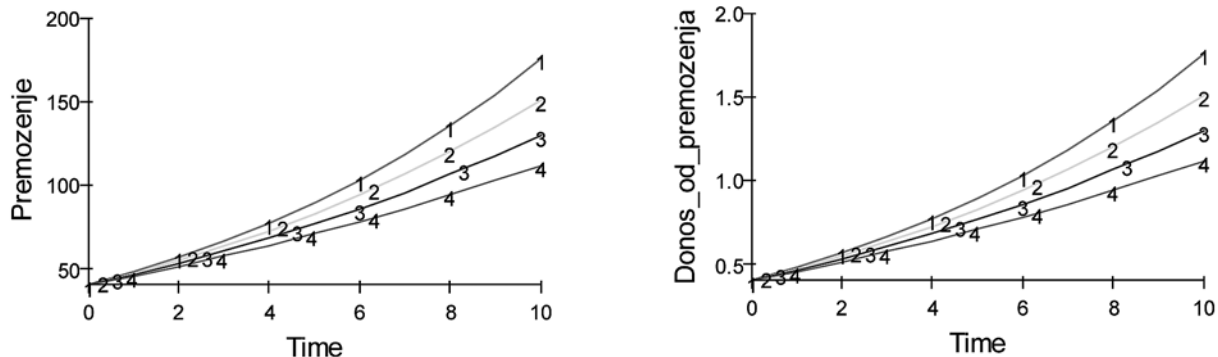
- Za količnik korupcije uporabimo štiri scenarije:
- Brez korupcije: 0 odstotkov,
 - Eurobarometer: 10 odstotkov (500 milijonov EUR od 5 milijard EUR investicij),
 - Kordež: 20 odstotkov (1 milijarda od 5 milijard EUR),
 - Mazzini: 30 odstotkov.
- Rezultati so prikazani v tabeli 4.

Tabela 4: **Scenariji stopnje korupcije**

Parameter	Scenarij korupcija 0 %	Scenarij korupcija 10 %	Scenarij korupcija 20 %	Scenarij korupcija 30 %
01 Čas simulacije (leta)	10	10	10	10
02 Začetni dolg	30,000	30,000	30,000	30,000
03 Dolg	149,130	134,850	122,040	110,560
04 Začetno premoženje	41,000	41,000	41,000	41,000
05 Premoženje	177,930	149,050	124,410	103,440
06 Količnik letne obrestne mere	0,020	0,020	0,020	0,020
07 Obresti	2,980	2,700	2,440	2,210
08 Količnik donosa od premoženja	0,010	0,010	0,010	0,010
09 Razpoložljivi prihodki	3,870	5,010	6,040	6,980
10 Količnik varčevanja	0,050	0,050	0,050	0,050
11 Varčevanje	0,190	0,250	0,300	0,350
12 Investicije	35,780	27,050	20,150	14,730
13 Denar za potrošnjo	3,670	4,760	5,740	6,630
14 Odplačilo dolga	14,910	13,480	12,200	11,060
15 Količnik odplačila dolga	0,100	0,100	0,100	0,100
16 Količnik odpisa	0,050	0,050	0,050	0,050
17 Odpis vrednosti	8,900	7,450	6,220	5,170
18 Količnik zadolževanja glede na premoženje	0,200	0,200	0,200	0,200
19 Zadolževanje	35,590	29,810	24,880	20,690
20 Plača	17,000	17,000	17,000	17,000
21 Količnik korupcije	0,000	0,100	0,200	0,300

Slika 7 prikazuje odziv sistema ob upoštevanju scenarijev 1–4. Predvidevamo, da višja stopnja korupcije prispeva k zmanjšanju vrednosti premoženja

in donosov. Pri tem velja izpostaviti eksponentno rast pri odzivu, kar pomeni, da se na daljši rok negativni učinki vedno večji.



Slika 7: **Odziv sistema; vrednost premoženja (levo) in donos od premoženja (desno)**

5 UGOTOVITVE

S pomočjo modela sistemske dinamike in razvite predstavljene tehnološke rešitve lahko podpremo zagotavljanje ravnovesja, ki je potrebno za trajnostno delovanje in razvoj. Ugotavljamo, da obstajata v obravnavanem modelu dve ključni povratni zanki – pozitivna povratna zanka rasti sistemskega kapitala in pozitivna povratna zanka zadolževanja. Sistem mora biti trajnostno stabilen, kar je moč doseči z uravnoteženjem vpliva omenjenih pozitivnih povratnih zank. V primeru, da sistem ni stabilen, to vodi v stečaj/propad. Z razvitim modelom je moč preveriti, ali je opazovani scenarij stabilen na dolgi rok. Namen prispevka je predvsem pokazati na povezavo med strukturo, povratnimi zankami ter odzivom sistema ter možnost uporabe modela prek svetovnega spleta za izboljšanje strateških odločitev ter razumevanja ključnih zakonitosti. Premoženje vpliva na prihodek od kapitala. Posledično to vpliva na obseg varčevanja, stopnjo investicij in povečanje premoženja. Delež donosa od premoženja tvori pozitivno povratno zanko (Čdobra« zanka). Na drugi strani imamo zadolževanje s pozitivno povratno zanko, ki je delno nezaželena (Čslaba« zanka). Večji kot je dolg, višje so obresti. Ta zanka je odvisna od obrestne mere. Obe zanki sta v idealnem primeru v medsebojnem ravnovesju; zaželeno je, da je pozitivna povratna zanka investicij oz. rasti sistemskega kapitala dominantna. Na sicer občutljivo ravnotežje zank negativno vpliva

količnik korupcije, ki zmanjša moč kapitala podjetja/gospodarstva. Kljub temu ima sistem z ustrezno stopnjo zadolževanja prednosti, saj se v primeru, da se zadolžimo in sredstva vložimo v smislu kapitala, hitreje razvijamo, če je dodana vrednost, ki jo ustvarimo, višja od stroška kredita, ki smo ga najeli za investiranje v razvoj/proizvodnjo. Na dolgi rok ima pravilna odločitev, sprejeta na podlagi izračunov modela, znatne pozitivne učinke. Uporaba modela lahko prepreči bankrot narodnega gospodarstva države in v primeru posameznika ali podjetja prepreči stečaj, ker z modelom poskrbimo za ravnovesje pri investiranju in zadolževanju obravnavanega subjekta. Kljub temu nam v primeru večjih težav bankrot/stečaj omogoča, da smo na dolgi rok uspešni. Izdelani model pomaga odločevalcu, da sprejme odločitve, ki so glede na natančne preračune v tem trenutku optimalni. V primeru, da se čez čas izkaže, da strategija ni več optimalna zaradi sprememb v okolju, je mogoče v nekaterih primerih spremeniti strategijo, in zopet lahko uporabimo izdelani model, s katerim izračunamo nov, potencialno optimalen rezultat glede na nove vhodne parametre. Pogosto se odločamo subjektivno; model nam tako omogoča, da uporabimo argumente, ki so ključni za uspešno odločanje. Pogoji za uvedbo rešitve so usposobljeni kadri in tehnična ter finančna sredstva za nadaljnji razvoj. Kadre je treba ustrezno izšolati za uporabo obravnavanega spletnega orodja. Za uporabo modela moramo zagotoviti le gostovanje spletne

aplikacije v oblaku. V okviru nadaljnega razvoja bi za več manjših nadgradenj morali vložiti štiri človek-mesece dela. V razvoj profesionalnih orodij bi morali vložiti skupno vsaj tri človek-leta (priporočena vključitev dodatnih specialistov za posamezna področja).

5.1 Pogoji za uvedbo

Pogoji za uvedbo rešitve so:

- kadri, ki jim moramo izšolati za uporabo modela;
- tehnična sredstva, ki vključujejo spletni strežnik za objavo spletne aplikacije in delovne postaje za uporabo aplikacije; aplikacija je lahko nameščena tudi lokalno na računalniku, tablici ali pametnem telefonu in tako ne potrebujemo strežnika in dostopa do svetovnega spleta;
- finančna sredstva za nadaljnji razvoj, pri čemer bi za več manjših nadgradenj potrebovali vložiti štiri človek-mesece enote dela. Za razvoj profesionalnih orodij bi morali vložiti razvoj v obsegu vsaj pet človek-let (priporočena vključitev dodatnih specialistov za posamezna področja).

5.2 Možnosti nadaljnega razvoja

Nadaljnji razvoj je možen v več smeri:

- vključitev količnika inflacije v model;
- nadgradnja orodja, razvitega v jeziku javascript, z namenom boljše uporabniške izkušnje;
- uvedba dodatnih parametrov, da bi dobili še bolj realen model;
- uvedba sistema v odločitveni model finančnih ustanov pri presoji kreditnega tveganja (ocena tveganosti stranke);
- uvedba več stopenj scenarija (stopničasti scenarij), da bi model omogočal npr. 100 stopenj ali več, ki bi jih podali kot funkcijo ali kot razpon vrednosti od do z možnostjo različnega koraka od 1;
- razširitev raziskave s časovno daljšimi poskusi;
- vključitev več obravnavanih subjektov (fizičnih oseb, podjetij, neprofitnih organizacij, držav ter drugih vrst skupnosti) za namen validacije modela.

6 LITERATURA

- [1] Panama Papers (2016). <https://panamapapers.icij.org/> Dostop 17. 6. 2016.
- [2] A lightweight library for manipulating and animating SVG. (2015). Dostopno na <http://svgjs.com/>.
- [3] Ali so zahtevane donosnosti na slovenske obveznice previsoke? (2015). Banka Slovenije. Dostopno na <https://www.bsi.si/library/includes/datoteka.asp?Datotekald=4589>.
- [4] Bine Kordež ve, kam je izginilo deset milijard. (2015). Dostopno na http://www.siol.net/novice/slovenija/2015/06/bine_kordez_kam_je_izginilo_deset_milijard_knjiga.aspx.
- [5] Bruto domači proizvod in gospodarska rast. (2015). Statistični urad RS. Dostopno na <http://www.stat.si/statweb/pre-gled-podrocja?idp=29&headerbar=0>.
- [6] Čisto preprosta odločitev. (2015). Dostopno na http://www.siol.net/priloge/kolumne/miha_mazzini/2015/03/cisto_preprosta_odlocitev.aspx.
- [7] Damijan, J. P. (2014). Kako varčevanje ni delovalo v Grčiji. Dostopno na <http://damijan.org/2014/02/04/kako-varcevanje-ni-delovalo-v-grciji/>.
- [8] Evroskupina načeloma odobrila pomoč Grčiji. (2015). Dostopno na http://www.siol.net/novice/svet/2015/07/grcija_reforme_potrditev.aspx.
- [9] BBC. (2015). Greece debt crisis. Dostopno na http://www.bbc.com/news/world-europe-33325886?ocid=global_bbc-com_email_30062015_top+news+stories.
- [10] Integriteta in etika. (2015). Komisija za preprečevanje korupcije. Dostopno na <https://www.kpk-rs.si/sl/korupcija-integriteta-in-etika/integriteta-in-etika>.
- [11] Iz te doline solz nas bo potegnili le gospodarska rast. (2013). Sobotna priloga. 5. 10. 2013.
- [12] jQuery Mobile. (2015). Preneseno s strani <https://jquerymobile.com/>.
- [13] JS Charts. (2015). jsCharts – Free JavaScript charts. Preneseno s strani <http://www.jscharts.com/>.
- [14] Kaj je NIS (2015). NIS – Nacionalni Sistem Integritete. Dostopno na <http://nis.integriteta.si/nacionalni-sistem-integritete/kaj-je-nis>.
- [15] Khalid, S., Škraba, A. (2006). Modeling and experimental analysis of complex systems. Predavanja in rešitve. Worcester: Worcester Politehnika.
- [16] Kljajić, M. (1994). Teorija sistemov. Kranj: Moderna organizacija.
- [17] Kordež, B. (2014). Premoženska slika Slovenije glede na druge države. Dostopno na <http://kordez.blog.drugisvet.com/2014/09/01/premozenjska-slika-slovenije-glede-na-druge-drzave/>.
- [18] Korupcija je. (2015). Komisija za preprečevanje korupcije. Dostopno na <https://www.kpk-rs.si/sl/korupcija-integriteta-in-etika/korupcija-je>.
- [19] Kozina, D. (2013–2015). Razvoj modela sistemske dinamike investiranja in donosnosti kapitala v organizaciji. JavaScript spletna aplikacija. Dostopno na <http://kozina.eu/fov/>.
- [20] Kozina, D., Škraba, A. (2016). Razvoj modela sistemske dinamike investiranja in donosnosti kapitala v organizaciji. Referat na 35. mednarodni konferenci o razvoju organizacijskih znanosti, Portorož 2016. Dostopno na http://kozina.eu/fov/doc/Referat-Portoroz-2016-Damijan-Kozina_20160211.doc.
- [21] Kurdija, S. (2009). Stališča o korupciji 2009, raziskava, str. 4, Komisija za preprečevanje korupcije Republike Slovenije, Fakulteta za družbene vede – Inštitut za družbene vede, Center za raziskovanje javnega mnenja in množičnih komunikacij.
- [22] Mojtahedzadeh, M. (2015). Understanding Economic Dynamics. Dostopno na http://www.iseesystems.com/community/connector/Zine/may-june_2003/economicdynamics.html.
- [23] Powersim Software AS. (2015). Uradna spletna stran podjetja. Dostopno na <http://www.powersim.com/>.
- [24] Rahmandad, H., Sterman, J. (2012). Reporting guidelines for simulation-based research in social sciences. Syst. dyn. rev., Vol. 28, No. 4, str. 396–411.

- [25] Rekordno: zahtevani donos na slovenske obveznice pod enim odstotkom. (2015). RTV Slovenija. Dostopno na <https://www.rtvsllo.si/gospodarstvo/rekordno-zahtevani-donos-na-slovenske-obveznice-pod-enim-odstotkom/359864>.
- [26] siol.net. (2015). Tuji lastniki in oživitvev STO rešitev za slovenski turizem. Dostopno na http://www.siol.net/novice/gospodarstvo/2013/11/zadolzenost_slo_turizma.aspx.
- [27] Slovarji Inštituta za slovenski jezik Frana Ramovša ZRC SAZU. (2015). Dostopno na <http://fran.si/>.
- [28] Slovenski turizem duši odplačevanje posojil. (2015). Dostopno na http://www.siol.net/novice/gospodarstvo/2013/10/studija_slovenski_turizem_posojilo.aspx.
- [29] Stališča do pojava korupcije v Sloveniji. (2015). Komisija za preprečevanje korupcije. Dostopno na <https://www.kpk-rs.si/sl/korupcija-integriteta-in-etika/analize-raziskave-in-statistika/raziskave-in-statistika/>.
- [30] Statistični urad. (2015). Neto posojanje/neto izposojanje in konsolidirani bruto dolg sektorja država, Slovenija, letno. Statistični urad RS. Dostopno na http://pxweb.stat.si/pxweb/Dialog/varval.asp?ma=0314905S&ti=&path=../Database/Ekonomsko/03_nacionalni_racuni/25_03149_racuni_drzave/&lang=2.
- [31] Sterman, J. (2000). *Business Dynamics: Systems Thinking and Modeling for a Complex World*. McGraw-Hill/Irwin: Boston, MA.
- [32] Strošek korupcije v EU je 120 milijard evrov. (2012). Časnik Finance, 032/2012, 16. 2. 2012. Dostopno na <http://www.finance.si/340607/Stro%C5%A1ek-korupcije-v-EU-je-120-milijard-evrov>.
- [33] Sušjan, A. (2002). *Teorija ekonomske rasti*. Klasična, neoklasična, keynesianska. Ljubljana: Ekonomska fakulteta.
- [34] Škraba, A., Kljajić, M., Leskovar, R. (2003). Group exploration of system dynamics models - is there a place for a feedback loop in the decision process?. *Syst. dyn. rev.*, Vol. 19, No. 3, str. 243–263.
- [35] Škraba, A., Stojanović, R., Zupan, A., Koložvari, A., Kofjač, D. (2015). Speech-controlled cloud-based wheelchair platform for disabled person. *Microprocessors and Microsystems*, Vol. 39/8, str. 819–828.
- [36] Teorija iger in grško-evropska praksa. (2015). Dostopno na http://www.siol.net/priloge/kolumne/ziga_turk/2015/07/teorija_iger_in_grsko_evropska_praksa.aspx.
- [37] Trgovec Franc Jager: Kako sem prišel do 35 milijonov evrov. (2014). Dostopno na http://www.siol.net/novice/rubrikon/siolov_intervju/2014/01/intervju_franc_jager.aspx.
- [38] V pričakovanju nove zadolžitve. (2014). Radio Slovenija 1. Studio ob 17:00. 31. 1. 2014.
- [39] Vizualno modeliranje. (2012). Kranj: Fakulteta za organizacijske vede. Laboratorij za kibernetiko in sisteme za podporo odločanju. Dostopno na <http://kibernetika.fov.uni-mb.si/vim/>.
- [40] Vrednost investicij (2015). Statistični urad RS. Dostopno na <http://www.stat.si/StatWeb/prikazi-novico?id=5658&idp=16&headerbar=14>.
- [41] Zadolženost Slovenije 1993–2014. (2015). Dostopno na <https://twitter.com/kricac/status/433218030621827074/>, 11. 2. 2014, Denis Oštir.

Damijan Kozina je magistriral na področju informacijskih sistemov na Fakulteti za organizacijske vede Univerze v Mariboru. Zaposlen je v UniCredit Banki Slovenije kot razvijalec in skrbnik programske opreme in podatkovnih zbirk ter analitik kakovosti podatkovnih zbirk.

Davorin Kofjač je docent za področje informacijskih sistemov na Fakulteti za organizacijske vede Univerze v Mariboru v Laboratoriju za kibernetiko in sisteme za podporo odločanju. Njegova področja raziskovanja obsegajo modeliranje in simulacijo sistemov, sisteme za podporo odločanju, operacijske raziskave in umetno inteligenco. Sodeloval je v raznih mednarodnih in nacionalnih projektih. Rezultate raziskovalnega dela objavlja v uglednih znanstvenih revijah, monografijah in na konferencah. Je član društev ACM, INFORMS in SLOSIM.

Andrej Škraba je izredni profesor za področje teorije sistemov ter modeliranja in simulacije na Fakulteti za organizacijske vede Univerze v Mariboru v Laboratoriju za kibernetiko in sisteme za podporo odločanju. Področja njegovega dela so sistemska dinamika, kibernetika, sistemi za podporo odločanju ter kiberfizični sistemi. Je član DSI, SLOSIM in System Dynamics Society (SDS).

Primerjava zmogljivosti večplatformsko razvitih mobilnih aplikacij

Boris Ovcjak, Tatjana Welzer Družovec, Gregor Polančič, Univerza v Mariboru, Fakulteta za elektrotehniko, računalništvo in informatiko, Inštitut za informatiko, Smetanova 17, 2000 Maribor
boris.ovcjak@um.si; tatjana.welzer@um.si; gregor.polancic@um.si

Izvleček

Prispevek prikazuje rezultate primerjalne analize zmogljivosti večplatformsko razvite mobilne aplikacije za iOS, Android in Windows Phone. Analiza je namenjena ocenitvi primernosti uporabe izbranih razvojnih ogrodij. V ta namen sta bila izbrana dva razvojna pristopa (hibridni in večplatformsko prevedeni) ter pripadajoča ogrodja (Ionic in Xamarin). Implementirana vzorčna aplikacija je bila na podlagi petih testov zmogljivosti primerjana z aplikacijo, razvito z domorodnim pristopom. Rezultati testov so pokazali primernost uporabe večplatformsko prevedenega pristopa, saj se je ta na podlagi izbranega ogrodja domorodnemu pristopu precej približal, medtem ko se je hibridni pristop predvsem v strojno zahtevnejših opravilih izkazal za neučinkovitega.

Ključne besede: mobilne aplikacije, mobilne platforme, večplatformski razvoj, zmogljivost, hibridne aplikacije, večplatformsko prevedene aplikacije, Xamarin, Ionic.

Abstract

Performance comparison of cross-platform mobile applications

The article presents the results of a performance based comparative analysis of cross-platform developed mobile application for iOS, Android and Windows Phone. The analysis is intended to assess the appropriateness of use of the selected development frameworks. To this end, two development approaches (hybrid and cross-compiled) and associated frameworks (Ionic and Xamarin) were selected. The implemented sample application was compared with the native application on the basis of five performance tests. Test results have shown the suitability of cross-compiled approach, since the selected frameworks' performance came close to the native approach, while the hybrid approach proved ineffective especially in regard to in complex hardware tasks.

Keywords: mobile applications, mobile platforms, cross-platform development, performance, hybrid applications, cross-compiled applications, Xamarin, Ionic.

1 UVOD

Trg mobilnih naprav ter s tem povezanih mobilnih aplikacij je že nekaj časa v polnem razcvetu. Leta 2017 je predvideno 4,77 milijarde mobilnih uporabnikov (Statista, 2016b), pomembne rasti pa je deležen predvsem mobilni podatkovni promet, ki je leta 2015 zrasel za 74 odstotkov v primerjavi z letom 2014 (Cisco, 2016). Vsi ti podatki kažejo na razširjenost mobilnih naprav za poslovno in vsakodnevno uporabo. Tega se zavedajo tudi razvijalci, saj število aplikacij, ki so na razpolago v aplikacijskih trgovinah, nenehno raste. Tri izmed najbolj razširjenih platform, Android, iOS in Windows Phone, skupaj dosegajo že skoraj pet milijard aplikacij (Android 2,2 milijarde, iOS 2 milijardi in Windows Phone 0,67 milijarde) (Statista, 2016a). Prav porazdeljenost platform je za razvi-

jalce eden večjih izzivov, saj je treba tako naloge razvoja kot tudi posodabljanja in vzdrževanja aplikacij opravljati na več platformah (Joorabchi, Mesbah in Kruchten, 2013). To zahteva večkratno izvajanje enakih nalog, kar vpliva tako na čas razvoja kot na s tem povezane stroške.

Omenjene težave pri razvoju aplikacij za več platform so razvijalce prisilile v iskanje alternativnih, učinkovitejših večplatformskih razvojnih tehnik. Trenutno je na voljo več pristopov, ki ob prihranjenem času in stroških zahtevajo določene kompromise, povezane z izgledom, razpoložljivostjo funkcionalnosti mobilne naprave ter učinkovitostjo delovanja aplikacij. Zaradi tega izbira večplaformskega razvojnega pristopa predstavlja svojevrsten izziv, s katerim

se spopadajo tudi raziskovalci, ki skušajo razvijalcem olajšati izbiro s predstavitvijo in primerjavo obstoječih pristopov tako s teoretičnega (znanstvenega) (R. M. de Andrade, B. Albuquerque, F. Frota, V. Silveira in A. da Silva, 2015; Charkaoui, Adraoui in Habib Benlahmar, 2014; Palmieri, Singh in Cicchetti, 2012; Smutný, 2012; Adinugroho in Gautama, 2015; Xanthopoulos in Xinogalos, 2013) kot tudi s strokovnega (praktičnega) stališča. Raziskave se pri tem usmerjajo v analize različnih vidikov večplatformskega razvoja (Dalmasso, Datta, Bonnet in Nikaein, 2013; Danielsson, Ameri, Friberg, Examiner in Lindell, 2014). Ob že uveljavljenih razvojnih pristopih raziskovalci razvijajo tudi nove, ki temeljijo na obstoječih, oz. razvijajo nove alternativne (Perchat, Desertot in Lecomte, 2013; Bouras, Papazois, & Stasinou, 2014 El-Kassas, Abdullah, Yousef in Wahba, 2014). Glede na količino različnih ogrodij in pristopov se raziskovalci posvečajo tudi izdelavi platform za ocenjevanje večplatformskih mobilnih ogrodij (Dhillon, 2015).

V članku bomo predstavili različne vrste večplatformskega razvoja, njihove lastnosti, prednosti in slabosti. Glede na razpoložljivost različnih ogrodij in platform se bomo natančneje osredinili predvsem na tiste, ki so razvijalcu v polni funkcionalnosti na voljo brezplačno in je z njimi mogoče razviti mobilno aplikacijo za platforme iOS, Android in Windows Phone. Za vsakega izmed preučenihih pristopov bomo podrobneje predstavili najbolj razširjenega predstavnika ter na podlagi vzorčne aplikacije s primerjalno analizo zmogljivosti ocenili primernost uporabe ogrodij v primerjavi z domorodnim razvojem.

2 PRISTOPI RAZVOJA VEČPLATFORMSKIH APLIKACIJ

Cilj večplatformskega razvoja mobilnih aplikacij je zagotoviti enovit pristop k razvoju mobilnih aplikacij za več platform. Poleg domorodnega načina razvoja aplikacij, ki zahteva ločen razvoj za vsako izmed mobilnih platform, je to mogoče zagotoviti na različne načine. Vsi omogočajo z enim samim razvojnim okoljem ter programskim jezikom razviti mobilno aplikacijo, ki jo je mogoče poganjati na različnih platformah. To je pomembno, saj se ob enkratni izbiri razvojnega ogrodja razvijalcem ni več treba ukvarjati z učenjem novih tehnologij oz. novih programskih jezikov za vse platforme, saj lahko uporabijo že obstoječe znanje. Glede na raznolikost načinov večplatformskega razvoja obstajajo različne prednosti in slabosti nači-

na razvoja. V splošnem večplatformski razvoj lahko prinese prednosti, kot so: 1) hitrejši čas razvoja, 2) nižji stroški razvoja, 3) ponovna uporaba programske kode, 4) lažji razvoj, 5) uporaba vtičnikov, 6) večji doseg tržišča ter 7) preprosto in hitro podajanje posodobitev. Omenjene prednosti pa lahko zahtevajo tudi določene kompromise, med katere štejemo: 1) manjšo kakovost 3D vsebin in grafike, 2) omejitve platforme (nezmožnost dostopanja do vseh funkcionalnosti platforme), 3) ni zagotovljene optimalne uporabniške izkušnje, 4) težave pri integraciji s platformo ter 5) počasnejše delovanje aplikacij (Rajput, 2016).

Težave, ki jih prinaša porazdeljenost mobilnih platform, so povzročile razcvet različnih kategorij večplatformskih pristopov, ki se med sabo razlikujejo tako po arhitekturi, načinu razvoja, uporabljenih tehnologijah ter razvojnih orodjih. V splošnem načine razvoja mobilnih aplikacij delimo na pet skupin (Rahul Raj in Toley, 2012; Xanthopoulos in Xinogalos, 2013).

- **Domorodne aplikacije** (angl. Native applications) pomenijo privzet način razvoja mobilnih aplikacij in so specifične za vsako mobilno platformo, torej uporabljajo razvojna orodja in programski jezik, ki ga določa specifična platforma. Običajno tovrstne aplikacije zagotavljajo največjo podporo funkcionalnostim naprave ter omogočajo aplikaciji specifičen izgled ter optimalno delovanje (Korf in Oksman, 2016).
- **Spletne mobile aplikacije** (angl. mobile web applications) so aplikacije, razvite z namenom izvajanja v brskalniku mobilne naprave. Razvite so s pomočjo spletnih tehnologij (HTML5, CSS3 in JavaScript). Za izvajanje tovrstnih aplikacij mobilna naprava ne potrebuje posebej nameščene aplikacije, saj se vsa vsebina prikazuje v brskalniku naprave.
- **Hibridne aplikacije** (angl. hybrid applications) so kombinacija spletnega in domorodnega razvojnega pristopa. Razvite so s pomočjo spletnih tehnologij in se na mobilni napravi izvajajo znotraj domorodne komponente WebView. Ta je skupna vsem mobilnim platformam in omogoča prikaz vsebine HTML na celoten zaslon naprave. Takšen način razvoja omogoča dostop do funkcionalnosti naprave preko abstrakcijskega nivoja, ki zmožnosti naprave izpostavlja v obliki JavaScript API (angl. Application Programming Interface). Hibridni pristop lahko uporabljamo tako za strežniške kot za samostojne spletne aplikacije. Za razliko od spletnih mobilnih aplikacij moramo

tovrstne aplikacije prenesti na telefon prek aplikacijskih trgovin.

- **Interpretirane aplikacije** (angl. Interpreted applications) delujejo tako, da se aplikacijska koda postavi na mobilno napravo in se interpretira šele v času izvajanja aplikacije. Aplikacija nato komunicira z abstrakcijskim nivojem, prek katerega dostopa do domorodnega API-ja, kar ji omogoča dostop do naprednejših funkcionalnosti mobilne naprave. Tako razvite aplikacije omogočajo uporabo platformsko specifičnih elementov za izgradnjo uporabniškega vmesnika, medtem ko je aplikacijska logika zajeta na način, neodvisen od platforme. To je lahko v obliki množice ukazov v XML ali v katerem drugem opisnem jeziku.
- **Večplatformsko prevedene aplikacije** (angl. cross-compiled applications) temeljijo na prevajalniku, ki omogoča pretvorbo izvorne v domorodno binarno kodo. Prevajalnik je v tem primeru odgovoren za generiranje in izvajanje programske kode za specifično platformo (Xamarin, 2016). Tako lahko razvijalec napiše izvorno kodo v enotnem skupnem programskem jeziku, ki ga prevajalnik nato prevede v domorodno kodo, določeno pri ciljni platformi. V tem primeru je celoten pristop odvisen od učinkovitosti in zanesljivosti prevajalnika.

Tabela 1 prikazuje različne pristope večplatformskega razvoja, pripadajoča ogrodja, njihovo dostopnost ter njihove prednosti in izzive (Rahul Raj in Tolety, 2012).

Tabela 1: **Pristopi večplatformskega razvoja mobilnih aplikacij (Rahul Raj in Tolety, 2012)**

Pristop	Ogrodja	Prednosti	Izzivi
Domorodne	Specifična glede na platformo (brezplačno)	<ul style="list-style-type: none"> – Domoroden uporabniški izgled in delovanje – Dostop do vseh funkcionalnih lastnosti platforme – Hitrost izvajanja 	<ul style="list-style-type: none"> – Razvoj aplikacij je različen za vsako mobilno platformo.
Spletne	Adobe Air (brezplačno)	<ul style="list-style-type: none"> – Instalacija aplikacije na mobilno napravo je nepotrebna. – Podatki aplikacije se hranijo na spletnem strežniku, kar lajša posodabljanje aplikacije. – Enoten uporabniški vmesnik 	<ul style="list-style-type: none"> – Nezmožnost distribucije prek aplikacijskih trgovin, kar lahko negativno vpliva na popularnost aplikacije – Odvisnost od hitrosti povezave s spletom – Nezmožnost dostopanja do funkcionalnosti in strojne opreme mobilne naprave – Potreba po upoštevanju velike količine različnih resolucij – Manjši nadzor mobilnega brskalnika nad prikazovanjem vsebin – Omejena podpora gestam (angl. Gestures) – Težja monetizacija aplikacije
Hibridne	Cordova (brezplačno), PhoneGap (brezplačno), MoSync (brezplačno), RhoMobile (brezplačno z omejitvami)	<ul style="list-style-type: none"> – Distribucija prek aplikacijskih trgovin – Ponovna uporaba uporabniškega vmesnika prek različnih platform – Dostop do funkcionalnosti in zmožnosti mobilne naprave 	<ul style="list-style-type: none"> – Slabša učinkovitost delovanja v primerjavi z domorodnimi aplikacijami, saj se izvajanje dogaja v komponenti brskalnika – Zaradi uporabe JavaScript programskega jezika so aplikacije izpostavljene ranljivostim CSC (angl. Cross space communication). – Ponovna uporaba uporabniškega vmesnika ne ponuja domorodnega izgleda aplikacije. To lahko dosežemo z oblikovanjem, specifičnim za vsako platformo.
Interpretirane	Titanium (plačljiv, brezplačno dostopen osnovni SDK), React Native (brezplačno)	<ul style="list-style-type: none"> – Prinaša izgled in občutek domorodne mobilne aplikacije. – Omogoča prenos poslovne logike med platformami. – Distribucija aplikacij poteka prek aplikacijskih trgovin. – Strojna oprema in lastnosti naprave so zapakirane znotraj specifičnega API-ja. 	<ul style="list-style-type: none"> – Ponovna uporaba uporabniškega vmesnika je odvisna od abstrakcije na nivoju ogrodja. – Razvoj je odvisen od množice funkcij, ki jih omogoča ogrodje. – Učinkovitost delovanja je slabša zaradi interpretiranja programske kode v času delovanja.
Večplatformsko prevedene	Xamarin (brezplačno), Marmelade (brezplačno z omejitvami), JavaFxPorts (brezplačno)	<ul style="list-style-type: none"> – Ponuja vse funkcionalnosti, ki jih omogoča domorodna aplikacija. – Omogočen dostop do strojne opreme mobilne naprave – Zmožnost uporabe vseh vmesniških komponent – Glavno prednost predstavlja ustrezna učinkovitost delovanja. 	<ul style="list-style-type: none"> – Ne omogoča ponovne uporabe uporabniškega vmesnika ter platform specifičnih lastnosti, kot so kamera, lokacijske storitve, lokalna obvestila ipd. Ker so te funkcionalnosti odvisne od platforme, se način dostopa razlikuje med platformami.

Iz tabele je razvidno, da smo ne glede na izbrani pristop soočeni z določenimi izzivi, ki jih ta prinaša. V povezavi s tem je bil glavni namen članka z vidika zmogljivosti aplikacije primerjati trenutno najbolj aktualna večplatformska ogrodja z domorodnim pristopom. Pri tem smo se omejili na brezplačno dostopna ogrodja, ki omogočajo razvoj mobilnih aplikacij za izbrane tri mobilne platforme. Zaradi tega smo iz primerjalne analize izključili pristop mobilnih spletnih aplikacij, saj se tovrstne aplikacije lahko izvajajo izključno le v brskalniku mobilne naprave. V podrobnejšo analizo prav tako nismo vključili interpretiranega pristopa, saj oba izmed pregledanih predstavnikov ne podpirata trenutno najbolj razširjenje verzije sistema Windows Phone, uporabljene v tej analizi (Windows Phone 8.1) (NetMarketShare, 2016), prav tako pa platforma Titanium v uradni obliki ni na voljo brezplačno (Titanium, 2016).

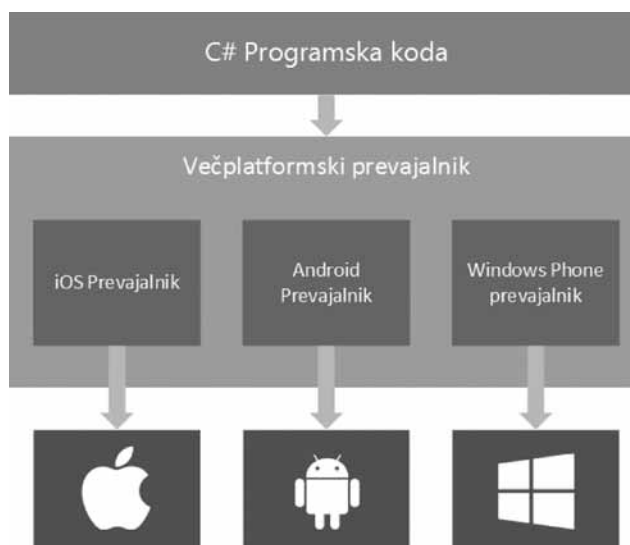
Ob upoštevanju omejitev se bomo v članku osredinili na pristop večplatformsko prevedenih in hibridnih aplikacij. Glede na to, da oba pristopa vključujeta zbirko različnih ogrodij, smo v nadaljevanju izbrali po enega predstavnika za vsak tip razvoja. V primeru večplatformskega pristopa smo se odločili za ogrodje Xamarin, saj predstavlja enega izmed najbolj celovitih ogrodij, podprtega s strani podjetja Microsoft. Hibridni pristop, za katerega smo se prav tako odločili, pa je predvsem zaradi priljubljenosti spletnih tehnologij, na katerih temelji, precej razširjen. Za potrebe študije smo ogrodje izbrali na podlagi priljubljenosti med razvijalci, in sicer ogrodje Ionic (Noeticforce, 2016; Markov, 2015). Oba izbrana predstavnika sta na kratko predstavljena v nadaljevanju.

2.1 Predstavniki večplatformsko prevedenega pristopa Xamarin

Xamarin je odprtokodna platforma, ki jo ponuja podjetje Microsoft in omogoča izgradnjo večplatformsko prevedenih mobilnih aplikacij na deljeni programski bazi z uporabo enotnega razvojnega okolja, programskega jezika in API-ja. Podjetje Xamarin je ustanovil Miguel de Icaza 16. maja 2011 (Allen, 2011) z namenom izgradnje mobilnih aplikacij na podlagi ogrodja Mono, ki omogoča večplatformsko izvajanje aplikacij .NET. Uporaba platforme je bila do nedavnega plačljiva, kar pa se je spremenilo, ko je 24. februarja 2016 podjetje prevzel Microsoft (Guthrie, 2016). Dober mesec po prevzemu je namreč podjetje Microsoft objavilo, da je Xamarin SDK postal odprto-

kodno ogrodje na voljo brezplačno znotraj integriranega razvojnega okolja Visual Studio.

Ogrodje Mono .NET, na katerem temelji platforma Xamarin, omogoča implementacijo širokega spektra ogrodja .NET, ki ga s programskim jezikom C# lahko poganjamo na več mobilnih platformah, in sicer iOS, Android ter Windows Phone. Razvoj za vsako izmed omenjenih platform je podprt prek paketov za razvoj programske opreme (angl. SDK), ki omogočajo povezavo do večine platform specifičnih funkcionalnosti (Xamarin, 2016a). SDK je specifičen tako za Android (Xamarin.Android SDK) kot za iOS (Xamarin.iOS SDK), medtem ko za platformo Windows Phone ni potreben. Glede na platformo, za katero razvijamo aplikacijo, prevajalnik Xamarin zgradi domorodno aplikacijo, kar pa zaradi raznolikosti ciljnih platform poteka na različne načine (slika 1).



Slika 1: Shema pristopa večplatformsko prevedene aplikacije

V primeru platforme iOS se C# programska koda vnaprej prevede v zbirni jezik ARM (angl. Assembly language), v katerega se vključi ogrodje .NET. Na drugi strani se v primeru platforme Android C# prevede v vmesni jezik (angl. Intermediate language), ki se zapakira v Mono virtualno okolje (MonoVM). Končna aplikacija se tako izvaja vzporedno z javanskim ali android izvajalnim okoljem (angl. Runtime environment) in komunicira z domorodnimi tipi prek domorodnega vmesnika Java (angl. Java native interface). V primeru obeh platform se pred vključitvijo ogrodja .NET temu prej odstranijo nerabljeni

razredi v namen zmanjšanja velikosti končne aplikacije. V primerjavi z omenjenima platformama je postopek priprave aplikacije za platformo Windows Phone enostavnejši, saj za delovanje ne potrebuje

ogrodja Xamarin, a je ta vseeno priporočljiv zaradi možnosti ponovne uporabe programske kode (Xamarin, 2016b). To je v primeru ogrodja Xamarin mogoče doseči na več načinov (tabela 2).

Tabela 2: **Metode ponovne uporabe programske kode**

Metoda ponovne uporabe programske kode	Opis
Xamarin.Mobile	Enoten API za dostop do skupnih virov mobilne naprave za vse vrste razvojnih platform (Xamarin, 2016d)
Deljeni projekt (angl. Shared Asset Project)	Projekt, namenjen organizaciji izvorne kode z uporabo direktiv na ravni prevajalnika za upravljanje platformsko specifičnih delov programske kode
Prenosljive razredne knjižnice (angl. Portable Class Libraries)	Projekt, namenjen izgradnji knjižnic za podprte platforme, ki za dostop do platformsko specifičnih funkcionalnosti uporabljajo vmesnike
Xamarin.Forms	Strani, napisane z uporabo programskega jezika C# ali XAML, ki se v času izvajanja skupaj z vsebujočimi kontrolami preslikajo v platformsko specifične vizualne elemente Predvsem so namenjene aplikacijam, osredinjenim na vnos podatkov.

Z uporabo različnih metod ponovne uporabe programske kode lahko dosežemo do 90 % deljene programske kode (Xamarin, 2016a).

Za razvoj aplikacij ogrodje ponuja dve programski orodji, in sicer Visual Studio (Windows) ter Xamarin Studio (Windows, macOS), ki glede na operacijski sistem, v katerem razvijamo, podpirata razvoj različnih mobilnih platform (tabela 3).

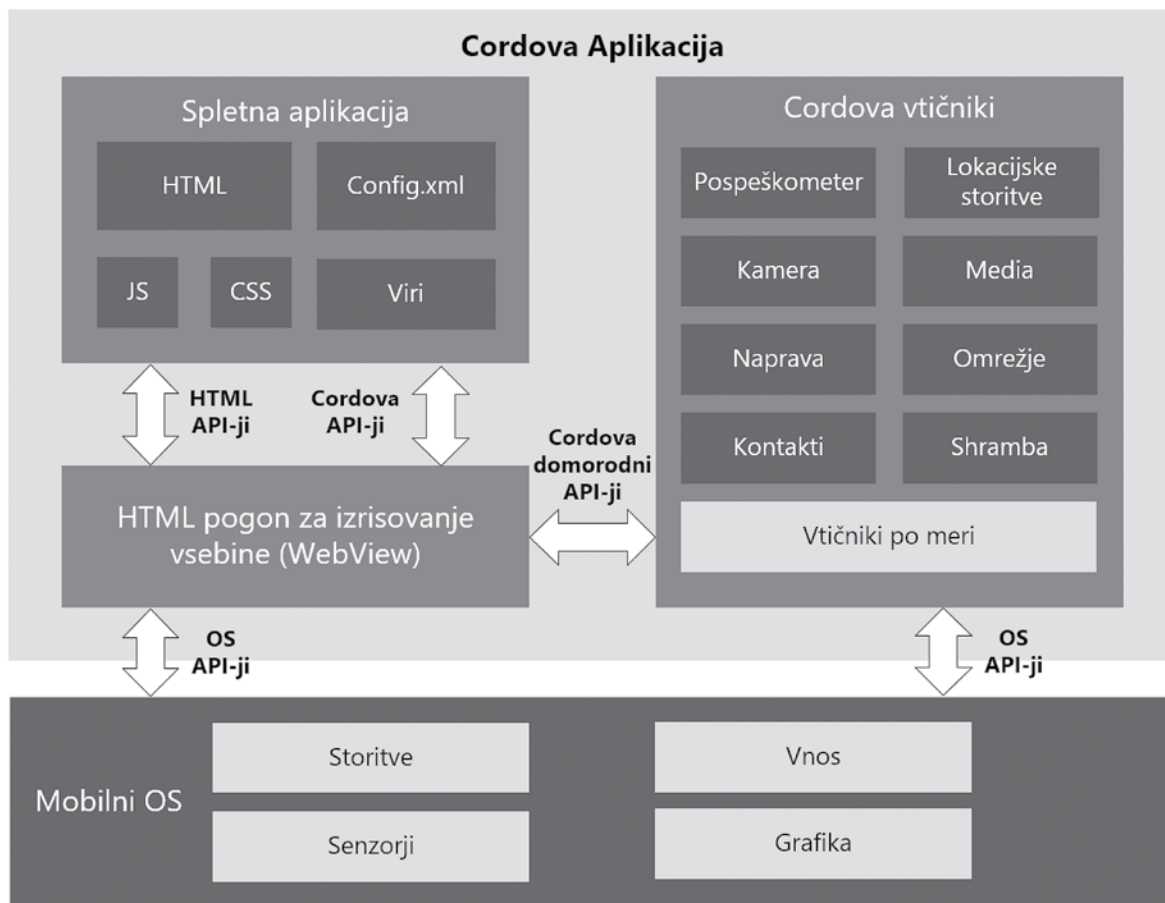
Čeprav je v operacijskem sistemu Windows mogoč razvoj aplikacij za vse podprte platforme, pa za poganjanje iOS aplikacij v vsakem primeru potrebujemo računalnik z nameščenim operacijskim sistemom macOS ter domorodnim razvojnim okoljem XCode, saj je za uspešno prevajanje aplikacije potreben iOS SDK (Xamarin, 2016b).

2.2 Predstavnik hibridnega pristopa Ionic

Ionic je knjižnica za razvoj hibridnih mobilnih aplikacij, ki temeljijo na ogrodjih, kot so Cordova, PhoneGap ali Trigger.io (Ionic, 2016). Vsa omogočajo razvoj večplatformskih aplikacij z uporabo standardnih spletnih tehnologij HTML5, CSS3 in JavaScript. Tako grajene aplikacije se izvajajo znotraj ovojnice, specifičnih za vsako ciljno platformo, in prek API-ja zagotavljajo dostop do zmožnosti mobilne naprave, kot so senzori, podatki ali omrežje. Arhitekturno zasnovo hibridne mobilne aplikacije v primeru uporabe hibridnega ogrodja Cordova predstavlja slika 2.

Tabela 3: **Podprtost platform glede na operacijski sistem in razvojno orodje**

Mobilna platforma	macOS		Windows	
	Xamarin Studio	Visual Studio	Xamarin Studio	Visual Studio
iOS	Da	Da (z macOS računalnikom)	Ne	Ne
Android	Da	Da	Da	Da
Windows Phone	Ne	Da	Da	Da



Slika 2: Arhitektura mobilne aplikacije na osnovi ogrodja Cordova (Cordova, 2012)

V primeru ogrodja Cordova celoten uporabniški vmesnik mobilne aplikacije predstavlja komponenta WebView, ki omogoča prikaz spletne vsebine znotraj aplikacije. Znotraj te se nahaja programska koda aplikacije, implementirana v obliki spletne strani skupaj z nastavitveno datoteko, ki vsebuje informacije o aplikaciji ter parametre za njeno delovanje. Ta nato prek API klicev komunicira z različnimi vtičniki, ki predstavljajo vmesnike tako za ogrodje Cordova kot za domorodne komponente mobilne naprav (Cordova, 2012).

Neodvisno od podpornega hibridnega ogrodja je glavni namen knjižnice Ionic uporabniku olajšati razvoj čela (angl. Front-end) mobilne aplikacije ter tako zagotoviti pristen izgled in občutek aplikacije predvsem na ravni interakcije z uporabniškim vmesnikom. To omogočata prilagojeno ogrodje CSS ter knjižnica JavaScript. Arhitektura ogrodja temelji na vzorcu View Controller, medtem ko za poslovno logiko skrbi odprtokodno ogrodje JavaScript Angu-

larJS (Ionic, 2016). Knjižnica v verziji 1.3 uradno podpira le razvoj aplikacij za platformi iOS in Android, a je zaradi uporabe spletnih tehnologij prenos aplikacije na platformo Windows Phone mogoč s preprosto modifikacijo nastavitvene datoteke (Hösl, 2014).

Zaradi uporabe spletnih tehnologij je aplikacijo Ionic mogoče razviti z uporabo katerega koli orodja, ki podpira razvoj spletnih aplikacij. Po drugi strani pa mobilna komponenta WebView zagotavlja, da je aplikacijo brez sprememb v programski kodi mogoče poganjati na vseh izbranih mobilnih platformah. Spremembe so potrebne le v primeru prilagajanja izgleda aplikacije pravilom posamezne platforme.

3 PRIMERJAVA PRISTOPOV

Glavni cilj raziskave je bil primerjava zmogljivosti delovanja večplatformsko razvitih mobilnih aplikacij z namenom ugotavljanja njihove primernosti kot alternative domorodnemu pristopu. V sklopu primerjave smo na vseh izbranih platformah opazovali

zmogljivost delovanja izbranih ogrodij Xamarin in Ionic v primerjavi z domorodnim pristopom, ki je podlaga za optimalno delovanje mobilnih aplikacij.

3.1 Opredelitev testov in testne aplikacije

Opredelitev testov je temeljila na standardnih testih mobilnih naprav (PassMark, 2016) ter vidikih odzivnosti aplikacije (DeCapua, 2015). Pri izvedbi testa smo se zgledovali po obstoječem delu Jakoba Danielssona (2014), ki je na podlagi mobilne platforme iOS primerjal različne pristope večplatformskega razvoja. Njegov pristop je temeljil na standardnih testih mobilnih naprav, ki so služili kot primerna podlaga za pripravo testov zmogljivosti naše raziskave. Naša analiza je od omenjene študije prevzela tri te-

ste, ki preverjajo zmogljivost delovanja procesorske enote, spomina in izrisovanja grafike, dodali pa smo primerjavo odzivnosti aplikacije ter časa, potrebnega za vzpostavitev aplikacije.

Na podlagi omenjenih dejstev smo zasnovali pet testov (tabela 4), pri katerih smo opazovali odzivni čas mobilne aplikacije, tj. metriko časovnega obnašanja (angl. Time behaviour metrics) zmogljivostne učinkovitosti, povzeto po standardu ISO/IEC 2010:2011. Metrika je namenjena merjenju povprečnega čakalnega časa, ki preteče od podane zahteve do pridobitve rezultata pod specifičnimi obremenitvami sistema v smislu sočasnih nalog ali sistemске uporabe (angl. Utilisation) (ISO, 2016).

Tabela 4: Testi zmogljivosti ter opazovana metrika

Test zmogljivosti	Razlaga	Scenarij izvedbe	Opazovana metrika
Test procesorja	Čas izvedba kalkulacije matematične konstante π na X decimalnih števil natančno z enovitim algoritmom (Code Codex, 2012)	X = 10.000	Odzivni čas (ms)
Test pomnilnika	Čas zapisovanja in branja zbirke X števil	X = 10.000.000	
Test 2D grafike	Čas, potreben za izris 1000 gumbov	X = 1000	
Test odzivnosti	Čas, ki preteče med interakcijo z aplikacijo in njenim odzivom v obliki preprostega obvestila	/	
Test vzpostavitev aplikacije	Čas, potreben za zagon aplikacije	/	

Za izvedbo testov zmogljivosti smo za vsako izmed izbranih mobilnih platform implementirali tri

mobilne aplikacije za vsakega izmed preučevanih razvojnih pristopov (slika 3).



Slika 3: Vzorčne aplikacije

Domorodni razvoj za vsako izmed platform je potekal s pripadajočim razvojnim orodjem in platformi specifičnem programskem jeziku. Aplikacijo iOS smo razvili z razvojnim orodjem XCode in programskim jezikom Objective-C, aplikacijo Android z orodjem Android Studio in programskim jezikom Java, aplikacijo Windows Phone pa z orodjem Visual Studio ter programskim jezikom C#. V okviru primerjave je razvoj za ogrodje Xamarin potekal nekoliko enostavneje, in sicer z enotnim programskim jezikom C#. Aplikacijo smo tako za iOS kot Android implementirali z orodjem Xamarin Studio, medtem ko smo za Windows Phone uporabili orodje Visual Studio. Najenostavnejše za razvoj se je izkazalo ogrodje Ionic, pri katerem smo po vzpostavitvi

projekta z uporabo ukazne vrstice vso programsko logiko za vse izmed platform zapisali v preprostem urejevalniku besedila. Glede na rezultirajoče aplikacije lahko opazimo, da aplikacije, razvite z domorodnim pristopom in ogrodjem Xamarin, posedujejo platformi domoroden izgled, medtem ko izgled aplikacije v primeru ogrodja Ionic ostaja enak skozi vse platforme.

3.2 Predstavitev testnih okolij

Za potrebe testov smo potrebovali tri izvajalna okolja (mobilne naprave) različnih platform, na katere smo namestili aplikacije vseh treh testiranih pristopov. Vse mobilne naprave ter njihovi pripadajoči operacijski sistemi so predstavljeni v tabeli 5.

Tabela 5: Testna izvajalna okolja

Atributi	iOS	Android	Windows Phone
Naprava	iPhone 6S	LG Nexus 5	Nokia Lumia 635
Procesor	Dual-core 1.84 GHz	Quad-core 2.3 GHz	Quad-core 1.2 GHz
Interni pomnilnik	2 GB	2 GB	512 MB
Grafična enota	PowerVR GT7600	Adreno 330	Adreno 305
Operacijski sistem	iOS 9.3.2	Android 6.0.1	Windows Phone 8.1

Enakovredno testiranje mobilnih aplikacij na različnih platformah je težko doseči predvsem zaradi razlik v strojni opremi. Enako velja za posnemovalnike mobilnih naprav, saj vsak izmed ponudnikov platforme določa svoj posnemovalnik, prilagojen specifični platformi. Ne glede na definirane razlike nam je uporaba dejanskih mobilnih naprav omogočila realno primerjavo rezultatov znotraj posamezne platforme.

3.3 Omejitve izvedbe testov

Primerjava različnih pristopov na različnih mobilnih napravah zahteva zapis mobilne aplikacije v več različnih programskih jezikih, ki imajo svoja sintaktična pravila. Če razvojna metoda zahteva programsko kodo, ki je specifična za določeno platformo, pride še do dodatne specializacije razvoja, kar lahko vpliva na rezultate analize. Na natančnost testiranja aplikacije lahko vpliva tudi metoda merjenja časa izvedbe določenega testa, saj je v nekaterih primerih (asinhroni klici) nemogoče programsko določiti konec testa. Dodatna omejitve so tudi same mobilne naprave ali njihovi posnemovalniki, saj je v prime-

ru uporabe katere koli možnosti testiranja aplikacije težko zagotoviti enakovrednost delovanja. Glede na našete pomanjkljivosti smo opredelili omejitve, ki lahko vplivajo na rezultate izvedenih testov.

- Različni programski jeziki. Pri razvoju za različne platforme se ne moremo izogniti različnim programskim jezikom, ki zaradi različne sintakse in tehnologije delovanja vplivajo na hitrost izvajanja procesov.
- Platformsko specifična koda. Različne tehnologije definirajo različne načine implementacije določenih platformsko specifičnih funkcionalnosti, česar v primeru različnih pristopov večplatformskega razvoja ne moremo poenotiti.
- Merjenje časa izvedbe testov. Za določena opravila je določitev časa, potrebnega za končanje naloge, mogoče natančno določiti v programski kodi, medtem ko to v nekaterih primerih ni mogoče (npr. asinhrono delovanje). V teh primerih je meritev mogoča le z uporabo zunanje meritve.
- Omejitve glede mobilnih naprav. Uporaba dejanskih naprav pri testiranju mobilnih aplikacij povzroči razlike, ki temeljijo na razlikah v strojni

opremi naprave. Neposredna primerjava rezultatov različnih mobilnih naprav je tako nemogoča, saj lahko razlike v strojni opremi povzročijo precejšnje spremembe v času izvedbe testov.

Zaradi navedenih omejitev smo v okviru razvitih mobilnih aplikacij uporabili le najosnovnejše komponente ter se z izjemo naslavljanja vizualnih elementov poskušali izogniti naprednejšim funkcionalnostim naprave. Omejitve merjenja časa izvedbe testov smo odpravili tako, da smo vsak test izvedli desetkrat ter nato spremljali povprečje časov izvedbe te-

stov. Ker nismo mogli zagotoviti enakovrednosti mobilnih naprav, smo podrobno analizo testov izvedli s primerjavo koeficientov sprememb povprečnega časa izvedbe testov med posameznimi pristopi.

3.4 Rezultati raziskave

Tabela 6 prikazuje rezultate testov, izvedenih na vseh izbranih mobilnih platformah ter pripadajočih mobilnih napravah. Za vsak test so navedeni povprečni čas izvedbe, maksimalni in minimalni čas ter mediana časa izvedbe.

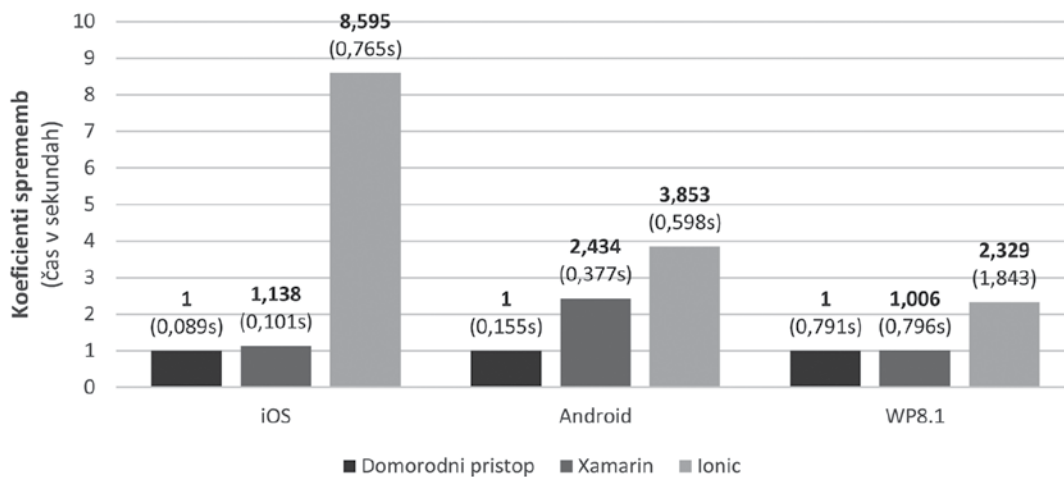
Tabela 6: **Rezultati testov zmogljivosti**

	Povprečni čas (s)	Maksimalni čas (s)	Minimalni čas (s)	Mediana (s)	Povprečni čas (s)	Maksimalni čas (s)	Minimalni čas (s)	Mediana (s)	Povprečni čas (s)	Maksimalni čas (s)	Minimalni čas (s)	Mediana (s)
Naprava	Apple iPhone 6S (iOS 9.3.2)				LG Nexus 5 (Android 6.0.1)				Nokia Lumia 635 (Windows Phone 8.1)			
Test	Test procesorja											
Dom.	0,089	0,118	0,053	0,095	0,155	0,178	0,145	0,148	0,791	0,797	0,785	0,793
Xamarin	0,101	0,112	0,096	0,098	0,377	0,393	0,361	0,381	0,796	0,807	0,783	0,797
Ionic	0,766	0,779	0,722	0,769	0,598	0,609	0,589	0,596	1,843	1,876	1,831	1,839
Test	Test spomina											
Dom.	0,147	0,157	0,140	0,146	0,208	0,232	0,186	0,202	0,701	0,720	0,681	0,700
Xamarin	0,166	0,171	0,152	0,167	0,302	0,325	0,289	0,301	0,713	0,740	0,695	0,709
Ionic	2,296	2,312	2,277	2,297	1,812	1,962	1,653	1,815	7,810	8,133	7,726	7,777
Test	Test 2D grafike											
Dom.	0,470	0,515	0,381	0,483	2,134	2,252	1,862	2,148	4,542	4,569	4,487	4,551
Xamarin	0,578	0,634	0,484	0,593	2,703	2,945	2,597	2,692	4,537	4,622	4,434	4,537
Ionic	3,081	3,229	3,004	3,069	23,97	27,47	22,71	23,01	100,8	106,9	94,49	99,88
Test	Test odzivnosti											
Dom.	0,042	0,056	0,029	0,042	0,072	0,086	0,055	0,071	0,087	0,097	0,071	0,087
Xamarin	0,074	0,078	0,068	0,073	0,088	0,096	0,081	0,088	0,092	0,102	0,083	0,092
Ionic	0,084	0,089	0,080	0,084	0,117	0,128	0,102	0,118	0,136	0,142	0,129	0,136
Test	Test vzpostavitve aplikacije											
Dom.	0,763	0,845	0,662	0,781	1,085	1,174	1,018	1,085	1,295	1,329	1,244	1,297
Xamarin	2,618	2,664	2,530	2,651	1,514	1,600	1,386	1,516	1,549	1,809	1,311	1,537
Ionic	0,782	0,897	0,725	0,765	2,264	2,390	2,181	2,250	2,415	2,476	2,294	2,421

Kot že omenjeno v razdelku 3.3, je enakovredna primerjava različnih mobilnih naprav otežena. Zaradi tega je nadaljnja analiza rezultatov potekala na podlagi koeficientov sprememb povprečnega časa, potrebnega za izvedbo posameznega testa, pri čemer smo za izhodiščno vrednost vzeli rezultat testa domorodne aplikacije.

3.4.1 Test procesorja

Pri testiranju mobilnih pristopov smo najprej opazovali čas, potreben za izračun konstante π na 10.000 mest natančno. Slika 4 prikazuje koeficiente sprememb povprečnega časa izvedbe ter dejanske vrednosti procesorskega testa na različnih mobilnih platformah.

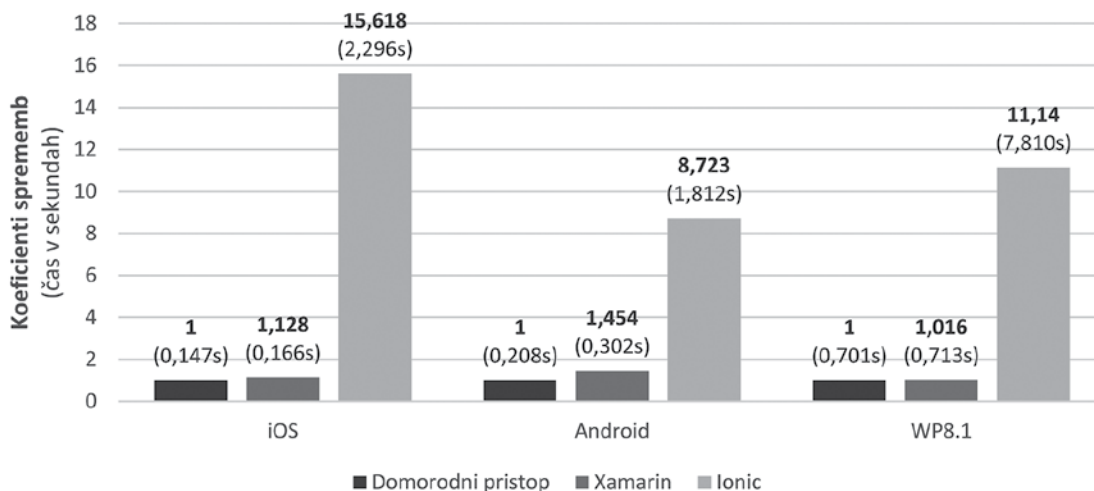


Slika 4: Koeficienti sprememb (test procesorja)

Iz slike je razvidno, da je v vseh primerih najkrajši čas za izvedbo testa potrebovala domorodna aplikacija, kateri je z manjšim odstopanjem sledilo ogrodje Xamarin (največje odstopanje v primeru platforme Android). V vseh primerih je največ časa potrebovala aplikacija, razvita s hibridnim pristopom Ionic, ki je v primeru platforme iOS od domorodnega pristopa odstopala za največ (8,6-krat).

3.4.2 Test pomnilnika

Testiranje pomnilnika mobilne naprave je potekalo na podlagi opazovanja časa zapisovanja zbirke 10,000.000 celih števil ter kopiranja nastalega seznama v nov seznam. Slika 5 prikazuje rezultate testa pomnilnika med mobilnimi platformami.

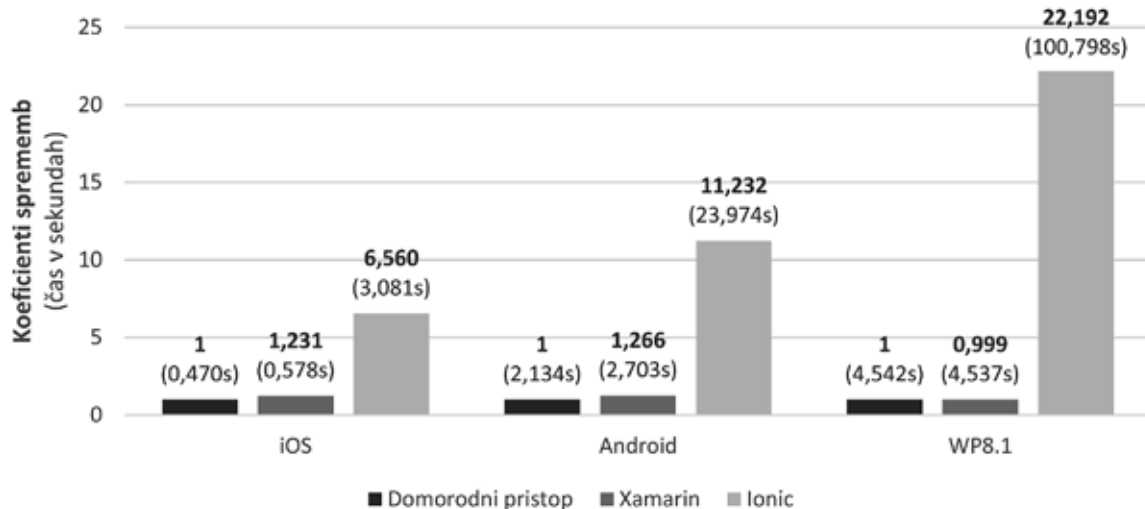


Slika 5: Koeficienti sprememb (test pomnilnika)

Rezultati testov pomnilnika so glede vrstnega reda pristopov podobni testu procesorja, a je v tem primeru prišlo do večjega odstopanja predvsem ogrodja Ionic, ki je odstopal za 8,7-krat v primeru platforme Android, za 11,1-krat v primeru platforme Windows Phone ter 15,6-krat v primeru platforme iOS. Razlog za večje odstopanje je mogoče pripisati tudi sami implementaciji zbirke na različnih platformah.

3.4.3 Test 2D grafike

Za testiranje izrisa vsebine smo definirali test, ki je na zaslon postopno izrisal 1.000 gumbov, pri tem pa smo merili čas, ki ga je aplikacija potrebovala, da je vse gumbe prikazala na zaslonu. Rezultati testa 2D grafike so predstavljeni na sliki 6.

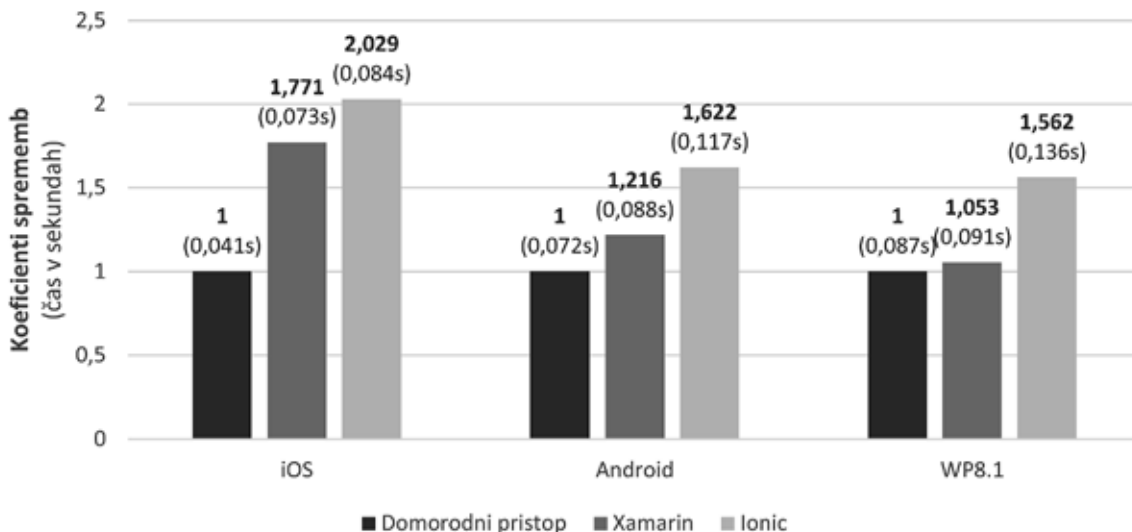


Slika 6: **Koeficienti sprememb (test 2D grafike)**

Izrisovanje vsebine na zaslon je grafično obremenjujoča operacija, kar je vplivalo predvsem na delovanje hibridne aplikacije. Na njeno zmogljivost na posamezni platformi vpliva tudi implementacija mobilne kontrole WebView, ki v primeru hibridnega pristopa skrbi za izrisovanje celotne vsebine mobilne aplikacije. Ionic se je tako v primeru vseh platform pri testu izkazal za najpočasnejšega. Najboljši rezultat je tudi v tem primeru dosegel domorodni pristop, kateremu je z manjšim odstopanjem sledil Xamarin.

3.4.4 Test odzivnosti

Delo z mobilno napravo poteka prek interakcije z različnimi vizualnimi kontrolami, ki se glede na uporabnikovo zahtevo odzovejo s spremembo vsebine. Raziskave so pokazale, da je za povprečnega uporabnika odziv instanten, če ta traja do 100 ms (Nielsen, 1993). Na podlagi tega smo definirali preprost test odziva mobilne aplikacije, pri katerem smo merili čas med interakcijo z aplikacijo in prikazom preprostega obvestila. Rezultati testa so prikazani na sliki 7.



Slika 7: Koefficienti sprememb (odzivni čas)

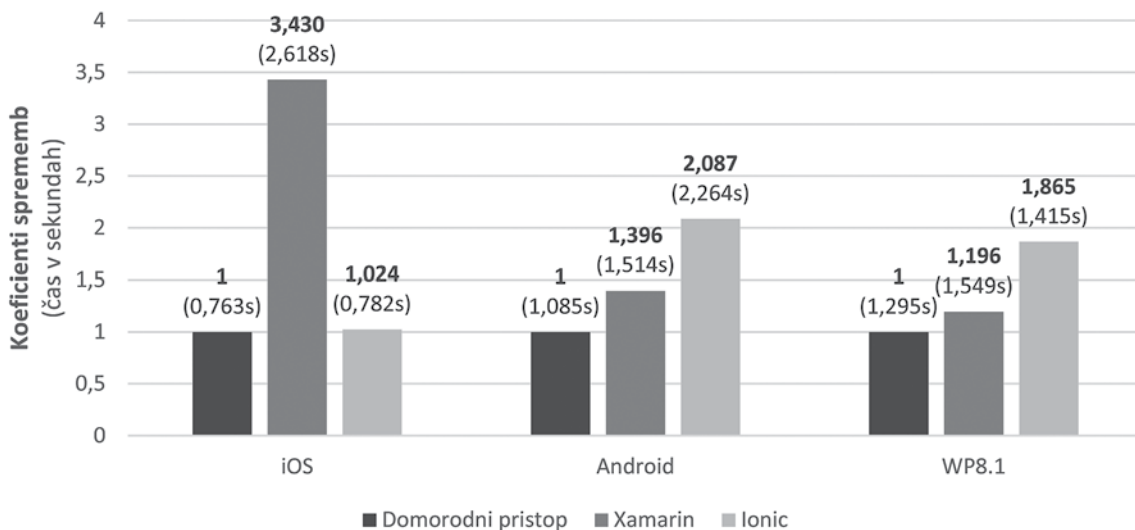
Rezultati testov vseh pristopov so pokazali, da je odzivni čas aplikacije na vseh platformah krajši od mejne vrednosti 100 ms ali nekaj nad to, saj je najdaljši povprečni čas znašal le 0,136 sekunde. Pri tem moramo upoštevati tudi čas prikaza obvestila, ki pogosto vključuje tudi kratko animacijo. V vseh primerih je znova najboljše rezultate dosegla domorodna aplikacija, kateri sta sledili ogrodji Xamarin in Ionic.

3.4.5 Test vzpostavitve aplikacije

Ne glede na kakovost mobilne aplikacije si uporabniki želijo, da je aplikacija na voljo v kar najkrajšem

možnem času. Slika 8 prikazuje koeficiente sprememb časa, ki ga aplikacija potrebuje za vzpostavitev in dosegljivost za uporabniško interakcijo.

Opazovanje časa zagona mobilne aplikacije je pokazalo, da največ časa v primeru platforme iOS potrebuje ogrodje Xamarin, in sicer 3,4-krat več v primerjavi z domorodnim pristopom, medtem ko je bil čas ogrođa Ionic ekvivalenten domorodnemu pristopu. V primeru platform Android in Windows Phone je največ časa za zagon potreboval Ionic in to za dvakrat več kot pri domorodnem pristopu v primeru platforme Android.



Slika 8: Koefficienti sprememb (čas zagona)

4 RAZPRAVA

Izvedba testov zmogljivosti razvitih aplikacij je bila podlaga za ocenjevanje primernosti različnih pristopov za razvoj zmogljivih ter zaradi tega učinkovitih mobilnih aplikacij za tri najbolj razširjene mobilne platforme. Pri vseh testih smo za izhodišče določili

domorodni pristop, s katerim smo na podlagi koeficientov sprememb primerjali ogrodji Xamarin in Ionic. V tabeli 7 predstavljamo združene rezultate koeficientov sprememb posameznih zmogljivostnih testov skupaj s povprečnimi izračuni vseh testov.

Tabela 7: **Združeni rezultati sprememb koeficientov**

	Večplatformsko prevedeni pristop (Xamarin)			Hibridni pristop (Ionic)		
	iOS	Android	WP	iOS	Android	WP
Test procesorja	1,138	2,434	1,006	8,595	3,853	2,329
Test pomnilnika	1,128	1,454	1,016	15,618	8,723	11,140
Test 2D grafike	1,231	1,266	0,999	6,560	11,232	22,192
Odzivni čas	1,771	1,216	1,053	2,029	1,622	1,562
Čas nalaganja	3,430	1,395	1,196	1,024	2,087	1,865
Povprečje	1,740	1,553	1,054	6,765	5,503	7,817

Glede na združene rezultate lahko opazimo velike razlike med preučevanima pristopoma. Ugotavljamo, da je med domorodnim pristopom in večplatformsko prevedenim pristopom le minimalna razlika, saj skupni povprečni koeficient sprememb vseh testov ne preseže vrednosti 1,74 v primeru platforme iOS, se nekoliko zmanjša v primeru platforme Android (1,55) in je skorajda enakovreden domorodnemu pristopu v primeru platforme Windows Phone (1,054). Večje razlike lahko opazimo v primerjavi s hibridnim pristopom, saj je najboljši rezultat dosegla platforma Android, vendar je kar 5,5-krat počasnejša v primerjavi z domorodnim pristopom. Sledi platforma iOS s koeficientom 6,76, medtem ko je najslabši rezultat dosegla platforma Windows Phone s koeficientom 7,8.

Na podlagi rezultatov lahko trdimo, da je z vidika zmogljivosti implementirane aplikacije večplatformsko prevedeni pristop primerna alternativa domorodnemu pristopu za dnevno uporabo pri razvoju mobilnih aplikacij. V primeru hibridnega pristopa pa lahko opazimo, da je šibka točka aplikacije predvsem zmogljivostno zahtevna poslovna logika. Zaradi tega je pristop primeren predvsem za mobilne aplikacije, ki ne vsebujejo veliko poslovne logike ali pa se ta nahaja na strojno močnejšem viru, do katerega nato dostopamo preko spletnih storitev.

5 SKLEP

Napredek mobilnih tehnologij prinaša vse več različnih pristopov k večplatformskemu mobilnemu razvoju, še več pa je ogrodij, ki jih implementirajo. Če je še pred časom domorodni pristop veljal za edinega, ki zagotavlja največjo učinkovitost delovanja aplikacije, se razlika v primerjavi z drugimi pristopi vztrajno zmanjšuje. To je potrdila tudi naša primerjalna analiza, saj se je pri merjenju zmogljivosti večplatformsko prevedeni pristop (ogrodje Xamarin) v primeru vseh mobilnih platform najbolj približal domorodnemu pristopu. Glede na rezultate ocenjujemo, da je v primeru ogrodja Xamarin večplatformsko prevedeni pristop primerna alternativa za domorodni razvojni pristop.

Nasprotno se je hibridni pristop izkazal za najmanj zmogljivega. Eden od razlogov za to je sama arhitektura pristopa, ki temelji na izvajanju aplikacije znotraj mobilne komponente WebView, katere zmogljivost je odvisna od zmogljivosti mobilnega brskalnika platforme. Pomanjkljivosti se nanašajo predvsem na strojno zahtevnejša opravila, kar pomeni, da je pristop lahko še vedno zanimiv za razvoj strojno nezahtevnih aplikacij ali takšnih, ki tovrstna opravila odlagajo na zmogljivejše vire.

Kljub razlikam v rezultatih je treba omeniti, da ti temeljijo predvsem na dveh specifičnih ogrodjih, kar pomeni, da lahko spremenjena izbira ogrodij vpliva na drugačne rezultate. Pri izbiri večplatformskih

ogrodij je treba prav tako upoštevati, da je zmogljivost le eden izmed faktorjev in da je treba pri odločitvi upoštevati tudi že obstoječo bazo znanja, kompleksnost grajenih aplikacij ter vse prednosti in slabosti ogrodja.

Rezultati primerjalne analize so namenjeni tako novim kot tudi že izkušenim razvijalcem mobilnih aplikacij, ki želijo na hiter in učinkovit način razviti zmogljive večplatformske mobilne aplikacije.

V prihodnje nameravamo raziskave na področju večplatformskega pristopa razširiti tako z vključevanjem večjega števila pristopov kot s primerjavo večjega števila različnih ogrodij znotraj enega samega pristopa. Na ravni vzorčne aplikacije nameravamo vključiti tudi teste 3D animacij ter obstoječe teste obogatiti z dodatnimi scenariji. Za večjo zanesljivost podatkov na posameznih platformah nameravamo vključiti še večje število naprav za vsako izmed testiranih platform.

6 LITERATURA

Dostopnost vseh elektronskih virov je bila potrjena dne 30. 7. 2016.

- [1] Adinugroho, T. Y. in Gautama, J. B. (2015). Review of Multi-platform Mobile Application Development Using WebView: Learning Management System on Mobile Platform. *Procedia Computer Science*, 59, 291–297. <http://doi.org/10.1016/j.procs.2015.07.568>.
- [2] Allen, J. (2011). The Death and Rebirth of Mono. <https://www.infoq.com/news/2011/05/Mono-II>.
- [3] Bouras, C., Papazois, A. in Stasinou, N. (2014). A framework for cross-platform mobile web applications using HTML5. V: *Proceedings - 2014 International Conference on Future Internet of Things and Cloud, FiCloud 2014* (str. 420–424). <http://doi.org/10.1109/FiCloud.2014.75>.
- [4] Charkaoui, S., Adraoui, Z. in Habib Benlahmar, E. (2014). Cross-platform mobile development approaches. *Information Science and Technology (CIST), 2014 Third IEEE International Colloquium in*, 188–191. <http://doi.org/10.1109/CIST.2014.7016616>.
- [5] Cisco. (2016). Cisco Visual Networking Index: Global Mobile Data Traffic Forecast Update, 2015–2020 White Paper. <http://www.cisco.com/c/en/us/solutions/collateral/service-provider/visual-networking-index-vni/mobile-white-paper-c11-520862.html>.
- [6] Cordova. (2012). Cordova Overview. <http://cordova.apache.org/docs/en/latest/guide/overview/index.html>.
- [7] Dalmasso, I., Datta, S. K., Bonnet, C. in Nikaein, N. (2013). Survey, comparison and evaluation of cross platform mobile application development tools. *2013 9th International Wireless Communications and Mobile Computing Conference (IWCMC)*, 323–328. <http://doi.org/10.1109/IWCMC.2013.6583580>.
- [8] Danielsson, J., Ameri, A., Friberg, M., Examiner, A. in Lindell, R. (2014). Comparison study of cross-platform developing tools for iPhone devices.
- [9] DeCapua, T. (2015). Front-end vs back-end performance metrics for mobile apps. <http://techbeacon.com/understanding-front-end-vs-back-end-performance-metrics-mobile-apps>.
- [10] Dhillon, S. (2015). An Evaluation Framework for Cross-Platform Mobile Application Development Tools. *Software: Practice and Experience*, 45(10), 1331–1357. https://atrium2.lib.uoguelph.ca/xmlui/bitstream/handle/10214/4949/Dhillon_Sanjeet_201212_MSc.pdf?sequence=3&isAllowed=y.
- [11] El-Kassas, W. S., Abdullah, B. A., Yousef, A. H. in Wahba, A. (2014). ICPMD: Integrated cross-platform mobile development solution. V: *Proceedings of 2014 9th IEEE International Conference on Computer Engineering and Systems, ICCES 2014* (str. 307–317). <http://doi.org/10.1109/ICCES.2014.7030977>.
- [12] Guthrie, S. (2016). Microsoft to acquire Xamarin and empower more developers to build apps on any device. <http://blogs.microsoft.com/blog/2016/02/24/microsoft-to-acquire-xamarin-and-empower-more-developers-to-build-apps-on-any-device/>.
- [13] Hösl, T. (2014). Using the Ionic Framework for Windows (Phone) 8.1 apps. <https://www.hoessl.eu/2014/12/on-using-the-ionic-framework-for-windows-phone-8-1-apps/>.
- [14] Ionic. (2016). Ionic Document Overview. <http://ionicframework.com/docs/overview/>.
- [15] ISO. (2016). ISO - International Organization for Standardization. <http://www.iso.org/iso/home.htm>.
- [16] Joorabchi, M. E., Mesbah, A. in Kruchten, P. (2013). Real challenges in mobile app development. V: *International Symposium on Empirical Software Engineering and Measurement* (str. 15–24). <http://doi.org/10.1109/ESEM.2013.9>.
- [17] Korf, M. in Oksman, E. (2016). Native, HTML5, or Hybrid: Understanding Your Mobile Application Development Options. https://developer.salesforce.com/page/Native,_HTML5,_or_Hybrid:_Understanding_Your_Mobile_Application_Development_Options.
- [18] Markov, D. (2015). Comparing The Top Frameworks For Building Hybrid Mobile Apps. <http://tutorialzine.com/2015/10/comparing-the-top-frameworks-for-building-hybrid-mobile-apps/>.
- [19] NetMarketShare. (2016). Mobile/Tablet Operating System Market Share. <https://www.netmarketshare.com/operating-system-market-share.aspx?qprid=10&qpcustomid=1>.
- [20] Nielsen, J. (1993). Response Times: The 3 Important Limits. <https://www.nngroup.com/articles/response-times-3-important-limits/>.
- [21] Noeticforce. (2016). 10 best hybrid mobile App UI Frameworks: HTML5, CSS and JS.
- [22] Palmieri, M., Singh, I. in Cicchetti, A. (2012). Comparison of cross-platform mobile development tools. V: *2012 16th International Conference on Intelligence in Next Generation Networks, ICIN 2012* (str. 179–186). <http://doi.org/10.1109/ICIN.2012.6376023>.
- [23] PassMark. (2016). PassMark PerformanceTest mobile - mobile device benchmark software. http://www.passmark.com/products/pt_mobile.htm.
- [24] Perchat, J., Desertot, M. in Lecomte, S. (2013). Component based framework to create mobile cross-platform applications. V: *Procedia Computer Science*, Vol. 19, str. 1004–1011. <http://doi.org/10.1016/j.procs.2013.06.140>.
- [25] R. M. de Andrade, P., B. Albuquerque, A., F. Frota, O., V. Silveira, R. in A. da Silva, F. (2015). Cross Platform App : A Comparative Study. *International Journal of Computer Science and Information Technology*, 7(1), 33–40. <http://doi.org/10.5121/ijcsit.2015.7104>.

- [26] Rahul Raj, C. P. in Tolety, S. B. (2012). A study on approaches to build cross-platform mobile applications and criteria to select appropriate approach. V: *2012 Annual IEEE India Conference, INDICON 2012* (str. 625–629). <http://doi.org/10.1109/INDICON.2012.6420693>.
- [27] Rajput, M. (2016). Pros and Cons of Developing Cross Platform Mobile Apps. <http://www.smallbizdaily.com/pros-cons-developing-cross-platform-mobile-apps/>.
- [28] Smutný, P. (2012). Mobile development tools and cross-platform solutions. V: *Proceedings of the 2012 13th International Carpathian Control Conference, ICC 2012* (str. 653–656). <http://doi.org/10.1109/CarpathianCC.2012.6228727>.
- [29] Statista. (2016a). Number of apps available in leading app stores as of June 2016. <http://www.statista.com/statistics/276623/number-of-apps-available-in-leading-app-stores/>.
- [30] Statista. (2016b). Number of mobile phone users worldwide from 2013 to 2019 (in billions). <http://www.statista.com/statistics/274774/forecast-of-mobile-phone-users-worldwide/>.
- [31] Titanium. (2016). Titanium pricing. <http://www.appcelerator.com/pricing/>.
- [32] Xamarin. (n. d.). Xamarin. <https://www.xamarin.com/>.
- [33] Xamarin. (2016a). Introduction to Mobile Development. https://developer.xamarin.com/guides/cross-platform/getting-started/introduction_to_mobile_development/.
- [34] Xamarin. (2016b). Understanding the Xamarin Mobile Platform. https://developer.xamarin.com/guides/cross-platform/application_fundamentals/building_cross_platform_applications/part_1_-_understanding_the_xamarin_mobile_platform/.
- [35] Xanthopoulos, S. in Xinogalos, S. (2013). A Comparative Analysis of Cross-platform Development Approaches for Mobile Applications. *Proceedings of the 6th Balkan Conference in Informatics*, 213–220. <http://doi.org/10.1145/2490257.2490292>.

Boris Ovcjak je asistent za področje informatike na Fakulteti za elektrotehniko, računalništvo in informatiko Univerze v Mariboru. Njegovo raziskovalno področje zajema področje mobilnih in spletnih tehnologij, računalništvo v oblaku in storitveno usmerjena arhitektura, vključno z vidikom sprejetosti omenjenih tehnologij. Sodeloval je na več projektih s področja razvoja informacijskih sistemov, prav tako pa je avtor in soavtor več strokovnih in znanstvenih prispevkov.

■

Gregor Polančič je docent na Fakulteti za elektrotehniko, računalništvo in informatiko Univerze v Mariboru. Med njegova raziskovalna področja spadajo tehnološki, metodološki in uporabniški vidiki sistemov za komuniciranje, sodelovanje in upravljanje poslovnih procesov, vključno z implikacijami sodobnih storitveno usmerjenih informacijskih rešitev na omenjena področja.

■

Tatjana Welzer Družovec je redna profesorica za področje informatike na Fakulteti za elektrotehniko, računalništvo in informatiko, Univerze v Mariboru. Je nosilka več predmetov s področja podatkovnih baz, varnosti in zasebnosti ter medkulturnega komuniciranja. Je vodja Laboratorija za podatkovne tehnologije - LPT. Njeno raziskovalno področje je predvsem povezano z modeliranjem podatkovnih baz in širše sistemov, kjer se upoštevajo tudi vplivi ekspertnih okolji, različnih kultur in ponovne uporabe že obstoječih rešitev. Tako kot drugi člani LPT je aktivna v številnih mednarodnih projektih. Rezultate raziskav je objavila v številnih znanstvenih in strokovnih prispevkih v domači in tuji literaturi.

Iz Islovarja

Islovar je spletni terminološki slovar informatike, ki ga ureja jezikovna sekcija Slovenskega društva Informatika. Islovar najdete na naslovu <http://www.islovar.org>. Tokrat objavljamo izbor novejših izrazov, ki smo jih vnesli v preteklem letu. Vabimo vas, da v Islovar prispevate svoje pripombe, predloge ali nove izraze.

Git -a m (angl. *Git*)

sistem za porazdeljeno dokumentiranje sprememb in izdajanje različic datotek; prim. CVS, RCS, SCCS, SVN

háptični -a -o prid. (angl. *haptic*)

ki sproža občutenje dotika fizičnega okolja s silami, vibracijami, premiki

héklab -a m (angl. *hackerspace, hacklab, makerspace, hackspace*)

javno dostopen delovni prostor skupnosti ljudi, navadno računalničarjev, digitalnih umetnikov, raziskovalcev, namenjen druženju, sodelovanju, izmenjevanju znanja, virov in ustvarjanju; sin. hekerplac; prim. heker

ígra RPG -e -- erpegé ž (angl. *RPG game, RPG, role-playing game, role playing game*)

računalniška igra, v kateri igralec prevzame eno od vlog v domišljjskem okolju; sin. igra igranja vlog; prim. množična večigralska internetna igra igranja vlog

izvózniški vtíčnik -ega -a m (angl. *exporter*)

program ali vtičnik za izvažanje

mnóžična večigrálska ígra RPG -e -e -e -- erpegé (angl. *massively multiplayer online game, MMO, MMOG*)

igra RPG, ki jo igra veliko igralcev; prim. množična večigralska internetna igra igranja vlog, večigralsko okolje

mnóžična večigrálska internétna ígra igránja vlóg

-e -e -e -e -- ž (angl. *massively multiplayer online role-playing game, MMORPG*)

računalniška igra igranja vlog v navideznem prostoru, igrana prek interneta, v kateri je udeleženo veliko število igralcev; prim. igra RPG, množična večigralska igra RPG, večigralsko okolje

naglávni komplét -ega -a m (angl. *headset*)

naprava, ki združuje naglavne slušalke in mikrofon

najáviti dov. (angl. *to declare*)

narediti, da postane kaj znano

napovédati dov. (angl. *to forecast*)

vnaprej povedati, določiti potek, pojavitev česa

napráva za mnóžično shranjevánje -e -- -- ž (angl. *mass-storage device, mass storage device*)

nosilec podatkov, ki omogoča trajno shranjevanje velike količine računalniških podatkov

ne slédi (angl. *do not track, DNT*)

del glave HTTP, ki spletnim stranem sporoča, ali uporabnik dovoljuje ali prepoveduje sledenje njegovim aktivnostim

nebotíčnik -a m (angl. *skyscraper*)

oglasna pasica navpične oblike, praviloma velikosti 120 x 600 pik; prim. široki nebotičnik

tvítati -m nedov. (angl. *to tweet, to twitter*)

objavljati sledilcem vidno kratko sporočilo na Twitterju

vèčigrálska ígra -e -e ž (*angl. multiplayer game, multiuser game*)
videoigra, ki jo igra hkrati več igralcev; sin.
večuporabniška igra; prim. enouporabniška igra

vedênsko usmêrjeni razvòj -- -ega razvòja m (*angl. behaviour driven development, BDD*)
razvoj programov, ki temelji na raziskavah o
vedenju uporabnikov

VHDL VHDL-ja krat. m (*angl. Very High Speed Integrated Circuit Hardware Description Language, VHDL*)
jezik za opis zelo hitrih digitalnih vezij

vôksel -sla m (*angl. voxel*)
element trirazsežne predstavitve

vtíčník -a m (*angl. plug-in, plugin*)
računalniški program, ki razširja funkcionalnost
osnovnega programa

zeló dólga ukázna beséda -- -e -e -e ž (*angl. very long instruction word, VLIW*)
arhitektura procesorja, ki omogoča paralelno
izvajanje opravil, tako da ukazno besedo razdeli
na osnovna opravila

Izbor pripravlja in ureja Katarina Puc s sodelavci

DNEVI SLOVENSKE INFORMATIKE 2017

Portorož, 11.–12. april 2017

› Zakaj obiskati konferenco DSI 2017?

V času vse hitrejših in korenitih sprememb, ko nič več ni tako, kot je bilo, ko smo vsi in vse povezani v digitalno življenje, je konferenca DSI 2017 pravo mesto za iskanje odgovorov in priložnosti. Skupaj lahko za kratek čas ustvarjamo prihodnost priložnosti za optimalno digitalizacijo poslovanja in ustvarjanje inovacij v sodelovanju širše strokovne javnosti iz zasebnega in javnega sektorja ter izobraževalnih in znanstvenih institucij. Pretvorimo izzive velikih in hitrih sprememb v priložnosti!

Spremembe so bile vodilo tudi pri oblikovanju programa letošnje konference, ki smo jo skrajšali na dva dni. K sodelovanju pri pripravi programa in organizaciji DSI 2017 smo pritegnili nove sodelavce s svežimi idejami. Na konferenco bomo povabili strokovne predavatelje, ki so inovativni, drzni in vizionarji na svojih področjih. Povabili bomo vodilne predstavnike podjetij, ki so uspešna na področju digitalizacije svojega poslovanja. Vabimo tudi vse vas, ki želite biti v digitalni preobrazbi uspešni in se zavedate pomena znanja in sodelovanja pri poslovanju. Naša vizija je tradicionalno konferenco slovenske informatike preobraziti v vodilni dogodek informacijske stroke, kjer se kalijo nove ideje in ustvarjajo nove povezave med udeleženci, saj skupaj lahko ustvarimo veliko več kot vsak na svojem vrtilčku.

› Komu je namenjena konferenca DSI 2017?

Vsem, ki sodelujete pri digitalni preobrazbi poslovanja in ste željni novih znanj ter stikov z ljudmi, ki imajo pred seboj podobne izzive. Na Dnevih slovenske informatike se bodo zbrali informatiki, ponudniki programske in strojne opreme, strokovnjaki in uporabniki informacijskih rešitev, letos že štiriindvajsetič zaporedoma v prijetnem okolju na slovenski obali.

Posebej pa ne zamudite konference tisti, ki ste odgovorni za digitalizacijo poslovanja v vašem podjetju!

› Kaj prinaša konferenca DSI 2017?

Rdeča nit letošnje konference Ustvarjamo prihodnost priložnosti predstavlja širši, bolj posloven pogled na informatiko, ki želi udeležencem predstaviti priložnosti, ki jih prinaša digitalna preobrazba poslovanja. Gre za spreminjanje ključnih elementov poslovanja, vključno s strategijo, poslovnim modelom, poslovnimi procesi, organizacijskimi strukturami in organizacijsko kulturo z izkoriščanjem zmožnosti sodobne digitalne tehnologije. Prav tako velja upoštevati digitalno preobrazbo celotne družbe.

Rdeča nit naj poveže gospodarstvo, javni sektor in izobraževalno-znanstvene ustanove pri izmenjavi idej in izkušenj. Posebna pozornost konference bo posvečena tudi študentom in nosilcem idej – zagonskim (»start-up«) podjetjem.

Skupaj bomo poskusili odgovoriti na vprašanje ali Slovenija lahko postane informacijski globalni inkubator?

Prepričani smo, da boste na konferenci strokovno in družabno uživali.

Vabljeni!



Pristopna izjava

za članstvo v Slovenskem društvu INFORMATIKA

Pravne osebe izpolnijo samo drugi del razpredelnice

Ime in priimek	
Datum rojstva	
Stopnja izobrazbe	srednja, višja, visoka
Naziv	prof., doc., spec., mag., dr.
Domači naslov	
Poštna št. in kraj	
Ulica in hišna številka	
Telefon (stacionarni/mobilni)	

Zaposlitev člana oz. člana - pravna oseba

Podjetje, organizacija	
Kontaktna oseba	
Davčna številka	
Poštna št. in kraj	
Ulica in hišna številka**	
Telefon	
Faks	
E-pošta	

Zanimajo me naslednja področja/sekcije*

- jezik
- informacijski sistemi
- operacijske raziskave
- seniorji
- zgodovina informatike
- poslovna informatika
- poslovne storitve
- informacijske storitve
- komunikacije in omrežja
- softver
- hardver
- upravna informatika
- geoinformatika
- izobraževanje

podpis

kraj, datum

Pošto društva želim prejemati na domači naslov / v službo.

Članarina znaša: 18,00 € - redna

7,20 € - za dodiplomske študente in seniorje (ob predložitvi dokazila o statusu)

120,00 € - za pravne osebe

Članarino, ki vključuje glasilo društva – revijo **Uporabna informatika**, bom poravnal sam / jo bo poravnal delodajalec.

DDV je vključen v članarino.



Naročilnica

 na revijo UPORABNA INFORMATIKA

Naročnina znaša: 35,00 € za fizične osebe

85,00 € za pravne osebe – prvi izvod

60,00 € za pravne osebe – vsak naslednji izvod

15,00 € za študente in seniorje (ob predložitvi dokazila o statusu)

DDV je vključen v naročnino.

ime in priimek ali naziv pravne osebe in ime kontaktne osebe

davčna številka, transakcijski račun

naslov plačnika

naslov, na katerega želite prejemati revijo (če je drugačen od naslova plačnika)

telefon/telefaks

elektronska pošta

Podpis

Datum

Znanstveni prispevki

Matjaž Roblek, Tomaž Kern

PRISTOP K PRENOVI IN INFORMACIJSKI PODPORI RAZVOJNEGA
PROCESA V SREDNJE VELIKEM PODJETJU

Mateja Grobelnik, Jurij Jaklič

ZNANJA IN SPOSOBNOSTI PODATKOVNIH ZNANSTVENIKOV:
PREGLED IN ANALIZA STANJA V SLOVENIJI

Damijan Kozina, Davorin Kofjač, Andrej Škraba

RAZVOJ MODELA SISTEMSKÉ DINAMIKE INVESTIRANJA IN DONOSNOSTI
KAPITALA V ORGANIZACIJI IN REALIZACIJA V OBLIKI SPLETNE APLIKACIJE

Strokovni prispevki

Boris Ovčjak, Tatjana Welzer Družovec, Gregor Polančič

PRIMERJAVA ZMOGLJIVOSTI VEČPLATFORMSKO RAZVITIH
MOBILNIH APLIKACIJ

Informacije

IZ ISLOVARJA

ISSN 1318-1882



9 771318 188001