

1

Osnove programskog inženjerstva. Životni ciklus razvoja programske podrške. Planiranje projekta

2020/21.01

Programska potpora

2

► Programska oprema/podrška/potpore, softver (engl. software)

- dio računalnog sustava koji nema fizikalnih dimenzija
- opći pojam za sve vrste programa, programskih jezika itd
- skup elemenata ili objekata u jedinstvenoj “konfiguraciji” koju čine računalni programi + podaci + dokumentacija
- svojstva:
 - lako se kopira (zajedno s pogreškama)
 - ne troši se, dugo se koristi, ali zastarijeva
 - složenost (problem održavanja, cijena nadogradnje)

► Primijenjena programska potpora = Računalna aplikacija (engl. application)

- namjenski program, primjenska programska oprema
- računalom podržano rješenje jednog ili više poslovnih problema ili potreba

► Informacijski sustav = sustav aplikacija za upravljanje ljudskim aktivnostima

Kako nastaje softver?

3

► Zanat ili inženjerstvo?

- Kome je namijenjen softver?
- Usporedba s proizvodnjom auta, izgradnjom građevina ...

- Tehnike, prakse, alati, predlošci ...
 - područje programskog inženjerstva

- Softver može nastati i bez korištenja tehnika i metoda programskog inženjerstva, ali takav softver je vjerojatno manje pouzdan, a lako moguće u konačnici i skuplji

Programsko inženjerstvo

4

► Programsко inženjerstvo (software engineering)

- “Software Engineering: The application of a systematic, disciplined, quantifiable approach to the development, operation, and maintenance of software; that is **the application of engineering to software**.
- ISO/IEC/IEEE 24765:2010 Systems and Software Engineering—Vocabulary
- **sistematičan, discipliniran i mjerljiv pristup razvoju, primjeni i održavanju softvera**
- **primjena inženjerskog pristupa na programsku opremu**
- Programsko inženjerstvo je inženjerska disciplina koja obuhvaća sve aspekte izrade programske opreme. [Sommerville, 2004]
- Područje programskog inženjerstva
 - poslovi kojima se oblikuje i razvija programska oprema
 - sustavna primjena prikladnih alata i tehnika na čitav proces razvoja programske potpore

Programsko inženjerstvo – „nulti pacijent”

5

- Pojam „programsko inženjerstvo” nastao 1963. godine
 - Margaret Hamilton (1936 -)
 - Serija NATO-vih konferencija od 1968.g.

Slika: Margaret Hamilton 1969. godine pored ispisa programskog koda za projekt Apollo.
Izvor: Wikimedia Commons



Programsko inženjerstvo i srodnna područja

6

“A scientist builds in order to learn; an engineer learns in order to build.” (Fred Brooks)

► Steve McConnell: „*Software Engineering is not Computer Science*”

Professional Software Development (originalni isječak na

https://www.gamasutra.com/view/feature/131817/software_engineering_is_not_.php

► [Sommerville, 2004]

► računarska znanost fokusira se na teorijske osnove

► programsko inženjerstvo orijentirano na praktičnu primjernu u razvoju i isporuci programske potpore

► Petter J. Denning

„Engineering has been marginalized by the unhealthy belief that engineering is the application of science” - Communications of the ACM, Vol. 60 No. 12, Pages 20-23, 2017.

Načela etike i profesionalnog rada u programskom inženjerstvu

7

1. JAVNI INTERES – Programski inženjeri djeluju u skladu s javnim interesom.
2. KLIJENT I POSLODAVAC – Programski inženjeri djeluju u skladu s interesima njihovog klijenta ili poslodavca, a koji nisu u suprotnosti s javnim interesom.
3. PROIZVOD – Programski inženjeri obvezuju se osigurati da njihovi proizvodi i prateće izmjene zadovoljavaju najviše moguće standarde struke.
4. PROSUDBA – Programski inženjeri održavaju integritet i neovisnost u svojim stručnim prosudbama
5. UPRAVLJANJE – Menedžeri i voditelji programskog inženjeringu potiču i promiču etički pristup upravljanju programskim razvojem i održavanjem.
6. STRUKA – Programski inženjeri unapređuju čestitost i ugled struke u skladu s javnim interesom.
7. KOLEGIJALNOST – Programski inženjeri međusobno se podržavaju i ophode pošteno.
8. ODNOS PREMA SEBI – Programski inženjeri se kontinuirano stručno unapređuju i promiču etički pristup radu u svojoj struci.

► Izvor: <https://www.open.hr/etika/nacela-etike-i-profesionalnog-rada-u-programskom-inzenjerstvu>

Poznavanje korisnikove domene

If your only tool is a hammer then every problem looks like a nail

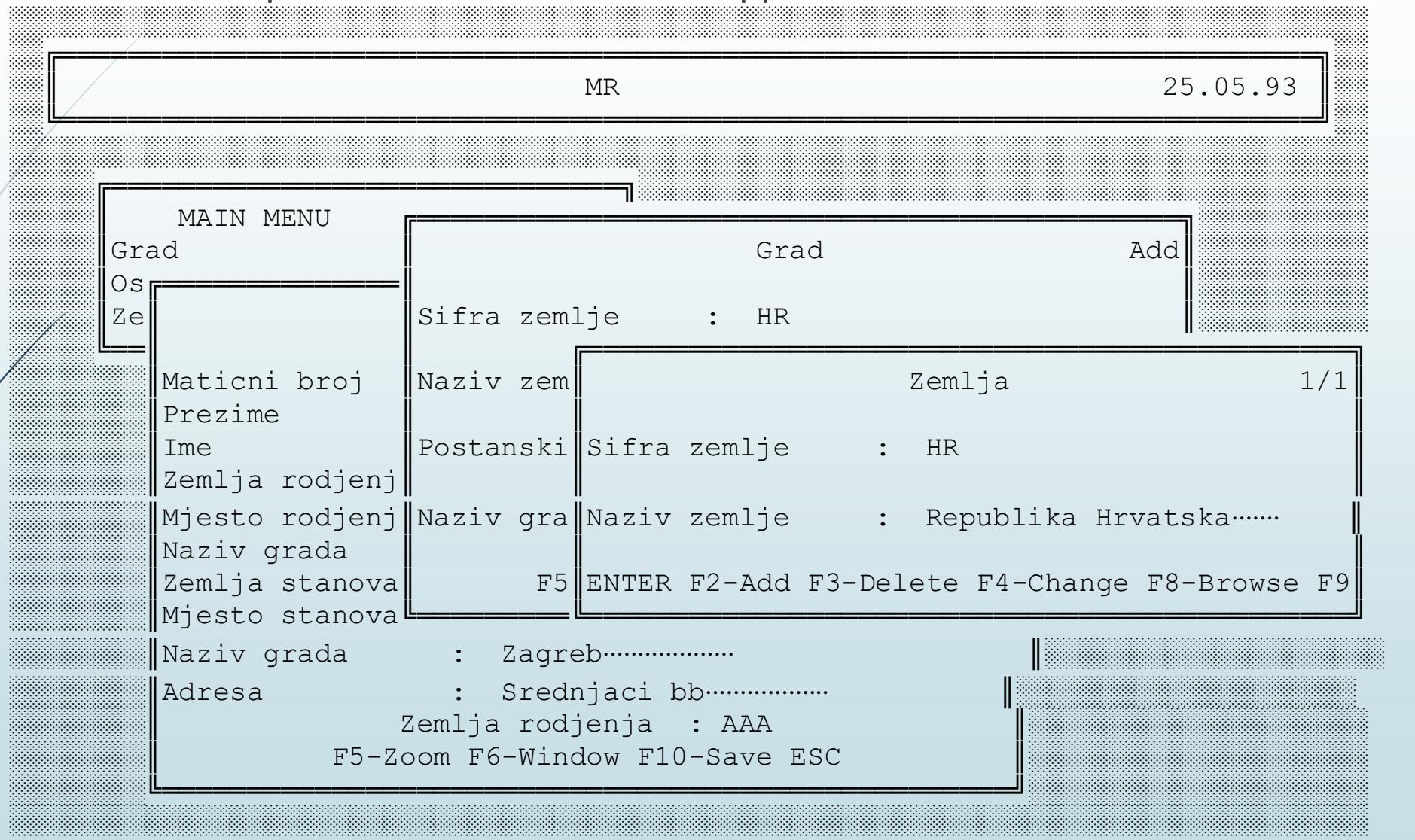
<https://quoteinvestigator.com/2014/05/08/hammer-nail/>

- ▶ Povećanje potražnje i činjenica da je relativno jednostavno pisati kod posljedično uzrokuje osmišljavanje boljih inženjerskih tehnika...
 - ▶ ... pri čemu treba znati kad i koju tehniku upotrijebiti
- ▶ Faktori koji utječu na odabir tehnologije i metodologije
 - ▶ veličina projekta, namjena, vijek trajanja, broj korisnika, oprema, ...
 - ▶ tanki klijent, debeli klijent, višeslojna aplikacija, web, mobilna, desktop aplikacija, izgled prilagođen različitim uređajima, ...
- ▶ Trendovi i tehnologije se mijenjaju, ali ne i činjenica da je potrebno ovladati jezikom (terminologijom) korisnika

Jednokorisničke, samostalne aplikacije

9

► „standalone”, početkom 90-ih, dBase, Clipper, ZIM, ...



Poslužiteljske aplikacije

10

► serverske, 90-ih, Informix, Oracle, ...

kfertalj (ansi) —| PIS - PERSONALNI INFORMACIJSKI SUSTAV —| (Sri) 04.12.96 |

OSOBA: Dohv Sljed Preth Unos Izmj Ostalo Lista Rasp Zap ...
Postavljanje uvjeta za dohvat zapisa

===== (LALIĆ MARIJAN) ===== [(2/3)] ===== 120/261 =====

Sprema (71) (Visoka VII/1
Br.svjed. () Dat. ()
Zanimanje (54131) (ORGANIZATORI S
Stamb.stanje (3) (Privremen korisni
Krvna grupa ()
Zdrav.broj ()
Ured za obranu (0) (*****
Osobni VPD (33207) (SIN-VSS I S
Početak voj.roka ()
Vojска voj.roka (0) (Nepoznata
Razl.prest.voj.roka ()
Sudjel. u dom.ratu (D)

Broj dokumenta _____ Orig. broj _____
Vrsta dokumenta BKA BANKOVNI TZVOD
Mj. troška/pri. 30 Pregled : radnik
Konto 99 radnik naziv
Iznos kuna _____
Dug/Pot D _____
Datum dokumenta 05
Analitika R
Program
Aktivnost
pror. glava
Izvor financir.
Pozicija
Opis
Valuta / devize _____
traz fm

Broj nalog 04 vs 0129
Datum knj. 09/11/2004

radnik	naziv			
90125	AGATIĆ	(-)	NENAD
90737	AGIC	(-)	DARKO
00953	AGLIĆ ALJINOVIC	(ANDRIJA)	ANDREA
00914	ALEKSIĆ-MASLAĆ	(KREŠIMIR)	KARMELA
90431	ALIĆ	(-)	MLADEN
90617	AMBRUŠ	(-)	DAVORIN
90731	ANDRASSY	(-)	MLADEN
90037	ANDREIĆ	(-)	VALENTIN
00899	ANDRES	(LADISLAV)	DALIBOR
01300	ANĐELINIC	(MARKO)	MATKO

Klijentske aplikacije

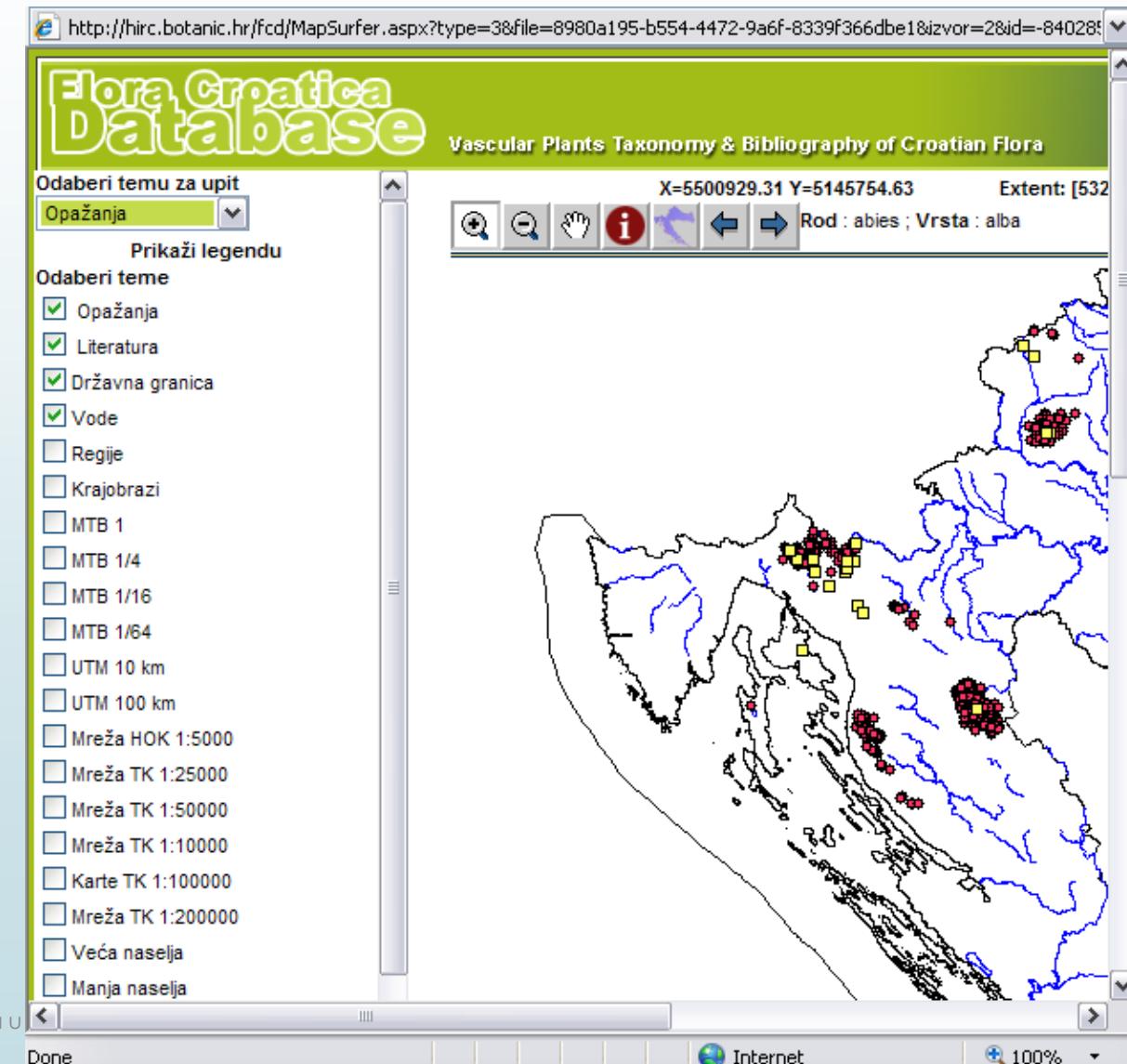
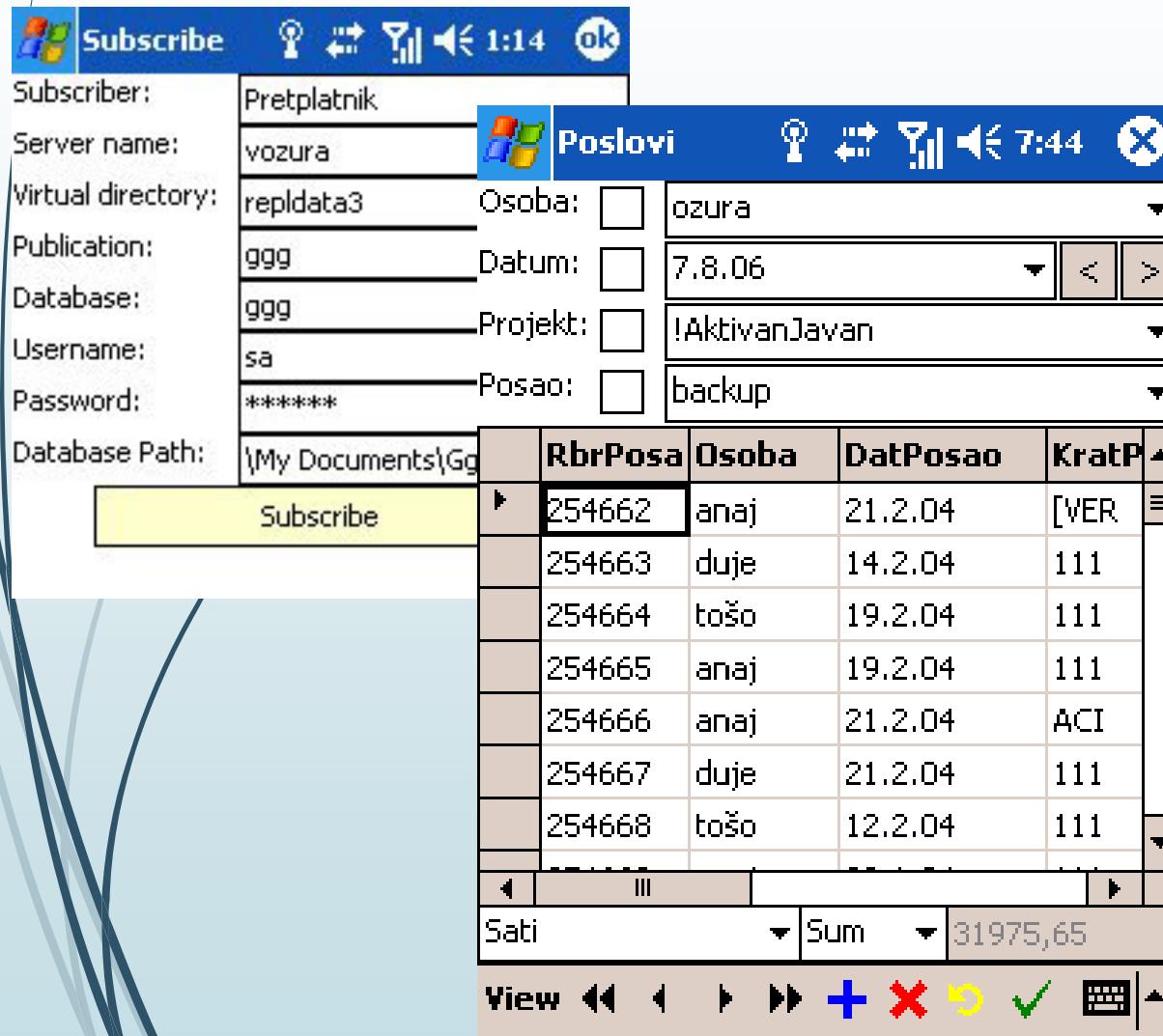
11

► “debeli” klijenti, kraj 90-ih, Microsoft Access, Visual Basic, Java, ...

Mobilne i distribuirane aplikacije (1)

12

► Internet, džepne, "tanki" klijenti, remoting, 2000-, .NET, J2EE



Mobilne i distribuirane aplikacije (2)

13

► Web servisi + mobilne, web aplikacije, jQuery

Operations at <http://hirc.botanic.hr/services/Herbar.svc>

Count	GET	Vraća ukupni broj podataka u herbaru koji zadovoljavaju traženi filter. Za postavke filtera pogledajte SearchCollection/{IdZbirke} .
Count/{IdZbirke}	GET	Vraća ukupni broj podataka u herbarskoj zbirci koji zadovoljavaju traženi filter. Za postavke filtera pogledajte SearchCollection/{IdZbirke} .
Sabiraci	GET	Vraća popis sabirača koje su zavedene u FCD-u koji počinju određenim nazivom (može se ispunjavati).
Search	GET	Vraća popis herbara koji zadovoljavaju određeni filter (može biti prazan). Opcionalno rezultat može sadržavati imesvojte, datumsabiranja, inventarnibroj, zbirka, idslike, tdwg, porodica, tip. Filter za search je godina, idherbara, idzbirke, inventarni broj (*), nagib opisnalažista (riječi odvojene razmakom), Toni/saslikom=true/opisnalažista=otok Vis/idzbirke=14/godina=20.
SearchCollection/{IdZbirke}	GET	Pretražuje herbarsku zbirku. Vidi Search za detalje.
TDWG	GET	Vraća popis država po TDWG-u. Rezultat je lista parova (oznaka države:ime).
TipoviPrimjeraka	GET	Vraća popis tipova primjeraka herbara. Rezultat je lista parova (oznaka tipa:ime).
Zbirke	GET	Vraća popis zbirki koje su zavedene u FCD-u. Rezultat je lista parova (oznaka zbirke:ime).

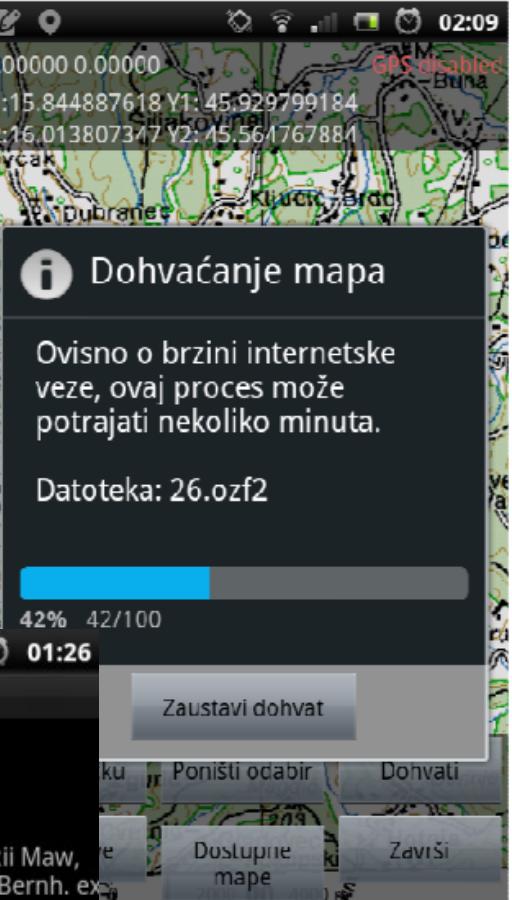
Opažanja Herbar Korisne poveznice Prikaži praznu kartu Kako koristiti ovu prijavu korisnika 1.3.2013. 12:22:32

Pretraga Rezultati pretrage

Kriterij pretrage: Godina sabiranja=2012, Naziv zbirke=CNHM Herbarium of Croatian Natural History Museum, TDWG=Croatia

1 2 Otisni herbarsku etiketu Zapisa po stranici 25 ▾ Ukupno rezultata: 42

Id herbara	Slika	Porodica	Ime svoje	Sabirač	God.	Tip	Naziv zbirke	Država (TDWG)	Ozn. koord.
<input type="checkbox"/> 31493		Rosaceae	Amelanchier ovalis Medik.	Vrbek, Mirjana	2012		CNHM Herbarium of Croatian Natural History Museum	Croatia	
<input type="checkbox"/> 31492		Boraginaceae	Lithospermum purpurocaeruleum L.	Vrbek, Mirjana	2012		CNHM Herbarium of Croatian Natural History Museum	Croatia	



Mbotanicar
Lilium bulbiferum L.
Liliaceae
Obj: Sp. Pl. 302 (1753)
S: Lilium aurantiacum Weston, Lilium chaixii Maw, Lilium croceum Chaix, Lilium pubescens Bernh. ex Hornem.
N: Bulbillentragende Feuerlilie, Giglio di San Giovanni, Giglio rosso, Lis a bulbillles, kruna, lukovičavi ljljan, narančasti krin, orange lily, turška lilija, zlatan, zlatoglav, zvjezdasti lijer, zvjezdasti ljljan, žilj
Status: VU, S3

Staništa:
C.3.5.3.4. - Travnjaci zmijka i pjegavog jastrebijaka, C.5.1.2.5. - Zajednica vlasnatog zmijka i planinske jeteline

Prilagodljivi dizajn

Kriterij pretrage: Vaskularna flora, Porodica: Liliaceae, Javno: Da
 Rezultati pretrage: Broj fotografija: 1036 , Broj svojti: 33 , Broj autora: 43

Broj zapisa po stranici

[1](#) [2](#) [3](#) [4](#) [»](#)

Danae racemosa (L.) Moench

 Id: 150022 Autor: Borovečki-Voska, Ljiljana Datum slike: 31.10.2018. Datum unosa: 21.11.2018.	 Id: 150023 Autor: Borovečki-Voska, Ljiljana Datum slike: 31.10.2018. Datum unosa: 21.11.2018.	 Id: 150021 Autor: Borovečki-Voska, Ljiljana Datum slike: 31.10.2018. Datum unosa: 21.11.2018.	 Id: 135868 Autor: Hrvatski prirodoslovni muzej, CNHM Datum slike: 4.8.2017. Datum unosa: 25.1.2018.
Tehnika: fotografija - digitalni aparat Objekt: habitus	Tehnika: fotografija - digitalni aparat Objekt: plod	Tehnika: fotografija - digitalni aparat Objekt: habitus	Tehnika: fotografija - digitalni aparat Objekt: habitus

Erythronium dens-canis L.

 Id: 22479 Autor: Prljić, Dragan Datum slike: 6.4.2011. Datum unosa: 6.4.2011.	 Id: 23128 Autor: Čičak, Marina Datum slike: Datum unosa: 23.4.2011.	 Id: 97252 Autor: Šarić, Šemso Datum slike: Datum unosa: 7.6.2016.	 Id: 97253 Autor: Šarić, Šemso Datum slike: Datum unosa: 7.6.2016.
Tehnika: makro snimak - digitalni aparat Objekt: plod	Tehnika: preuzeto iz publikacije Objekt: habitus	Tehnika: fotografija - digitalni aparat Objekt: ostalo	Tehnika: fotografija - digitalni aparat Objekt: ostalo
 Id: 97254 Autor: Šarić, Šemso Datum slike: Datum unosa: 7.6.2016.	 Id: 97255 Autor: Šarić, Šemso Datum slike: Datum unosa: 7.6.2016.	 Id: 97256 Autor: Šarić, Šemso Datum slike: Datum unosa: 7.6.2016.	 Id: 97257 Autor: Šarić, Šemso Datum slike: Datum unosa: 7.6.2016.
Tehnika: fotografija - digitalni aparat Objekt: ostalo	Tehnika: fotografija - digitalni aparat Objekt: ostalo	Tehnika: fotografija - digitalni aparat Objekt: ostalo	Tehnika: fotografija - digitalni aparat Objekt: ostalo
 Id: 138297	 Id: 138298	 Id: 139840	 Id: 154623

← → C ⌂ 🔍 visiani.botanic.hr/fcd... ⌂ ⌂



Kriterij pretrage: Vaskularna flora, Porodica: Liliaceae, Javno: Da
 Rezultati pretrage: Broj fotografija: 1036 , Broj svojti: 33 , Broj autora: 43
 Broj zapisa po stranici

[1](#) [2](#) [3](#) [4](#) [»](#)

Danae racemosa (L.) Moench


Id: 150022
Autor: Borovečki-Voska, Ljiljana
Datum slike: 31.10.2018.
Datum unosa: 21.11.2018.

Tehnika: fotografija - digitalni aparat
Objekt: habitus


Id: 150023
Autor: Borovečki-Voska, Ljiljana
Datum slike: 31.10.2018.
Datum unosa: 21.11.2018.

Tehnika: fotografija - digitalni aparat
Objekt: plod


Id: 150021
Autor: Borovečki-Voska, Ljiljana
Datum slike: 31.10.2018.
Datum unosa: 21.11.2018.

Debeli klijenti i višeslojne aplikacije

15

► Ovisno o namjeni i debeli klijenti i višeslojne aplikacije

IT produkt katalog

Produkt katalog SeCO CCA LRIC Izvještaji Baza

Indikatori proizvoda Tipovi naplate Indikatori Proizvod

Kategorije Klase imovine (vijek imovine) FAR Stavke Opexa Neaktivirana imovina IT produkt podaci detaljno (FAR, FTE, ...)

FTE CAPEX plan Neaktivirana imovina - Capex - Kategorije i proizvodi Mjesta troška

OPEX

Kalkulacija kataloga

Šiframnici Matični Dokumenti PIS - Prebac Demo Opcije Prozor Iznaj Izvještaji

VrstaShemeKnjiženja: Obracun PDV

Šifra sheme knjiženja 702000 Obracun PDV

Vrsta entiteta 702 Obracun PDV-a

Kontni plan 1 RRIF

Grupa pozicije 4 obračun PDV i PDV-K

Vr. poreznog dokumenta 14 Uzlazni račun, R1

Vr. plaćanja virmanom 0 Ne plaća se

Opis Obracun PDV

Ime ograničenu primjenu

Aktivan

Dovucite stupac kako bi ste grupirali podatke po tom stupcu

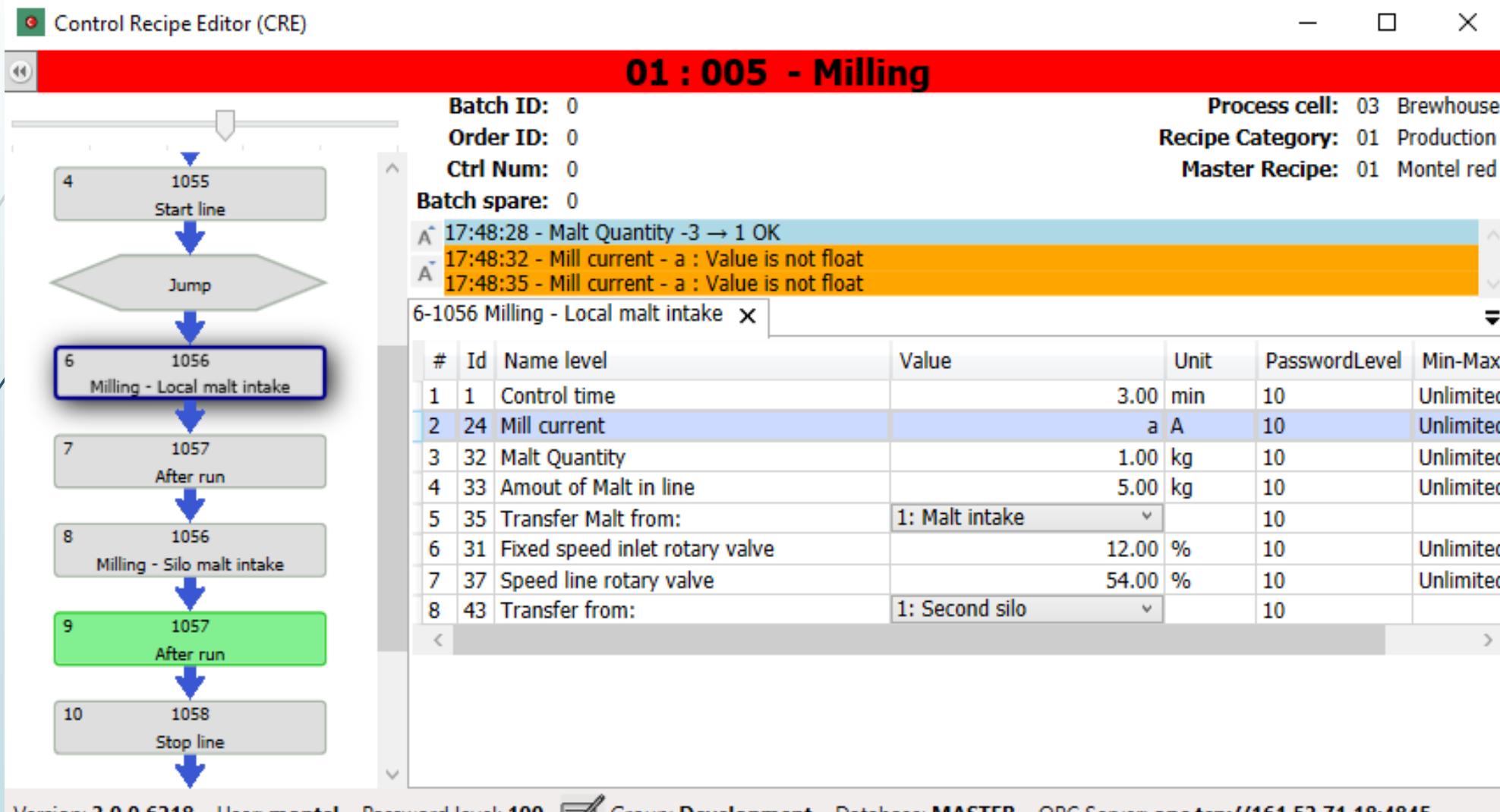
Rbr	Vrsta pozicije	Ds	Ps	P	K	Z	O	Algoritam	Kontrola	Rbrl	U
1	4194 Neupla	0	...	1...	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	=P4061+P4066+P4071+P407...	=	1	<input type="checkbox"/>
2	4192 Više u...	0	...	1...	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	=P4170+P4160+P4174+P418...	=	2	<input type="checkbox"/>
4	4004 Isporu...	0	...	1...	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	=	=	4	<input type="checkbox"/>
3	4000 Isporu...	0	...	1...	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	=	=	3	<input type="checkbox"/>
5	4002 Isporu...	0	...	1...	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	=	=	5	<input type="checkbox"/>
6	4006 Isporu...	0	...	1...	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	=	=	6	<input type="checkbox"/>
7	4010 R1 isp...	0	...	1...	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	=	=	7	<input type="checkbox"/>
9	4020 R2 isp...	0	...	1...	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	=	=	9	<input type="checkbox"/>
8	4015 Preduj...	0	...	1...	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	=	=	8	<input type="checkbox"/>
10	4025 Nezar...	0	...	1...	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	=	=	10	<input type="checkbox"/>

Prikazano zapisa: 228 / Ukupno zapisa: 228

Sustavi više međusobno povezanih aplikacija

16

- Različiti izvori podataka (OPC server, baza podataka, konfiguracijske datoteke, ...) + specifičan žargon domene



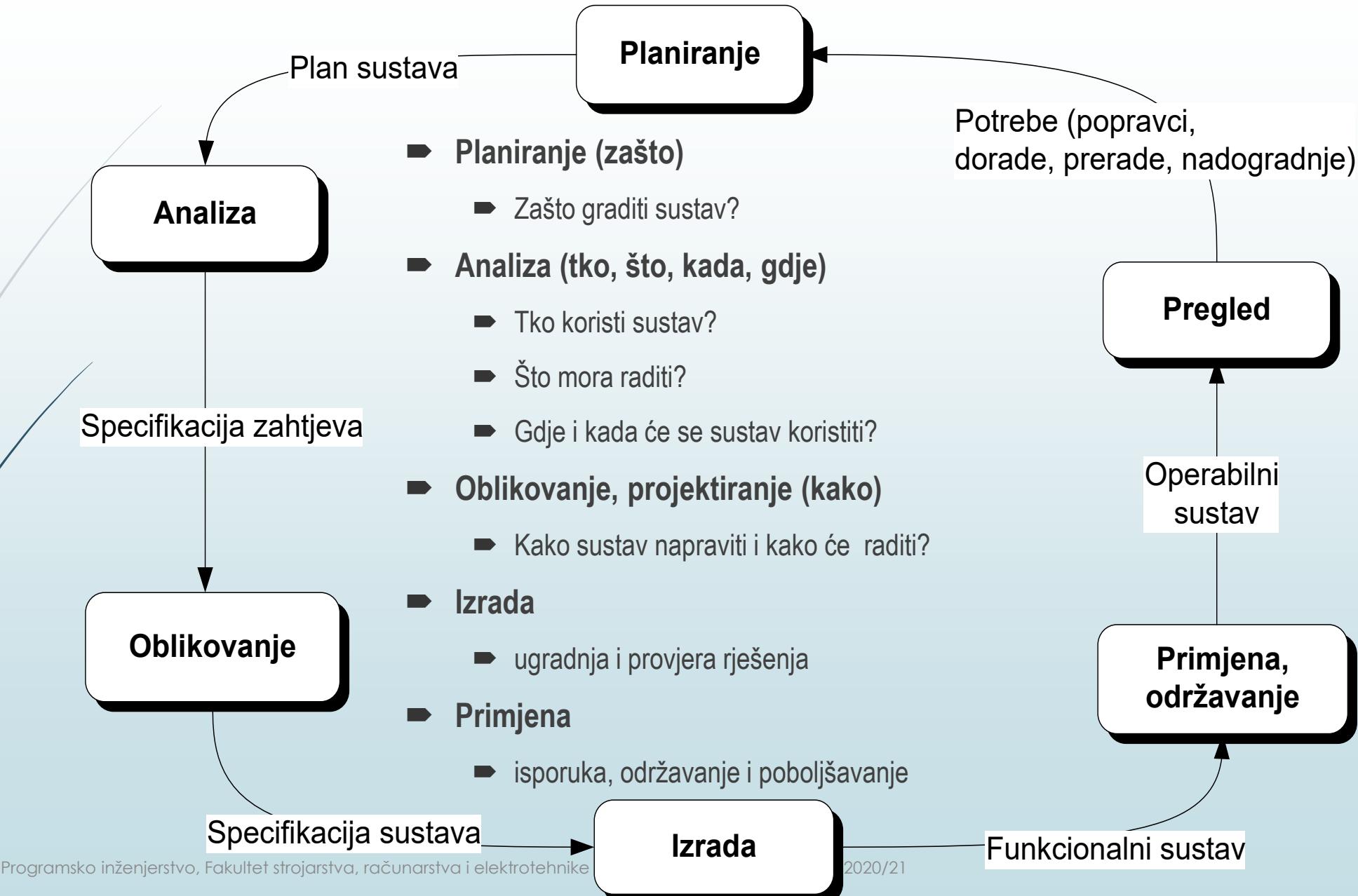
Fundamentalne aktivnosti razvoja softvera

17

- Izrada aplikacije nije samo kodiranje.
- Sommerville ističe 4 fundamentalne aktivnosti u svakom procesu razvoja softvera:

1. Specificiranje
 - Definiranje funkcionalnosti softvera i ograničenja pri radu
2. Dizajn i implementacija u skladu sa specifikacijama
3. Validacija
 - ispunjava li proizvod ono što je korisnik htio
 - Validacija (*Are we building the right product*) ≠ Verifikacija (*Are we building the product right*)
4. Evolucija
 - Izmjene u skladu s korisničkim potrebama.

Životni ciklus programske potpore



Faze životnog ciklusa

► Planiranje

- Utvrđivanje ciljeva (poslovne koristi)
- Analiza izvedivosti
- Izrada plana rada
- Ekipiranje projekta
- Upravljanje projektom

► Analiza

- Prikupljanje informacija
- Modeliranje procesa
- Modeliranje podataka
- Specifikacija zahtjeva

► Projektiranje, oblikovanje

- Dizajn arhitekture
- Dizajn baze podataka i datoteka
- Dizajn sučelja
- Dizajn programa

► Izrada, ugradnja (implementacija)

- Konstrukcija
- Testiranje
- Instalacija

► Primjena

- Rad
- Održavanje

► Projekt

- Projekt je vremenski određeno nastojanje da se proizvede jedinstven proizvod, usluga ili rezultat. [PMBOK – Project Management Book of Knowledge, PMI]
- Projekt je niz jedinstvenih, složenih i povezanih aktivnosti koje imaju određeni cilj i koji se mora postići u zadanom vremenskom roku, u okviru zadanog proračuna i u skladu sa specifikacijama. [Wisocky, Beck and Crane]

► Vremenska određenost

- Svaki projekt mora imati jasno određen početak i kraj. Projekti mogu biti kratki ili trajati godinama, ali će svakako završiti.
- Projekt završava u trenutku kada postane jasno da su ciljevi projekta dostignuti ili kada se zaključi da ciljevi projekta ne mogu ili neće biti dostignuti.

► Jedinstvenost

- Projekt se odnosi na rad na nečemu što prije nije postojalo i što se razlikuje od rezultata nastalih sličnim projektima.

Izrada plana projekta

- Koraci izrade plana projekta
 - Izrada liste zadataka
 - Izrada hijerarhije zadataka (work breakdown structure)
 - Procjena trajanja zadataka
 - Izrada ovisnosti među zadacima
 - Dodjela resursa

Zadatak	Trajanje (u danima)	Prethodnici	Naziv resursa
1 Doseg	7		
2 Određivanje dosega	5		Voditelj projekta
3 Određivanje sponzorstva	1,5	2	Sponzor projekta; Voditelj projekta
4 Određivanje resursa	22		Voditelj projekta
5 Dovršetak dosega	03;4		
6 Analiza/Softverski zahtjevi	16		
7 Analiza potreba	55		Sistem analitičar
8 Prikupljanje informacija	67		Sistem analitičar
9 Prijedlog izvedbe sustava	58		Sistem analitičar
10 Dovršetak analize	09		
11 Dizajn	5		
12 Razvoj funkcionalnih specifikacija	510		Sistem analitičar; Projektant
13 Izrada baze podataka	510		Administrator baze podataka
14 Dovršetak dizajna	013;12		
15 Razvoj	35		
16 Izrada formi korisničkog sučelja	412		Razvojnik
17 Izrada funkcija za pohranu podataka	1214		Razvojnik
18 Izrada funkcija za ispis izvještaja	1514		Razvojnik
19 Izrada izvještaja	1018		Razvojnik
20 Izrada funkcija izračuna	1216		Razvojnik
21 Razvojno testiranje(debugiranje)	1016;17;18;19;20		Razvojnik
22 Dovršetak razvoja	021		
23 Testiranje	12		
24 Izrada testova programske cjeline prema specifikacijama proizvoda	514		Tester
25 Izrada plana integracijskog testiranja prema specifikacijama proizvoda	514		Tester
26 Testiranje komponenti prema specifikacijama proizvoda	724		Tester
27 Provjera integracijemodula	725		Tester
28 Dovršetak testiranja	026;27		
29 Dokumentacija	5		
30 Razvoj specifikacija i sustava pomoći	522		Razvojnik
31 Dovršetak dokumentacije	030		Razvojnik
32 Uvođenje sustava i poduka korisnika	5		
33 Ugradnja programske potpore	528		Razvojnik; Sistem administrator
34 Dovršetak uvođenja sustava i poduke korisnika	033		

Izrada liste zadataka

22

► Zadaci – osnovni gradbeni elementi svakog projekta

- predstavljaju posao koji se mora obaviti da bi se postigao cilj projekta
- opisuju tijek događaja, trajanja i zahtjeva za resursima na projektu
- primitivni zadaci
 - zadaci koji se dekompozicijom ne mogu podijeliti na jednostavnije zadatke
- skupni zadaci (summary tasks)
 - zbrajaju trajanje i troškove primitivnih zadataka
 - trajanje, datum te izračunate vrijednosti se automatski izvode iz skupa primitivnih zadataka
- prekretnice ili miljokazi (milestones)
 - ključni događaj ili krajnji rok odnosno cilj koji treba postići
 - trajanja 0
 - služe za provjeru stupnja dovršenosti drugih zadataka
 - pomak ključnog događaja ima za posljedicu vremenski preraspored

Izrada hijerarhije zadataka

23

- Faza - grupa povezanih zadataka koji se odnose na fazu projekta
 - Zbirni zadaci se odnose na faze
- WBS (work breakdown structure)
 - hijerarhijska lista faza, zadataka i prekretnica
 - osnova za pregledni raspored projekta
- Dva su pristupa razvoju zadataka i faza:
 - Planiranje s vrha prema dolje (Top-down)
 - pristup od općeg prema specifičnom
 - identificira glavne faze i rezultate projekta prije dodavanja zadataka potrebnih za završetak tih faza
 - složeni projekti mogu imati nekoliko slojeva faza
 - Planiranje s dna prema gore (Bottom-up)
 - pristup od specifičnog prema općem
 - identificira što više zadataka najnižeg sloja prije grupiranja u faze

Procjena trajanja zadataka

24

► Trajanje zadatka

- očekivana količina vremena za završetak zadataka
 - minute (m), sati (h), dani (d), tjedni (w), mjeseci (mo)

Zadatak	Trajanje (u danima)	Prethodnici	Naziv resursa
1 Doseg	7		
2 Određivanje dosega	5		Voditelj projekta
3 Određivanje sponzorstva	1,5	2	Sponzor projekta;Voditelj projekta
4 Određivanje resursa	22		Voditelj projekta
5 Dovršetak dosega	03;4		
6 Analiza/Softverski zahtjevi	16		
7 Analiza potreba	55		Sistem analitičar
8 Prikupljanje informacija	67		Sistem analitičar
9 Prijedlog izvedbe sustava	58		Sistem analitičar
10 Dovršetak analize	09		
11 Dizajn	5		
12 Razvoj funkcionalnih specifikacija	510		Sistem analitičar;Projektant
13 Izrada baze podataka	510		Administrator baze podataka
14 Dovršetak dizajna	013;12		

Međuzavisnost zadataka

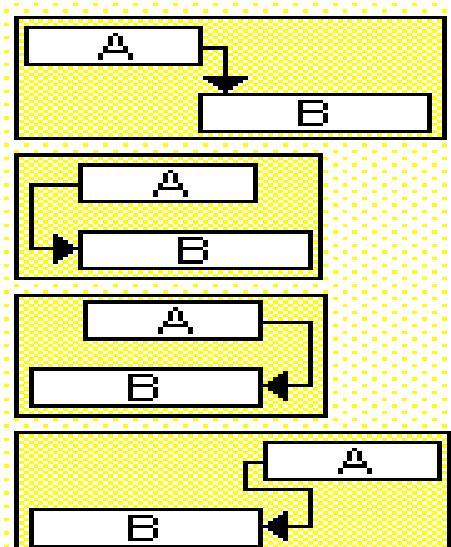
- Projekt može zahtijevati da zadaci budu napravljeni u određenom redoslijedu
 - Niz – iza jednog slijedi drugi zadatak
 - Zavisnost – sljedbenik (successor) može biti izvršen ako je dovršen prethodnik (predecessor)
 - Bilo koji zadatak može biti prethodnik jednom ili više sljedbenika

Finish-to-start (FS)

Start-to-start (SS)

Finish-to-finish (FF)

Start-to-finish (SF)



- Odnosi između zadataka:

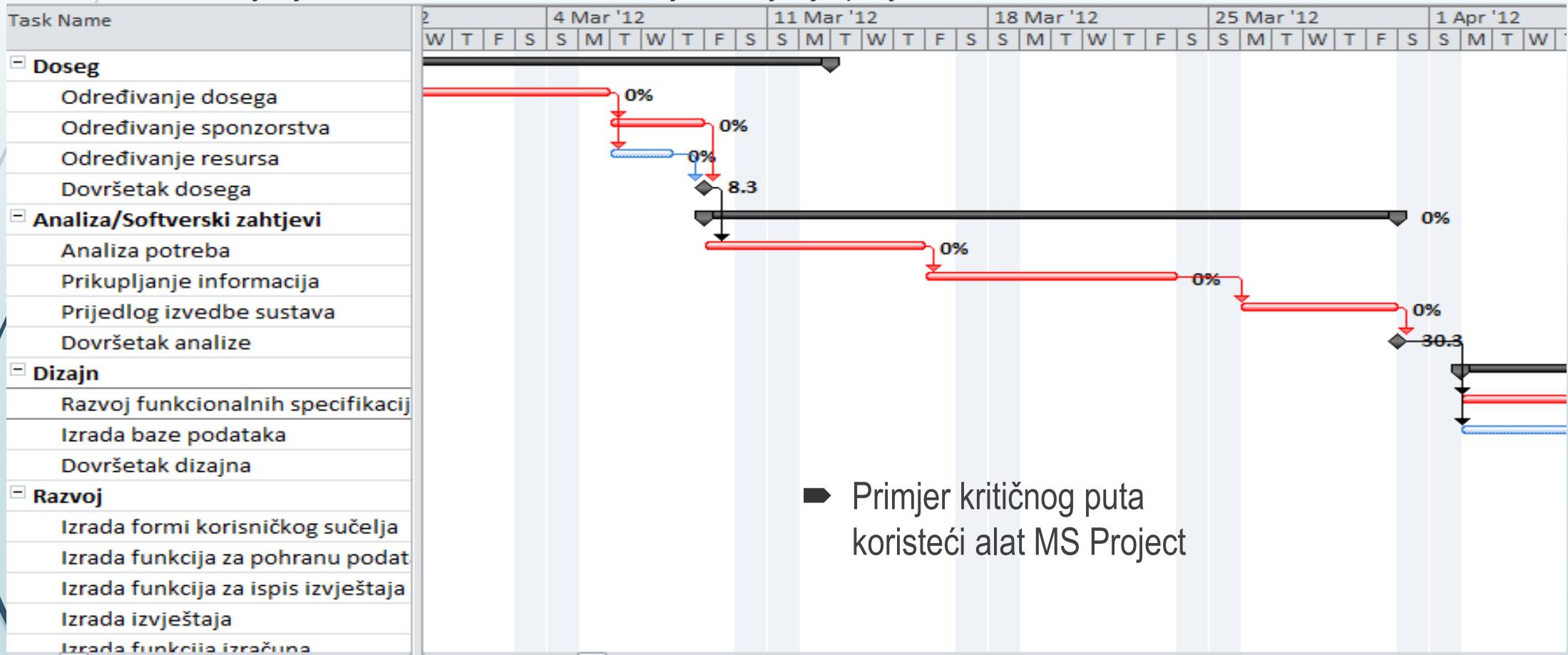
- Finish-to-start (FS) - završni datum prethodnika jest početni sljedbenika
- Start-to-start (SS) - početni datum prethodnika utvrđuje početni sljedbenika
- Finish-to-finish (FF) - završni datum prethodnika utvrđuje završni sljedbenika
- Start-to-finish (SF) - početni datum prethodnika utvrđuje završni sljedbenika

Kritični put

26

► Kritični put

- niz zadataka koji moraju završiti na vrijeme da bi projekt završio na vrijeme
- svaki zadatak na kritičnom putu je kritični zadatak
- kašnjenje kritičnih zadataka uzrokuje kašnjenje projekta



► Resursi - sredstva

- ljudi, oprema i materijal potrebni za obavljanje zadataka

► Vrste resursa:

► Resursi rada (work resources)

- ljudi (ograničeno vrijeme rada)

- oprema (neograničeno vrijeme rada)

► Resursi materijala (material resources)

- potrošni materijal koji predstavlja projektni utržak

- daje informaciju o brzini konzumiranja resursa

► Dva važna pogleda na resurse:

- Raspoloživost - u koje vrijeme određeni resurs može raditi na zadatku i koliko posla može obaviti
- Trošak – koliko novca će biti potrošeno na resurse

Raspodjela resursa

28

- Unos resursa i pratećih podataka (dostupnost, trošak)
- Maksimalne jedinice (max. units)
 - prikazuju vrijednosti raspoloživosti resursa u postocima
 - 100% predstavlja jednog čovjeka punog radnog vremena
 - 300% predstavlja tri čovjeka punog radnog vremena
- Osnova po kojoj je resursu dodijeljeno obavljanje posla
- Za pojedinačnu prilagodbu uvažavaju se radni i neradni dani resursa
 - Primjer: ako kalendar evidentira radno vrijeme samo četvrtkom i petkom 13-17 sati, 100% raspoloživosti nekog resursa ne znači 40 satno tjedno radno vrijeme, nego 8 sati rada tjedno

Dodjela resursa

29

- ▶ Dodjelom resursa zadatku, dodjeljuje mu se određeni posao
- ▶ Razlikovati posao (work) od trajanja (duration)
 - ▶ Posao je stvarni rad i odnosi se na
 - ▶ zadatak - stvarni rad potreban za završetak zadatka
 - ▶ dodijeljeni zadatak - stvarni rad nekog resursa na nekom zadatku
 - ▶ resurs - ukupni rad neke osobe na svim zadacima
 - ▶ Trajanje = Posao / Jedinice (Duration = Work / Units)
 - ▶ Ovisi o kalendaru rada
- ▶ Raspoređivanje temeljem napora (effort-driven scheduling)
 - ▶ metoda planiranja koja se koristi kod ažuriranja resursa zadatka
 - ▶ trajanje obrnuto količini resursa

► Rješavanje problema:

- ▶ Pomaknuti rokove (trajanje)
- ▶ Dodati dodatne resurse
- ▶ Produžiti radno vrijeme
- ▶ Povećati jedinice posla
- ▶ Smanjiti količinu posla

Dokumentiranje projekta

- ▶ Povelja projekta (Project Charter)
 - ▶ Dokument kojim pokretač projekta ili sponzor odobrava projekt i ovlašćuje voditelja za primjenu organizacijskih resursa u provedbi projekta.
- ▶ Plan projekta = Plan upravljanja softverskim projektom
 - ▶ IEEE Standard for Software Project Management Plans 1058-1998
 - ▶ dokument koji opisuje sveukupnu organizaciju projekta
- ▶ Primjeri  Prilozi\
 - ▶ PlanProjekta.dot
 - ▶ Firma-PlanProjekta.doc
- ▶ Plan može sadržavati raspored
 - ▶ općenito: SoftwareDevelopment.mpp,
PlanRazvoja.mpp
 - ▶ konkretno: Firma-PlanProjekta.mpp

1. Introduction

- 1.1 Project Overview
- 1.2 Project Deliverables
- 1.3 Evolution of the Software Project Management Plan
- 1.4 Reference Materials
- 1.5 Definitions and Acronyms

2. Project Organization

- 2.1 Process Model
- 2.2 Organizational Structure
- 2.3 Organizational Boundaries and Interfaces
- 2.4 Project Responsibilities

3. Managerial Process

- 3.1 Management Objectives and Priorities
- 3.2 Assumptions, Dependencies, and Constraints
- 3.3 Risk Management
- 3.4 Monitoring and Controlling Mechanisms
- 3.5 Staffing Plan

4. Technical Process

- 4.1 Methods, Tools, and Techniques
- 4.2 Software Documentation
- 4.3 Project Support Functions

5. Work Packages, Schedule, and Budget

- 5.1 Work Packages
- 5.2 Dependencies
- 5.3 Resource Requirements
- 5.4 Budget and Resource Allocation
- 5.5 Schedule

6. Additional Components

7. Index

8. Appendices

Upravljanje projektom

Upravljanje projektima je primjena znanja, vještina, alata i tehnika u projektnim aktivnostima da bi se ispunili projektni zahtjevi. [PMI]

- ▶ Planiranje
 - ▶ Utvrđivanje zahtjeva
 - ▶ Postavljanje jasnih i ostvarivih ciljeva
 - ▶ Uravnoteženje zahtjeva na kvalitetu, doseg, vrijeme i trošak,
 - ▶ Prilagodbu interesima i očekivanjima zainteresiranih strana – dionika (eng. Stakeholders)
- ▶ Organiziranje
 - ▶ Formiranje projektnog tima
- ▶ Raspoređivanje obaveza
 - ▶ Tko što i kada treba napraviti
- ▶ Usmjeravanje
 - ▶ Nadgledanje, omogućavanje izvršenja
- ▶ Kontroliranje
 - ▶ Provjera učinka i rezultata

Neke tipične uloge na projektu

33

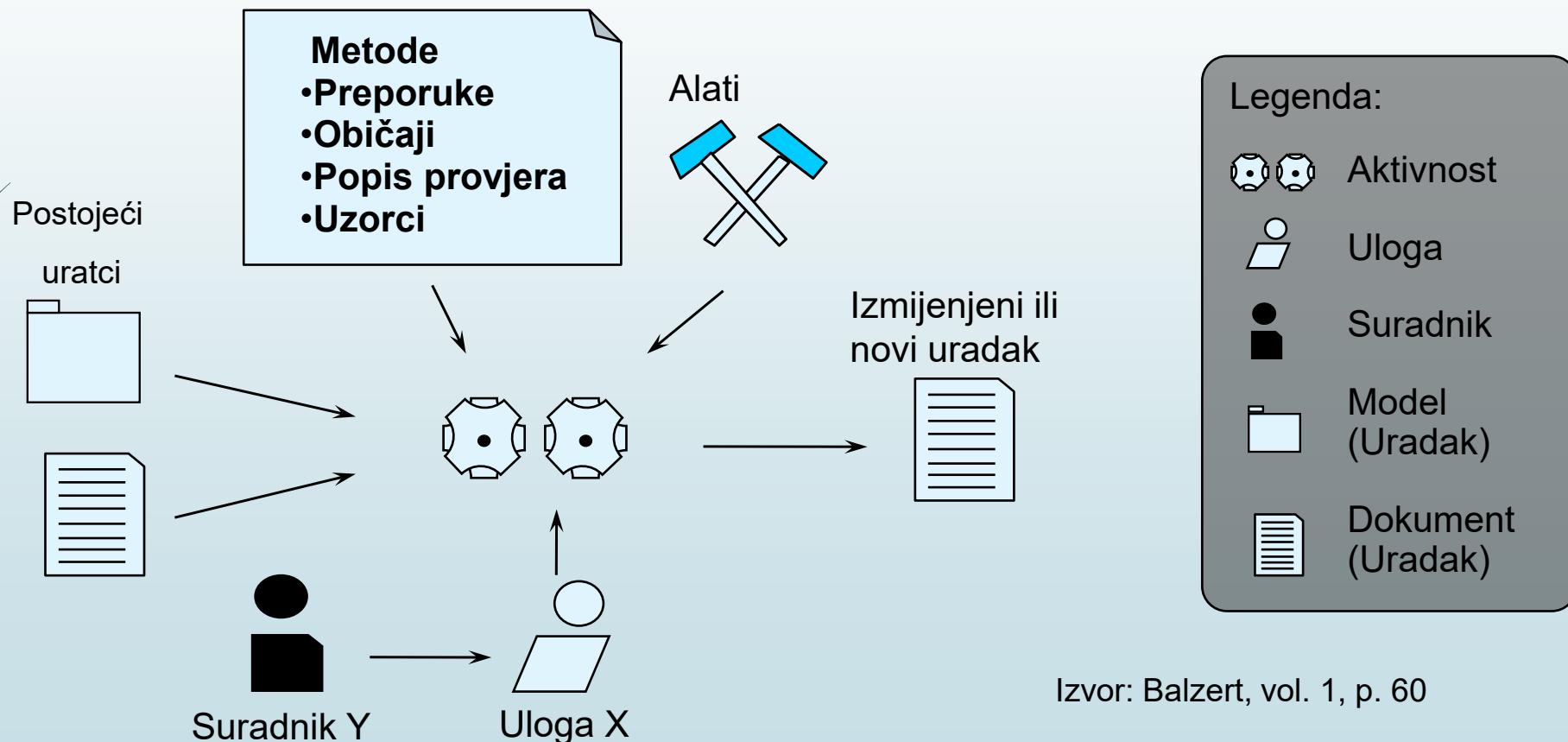
- ▶ Korisnik, Korisnik usluga, Klijent (User, Customer, Client)
 - ▶ osoba ili grupa, naručitelj ili krajnji korisnik
- ▶ Sponzor projekta (project sponsor)
 - ▶ Osoba ili grupa koja osigurava (financijske) resurse za projekt
- ▶ Voditelj projekta (project manager)
 - ▶ Osoba imenovana kako bi ostvarila ciljeve projekta
- ▶ Resursi projekta
 - ▶ Osobe, oprema, usluge, materijal, budžet ili druga sredstva.
- ▶ Projektna ekipa
 - ▶ Svi članovi ekipe, uključujući upravljačke, a u nekim slučajevima i sponzora
 - ▶ voditelj – upravljanje projektom
 - ▶ sistem analitičar – određivanje potreba, specifikacija zahtjeva i dizajna
 - ▶ projektant/arhitekt – uspostava osnovne arhitekture
 - ▶ razvojnik (developer, builder) – kodiranje, testiranje
 - ▶ administrator baza podataka – administriranje DBMS
 - ▶ sistem inženjer / sistem administrator – administriranje OS i mreže

Životni ciklus i modeli razvoja

Aktivnosti procesa

35

- Općenito: Plan razvoja, koji navodi opće postupke razvoja programskog proizvoda.
- Preciznije: Definicija koja kaže koje aktivnosti treba obaviti, tko ih treba obaviti i u kojoj ulozi; kojim redoslijedom, koji će proizvodi (artefakti) biti razvijeni i kako ih vrednovati.



Izvor: Balzert, vol. 1, p. 60

Zašto je potreban model procesa

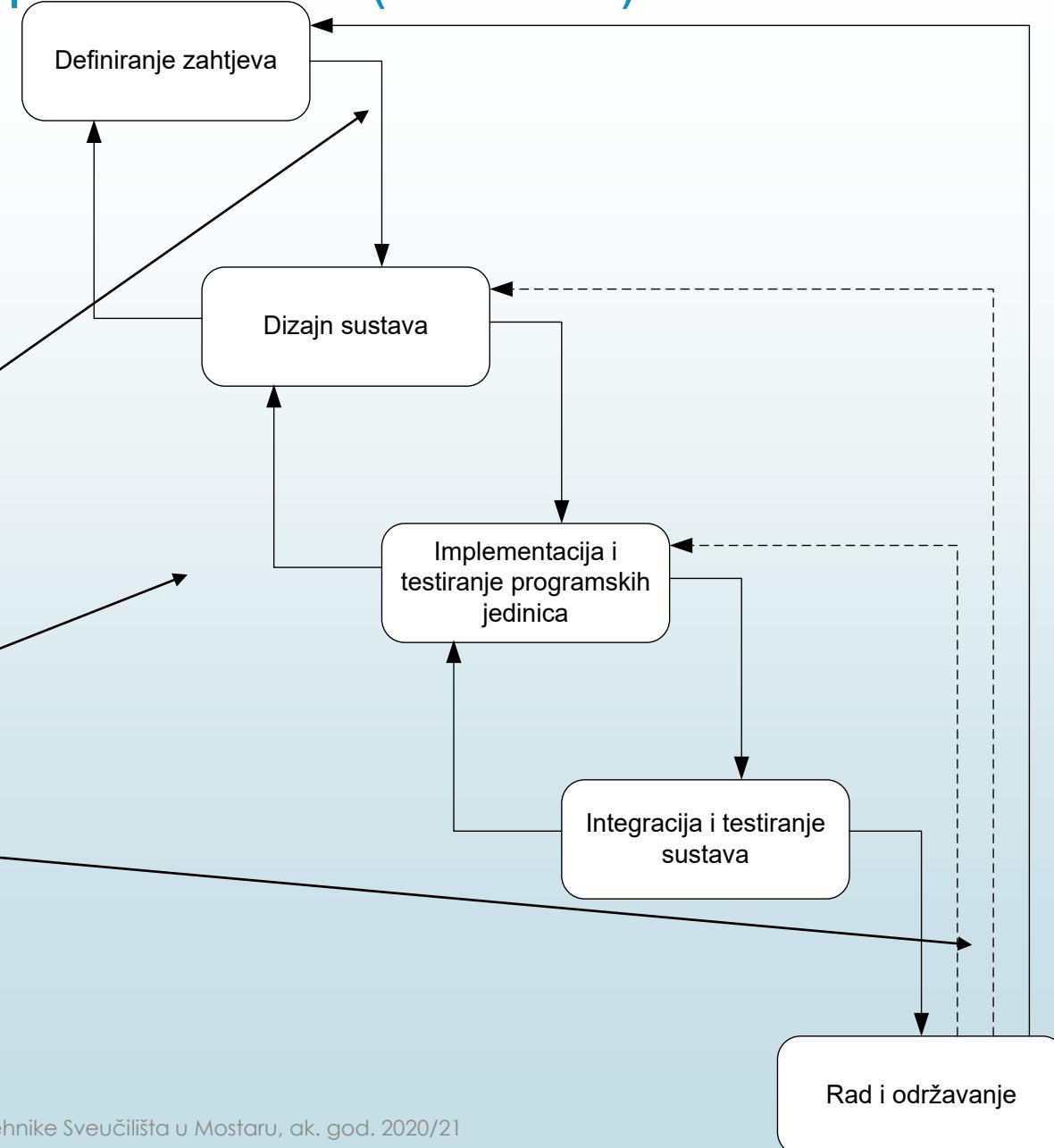
36

- ▶ Bitan za organizaciju projekta
 - ▶ U protivnom je upravljanje projektom sporadično i nekoordinirano
- ▶ Omogućava vremensko i finansijsko planiranje i standardizaciju rada
- ▶ Pruža „kontrolne točke“ za praćenje napretka, procjenu postignutih rezultata i donošenje odluka o dalnjim koracima
- ▶ Artefakti:
 - ▶ dokumenti, modeli, programi, ...
- ▶ Artefakti i ekipe određene pojedinim modelom, alatom i/ili metodologijom rada
- ▶ Model predstavlja apstrakciju životnog ciklusa razvoja softvera

Vodopadni model (waterfall)

► Najstariji model
(Royce 1970) i
slijedi
generički SDLC

- Varijante
 - klasični
 - pseudostruktturni
 - radikalni



Vodopadni model razvoja

- ▶ Definiranje zahtjeva (*requirements analysis and definition*)
 - ▶ definira funkcionalnost programske potpore prema zahtjevima korisnika.
- ▶ Dizajn sustava (*system and software design*)
 - ▶ definira cijelokupnu arhitekturu programske podrške
 - ▶ grubi model sustava razrađuje se u detaljni opis izvedbe
- ▶ Ugradnja i testiranje jedinica (*implementation and unit testing*)
 - ▶ faza kodiranja tijekom koje se zahtjevi prevode u programski kod
 - ▶ programske jedinice zasebno se testiraju provjerom naspram specifikacije
- ▶ Integracija i testiranje sustava (*integration and system testing*)
 - ▶ programske jedinice povezuju se u cjelinu
 - ▶ provjerava se odgovara li programska potpora zahtjevima korisnika
- ▶ Primjena i održavanje (*deployment/operation and maintenance*)
 - ▶ započinje predajom sustava korisnicima na uporabu
 - ▶ tijekom održavanja uklanjaju se naknadno uočene neispravnosti te se sustav proširuje i poboljšava prema potrebama

Iterativni i inkrementalni postupak razvoja

39

► Iterativni i inkrementalni razvoj

- softver se razvija i objavljuje po dijelovima
- glavne faze obavljaju se kroz niz iteracija
 - svaka iteracija obavlja se standardnim životnim ciklusom koji uključuje analizu, oblikovanje, ugradnju i provjeru
 - rezultat iteracije je proizvod završne kakvoće (production-quality), provjeren i integriran, koji zadovoljava podskup ukupnih zahtjeva
 - isporuke mogu biti interne ili prema korisnicima

► Primjer modela procesa: IBM Rational Unified Process (RUP)

- Niz "predložaka" razvojnih procesa s glavnim gradivnim blokovima:
uloga (tko), artefakt (što), aktivnost (kako)
- Derivati: Open UP, Agile UP

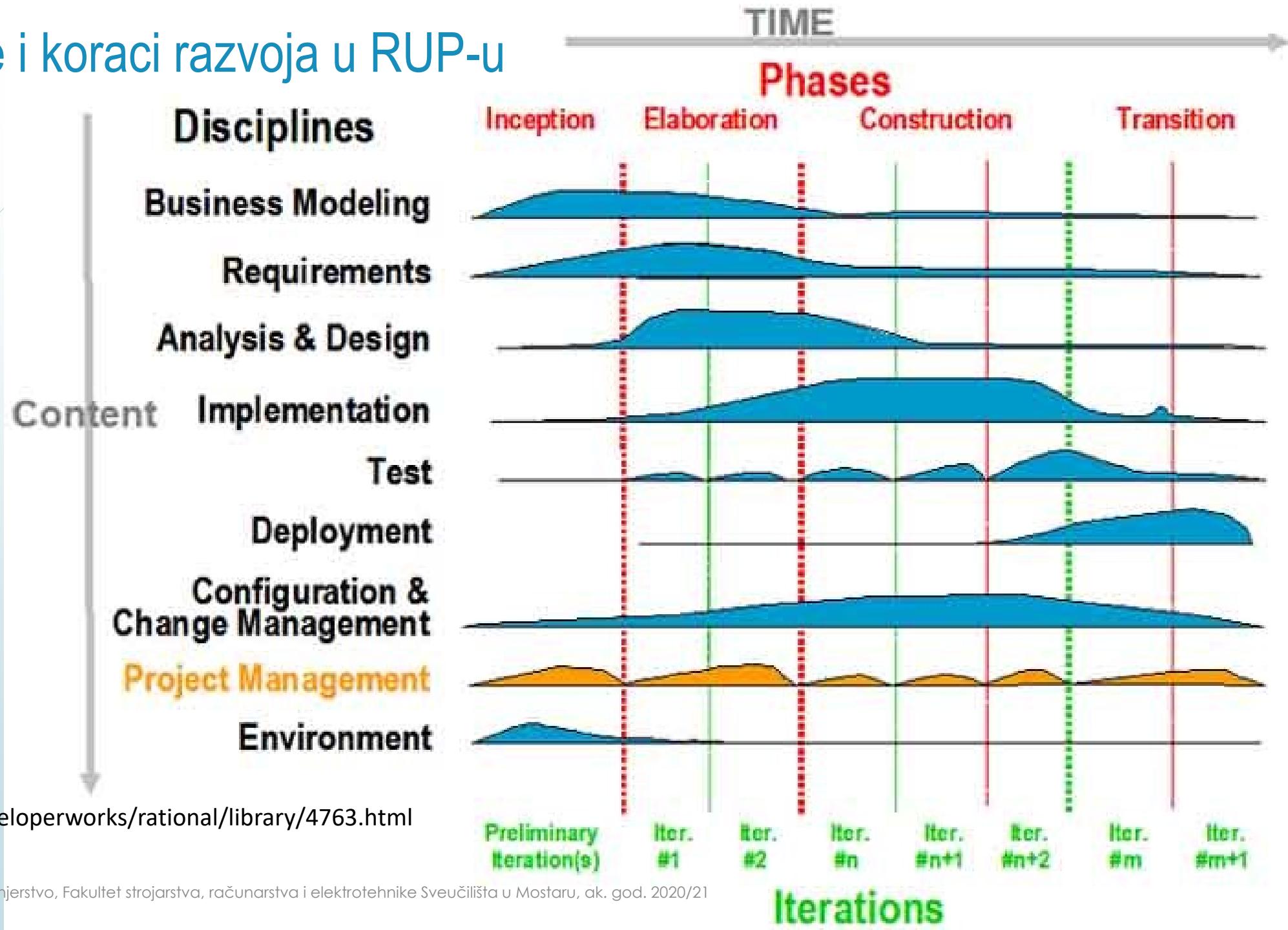
Glavne faze razvoja RUP-a

40

- ▶ Počinjanje (Inception)
 - ▶ opravdanje razloga i isplativosti pokretanje projekta
 - ▶ prikupljanje najvažnijih zahtjeva
 - ▶ određivanje dosega projekta, okvirnu cijenu i potrebno vrijeme
- ▶ Elaboracija (Elaboration)
 - ▶ prikupljanje detaljnih zahtjeva i uspostavljanje arhitekture rješenja
 - ▶ planiranje konstrukcije
- ▶ Konstrukcija, gradnja (Construction)
 - ▶ prikupljanje ostalih zahtjeva + promjene zahtjeva
 - ▶ razrada arhitekture i izrada sustava uz kontinuiranu integraciju
- ▶ Prijelaz (Transition)
 - ▶ beta testiranje, podešavanje performansi, poduka korisnika
 - ▶ provjera prihvatljivosti i zadovoljstva korisnika
- ▶ Post-implementacija (Post-deployment)
 - ▶ nastavak evolucijskog razvoja uz očuvanje integriteta aplikacije

Faze i koraci razvoja u RUP-u

41

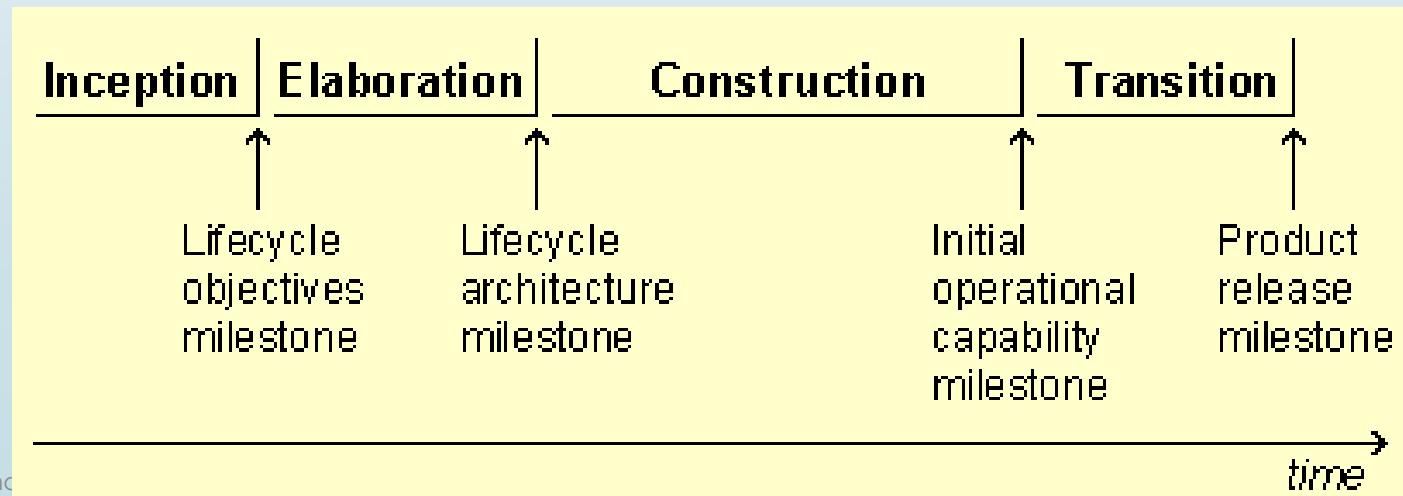


<https://www.ibm.com/developerworks/rational/library/4763.html>

Faze, discipline i kontrolne točke

42

- ▶ Horizontalna os predstavlja vrijeme i aspekte životnog ciklusa
 - ▶ cikluse, faze, iteracije i kontrolne točke
- ▶ Vertikalna os predstavlja discipline - logički grupirane aktivnosti
 - ▶ u ranijim fazama naglasak je na poslovnom modeliranju i zahtjevima
 - ▶ u kasnijima na implementaciji, testiranju i ugradnji te upravljanju izmjenama i konfiguracijom
 - ▶ disciplina upravljanja projektom ujednačenog je intenziteta.
- ▶ Osnovicu životnog ciklusa čine četiri slijedne faze od kojih svaka završava velikom kontrolnom točkom



Glavne discipline

43

- ▶ Poslovno modeliranje (Business Modeling)
 - ▶ Identificira poslovni kontekst sustava i oblik organizacije
 - ▶ Definiraju se ciljevi i okvirna funkcionalnost, poslovna pravila i sl.
- ▶ Requirements (Zahtjevi)
 - ▶ Definira kako saznati i prikupiti želje te ih pretvoriti u skup zahtjeva
- ▶ Analiza i dizajn (Analysis & Design)
 - ▶ Definira pretvorbu zahtjeva u dizajn
 - ▶ Analiza usmjerenja na logički pogled i funkcionalne zahtjeve
 - ▶ Dizajn usmjeren na fizički pogled i nefunkcionalne zahtjeve
- ▶ Implementacija (Implementation)
 - ▶ Kako razviti, organizirati, testirati i integrirati komponente
- ▶ Provjera (Test)
 - ▶ Kako testirati i procijeniti kvalitetu rješenja
- ▶ Uvođenje u primjenu (Deployment)
 - ▶ Aktivnosti potrebne da sustav bude dostupan krajnjim korisnicima

Podupiruće discipline

44

- ▶ Upravljanje konfiguracijom i promjenama (Configuration & Change Management)
 - ▶ kako kontrolirati i sinkronizirati evoluciju skupa komponenti i isporuka
- ▶ Upravljanje projektom (Project Management)
 - ▶ planiranje projekta, upravljanje rizicima, praćenje napretka i metrika
- ▶ Okolina (Environment)
 - ▶ organizira dijelove metodologije, procese i alate kao okruženje timu
- ▶ Životni ciklus iteracije: mini-vodopad (Mini-Waterfall)
 - ▶ usitnjeni standardni životni ciklus razviti
 - ▶ zasnovan i vođen na slučajevima korištenja

Glavne faze razvoja - Počinjanje

45

► Formuliranje opsega projekta

- opis problemskog konteksta te najvažnijih zahtjeva i ograničenja
- **prikupljanje najvažnijih zahtjeva (10% detaljno)**
- preporuča se istaknuti i kritične scenarije korištenja (UC scenariji)

► Inicijalna procjena ukupnog troška, vremena i rizika

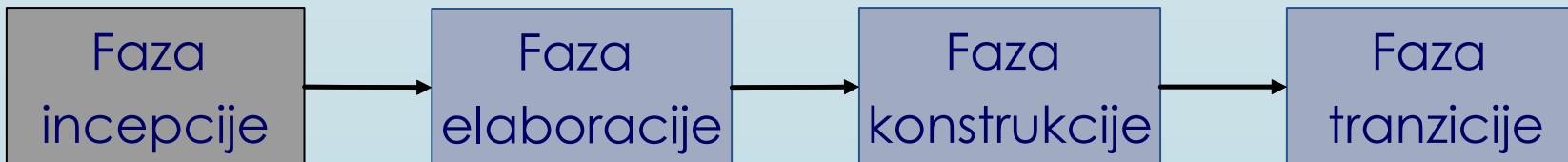
► Planiranje i priprema poslovnog slučaja

► Priprema okruženja za projekt

- Procjena projekta i organizacije, odabir alata, razvojnih okruženja i procesa

► Izrada prijedloga arhitekture

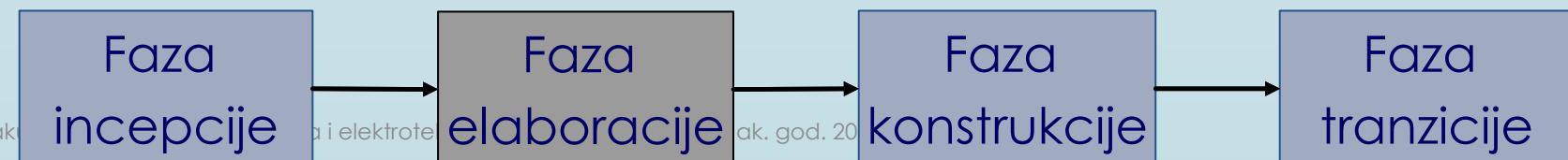
- demonstrirati izvedivost simulacijom, inicijalnim prototipom i sl.



Glavne faze razvoja – Elaboracija (razrada)

46

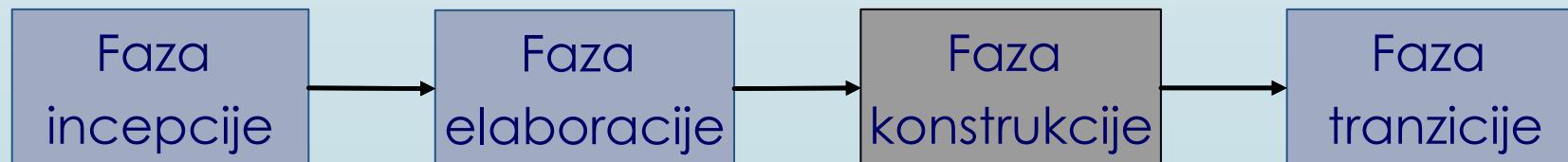
- ▶ Definiranje, validacija i zacrtavanje arhitekture
- ▶ Osiguranje da su arhitektura, zahtjevi i planovi stabilni, a rizici ublaženi
 - ▶ tako da se može pouzdano odrediti trošak i završetak projekta
- ▶ **Prikupljanje detaljnih zahtjeva (80%)**
- ▶ Ažuriranje vizije projekta
 - ▶ razumijevanjem kritičnih UC koji su ujedno i nositelji najvećih rizika
- ▶ Izrada plana iteracija za fazu konstrukcije
- ▶ Dorada razvojnog procesa i uspostava razvojnog okruženja
 - ▶ uključujući proces, alate i podršku za automatizaciju
- ▶ Dorada arhitekture i odabir komponenti
 - ▶ procjena potencijalnih komponenti - cijena i trajanje naredne faze



Glavne faze razvoja – Konstrukcija (izgradnja)

47

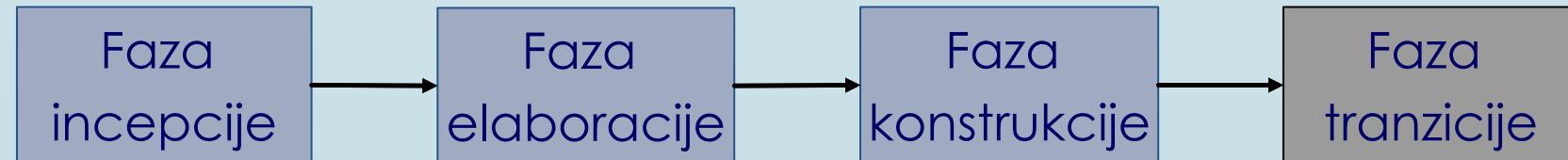
- ▶ Upravljanje resursima, kontrola projekta i optimizacija procesa
 - ▶ paralelni razvoj nekoliko razvojnih timova s ciljem ubrzanja razvoja
- ▶ Završetak iterativnog i inkrementalnog razvoja konačnog proizvoda
 - ▶ provjera prihvatljivosti
 - ▶ podrazumijeva dovršetak analize, dizajna, razvoja i testiranja
- ▶ Procjena razvijenih isporuka naspram definirane Vizije projekta
- ▶ Provjera da li su programska podrška, lokacije i korisnici spremni za beta isporuku



Glavne faze razvoja – Tranzicija (prijelaz)

48

- ▶ Izvršavanje planova uvođenja u primjenu
- ▶ Dovršavanje korisničke dokumentacije i uputa
- ▶ Poduka krajnjih korisnika i održavatelja
- ▶ Testiranje programskog rješenja na lokaciji isporuke
- ▶ Izrada isporuke (release) konačnog programskog rješenja
- ▶ Omogućavanje proizvoda dostupnim svim krajnjim korisnicima
- ▶ Prikupljanje povratne informacije od krajnjih korisnika
- ▶ Fino podešavanje rješenja (popravak manjih pogrešaka, poboljšanje performansi) na temelju povratne informacije



Agilni postupci razvoja

Ekstremno programiranje i Scrum kao neki od primjera agilnih postupaka razvoja

Moderniji (agilniji) pristupi

50

- Vodopadni model i RUP su samo neki od mnogih modela
 - Spadaju u tradicionalne pristupe (metodologije), planski usmjerene i dobro dokumentirane.
- Moderniji pristup – agilne metode
 - Scrum i XP među najpoznatijima
 - Prikladne za sustave sa slabo definiranim i/ili čestim promjenama zahtjeva, naglasak na čestim izdanjima i interakciji s korisnicima
 - Aktualni trend i dominantne na tržištu, ali ne bez kritičara
 - za Ivara Jacobssona (UML, RUP) korak natrag iz inženjerstva prema zanatu
 - Odmak od krutih prethodnih modela
 - Dokumentacija često u obliku kartica s korisničkim pričama uz upotrebu stvarnih ili virtualnih ploča sa statusima

Primjer evidencija korisničkih priča

i You can now add more backlog levels to this team. Access this setting through the [settings dialog](#).

Backlog items

	New	+	-	Create query	Column options	✉	Forecast Off	Mapping Off	Parents Hide	In progress items Show	⚙️	Filter	🔍	▼
Backlog	Board													
Past														
Sprint 1														
Sprint 2														
Sprint 3														
Bugovi i dorade 1														
Current														
Bugovi i dorade 2														
	Order	Work Item Type	Title		State	Ef Area Path	Iteration Path							
	19	Bug	Klik na rezultat na počeoj stranici treba ažurirati rezult...	New	Elektrijada\Andr...	Elektrijada\Bugovi i dorade 2								
	20	Bug	Promjena datuma za natjecanje u znanju 2x pita za dan,...	New	Elektrijada\Andr...	Elektrijada\Bugovi i dorade 2								
	21	Bug	U postavke za sport dodati koliko bodova nosi pobeda...	Committed	Elektrijada\Andr...	Elektrijada\Bugovi i dorade 2								Leon Hrn...
	22	Product Backlog Item	Rezultati krosa (M)	New	Elektrijada\SQL	Elektrijada								
	23	Product Backlog Item	Rezultati krosa (Ž)	New	Elektrijada\SQL	Elektrijada								
	24	Bug	Dodatne informacije se ne vidi u susretima po skupinama	Committed	Elektrijada\Andr...	Elektrijada\Bugovi i dorade 2								Leon Hrn...
	25	Bug	Dodati u tablici grupne faze broj bodova i po tome sorti...	Committed	Elektrijada\Andr...	Elektrijada\Bugovi i dorade 2								Leon Hrn...
	26	Bug	Dodati u postavke sporta tipove komparatora	Committed	Elektrijada\Andr...	Elektrijada\Bugovi i dorade 2								Leon Hrn...
	27	Bug	Ako je nešto u favoritima, onda zvjezdica mora biti obo...	New	Elektrijada\Andr...	Elektrijada\Bugovi i dorade 2								
	28	Bug	Gdje bi se jednostavno mogao vidjeti poredak u nekom...	New	Elektrijada\Andr...	Elektrijada\Bugovi i dorade 2								
	29	Product Backlog Item	Rezultati veslanja	New	Elektrijada\SQL	Elektrijada								
	30	Product Backlog Item	Prikaz rezultata znanja	New	Elektrijada\Web	Elektrijada\Sprint 3								Luka Juki...
	31	Product Backlog Item	Rezultati šaha	New	Elektrijada\SQL	Elektrijada								
	32	Product Backlog Item	Skica Android aplikacije	Committed	Elektrijada\Andr...	Elektrijada\Sprint 2								Ivan Oreh...
	33	Bug	Nemogu ukloniti događaj iz favorita	New	Elektrijada\Andr...	Elektrijada\Bugovi i dorade 2								
	34	Product Backlog Item	Rezultati - Osnove elektrotehnike	New	Elektrijada\SQL	Elektrijada								
	35	Product Backlog Item	Rezultati - Matematika 2	New	Elektrijada\SQL	Elektrijada\Sprint 2								Toni Baka...
	36	Product Backlog Item	Rezultati - Matematika 1	New	Elektrijada\SQL	Elektrijada\Sprint 2								Toni Baka...

Programsko inženjerstvo, Fakultet strojarstva, računarstva i elektrotehnike Sveučilišta u Mostaru, ak. god. 2020/21

Ploče sa statusima zadataka (1)

52

► Po uzoru na Kanban ploče (stupci propisani pojedinim modelom/metodologijom)

The image shows a digital Kanban board with the following columns and tasks:

- Ideas**:
 - Korisnici glasaju na koji izlet žele ići
 - Korisnik može predlagati ili glasati za redoslijed putovanja
 - Generiranje brošure puta
 - Integracija s vanjskim kalendarom (npr. Google calendar)
 - Grupa korisnika šalje agenciji prijedlog putovanja
 - Sustav šalje podsjetnike na putovanja i plaćanja
 - Agencija evidentira uplatu
 - Konferencija: sudionik, predavač i organizator
 - Sudionici mogu vidjeti predavanja na konferenciji, ocijeniti predavanje i dodati podsjetnik u kalendar
 - Organizator može unijeti ponude od agencija
 - Organizatori mogu glasati za najbolju ponudu
 - Omogućiti pregled putovanja po
- To Do**:
 - Modul za administraciju korisnika
 - Kreirati web-aplikaciju sa web servisom koji vraca trenutno vrijeme
 - + Add another card
- In Progress**:
 - Agencija može otkazati putovanje
 - Agencija može vidjeti popis prijavljenih korisnika za putovanje
 - Korisnici pregledavaju i daju nove sugestije oko autobusa i hotela
 - Korisnik unosi recenziju puta
 - uvedi swagger
 - + Add another card
- Pull Request**:
 - + Add a card
- Testing**:
 - Složiti docker?
 - Pripremiti Camundu s nekim trivijalnim procesom
 - Generirati kod koji puni bazu s nekoliko inicijalnih podataka
 - Razdvojiti entitet Putovanje na 2 entiteta (konkretno putovanje s datumom i općenito putovanje)
 - Agencija objavljuje putovanja
 - Agencija ažurira putovanja
 - Ažurirati verziju na herokuapp
 - + Add another card
- Done**:
 - Kostur angular klijenta word)
 - Popraviti .gitignore
 - Integrirati s bitbucket
 - Korisnik može otakaz
 - Korisnik se prijavljuje :
 - Korisnik se može prijava formu za prijavu
 - Uspostaviti hosting
 - Pripremiti bazu podat
 - Korisnik pregledava pi putovanja
 - + Add another card

Ploče sa statusima zadataka (2)

53

New

Approved

0/5

Committed

21/5

Done

New item



Na početnoj stranici ne radi ažuriranje/brisanje događaja

Definirati build i deploy za web aplikaciju



Boris Milašinović

Login preko FER weba



Luka Jukić

Favoriti, sport i znanje prikazati na ekranu, a ne kroz opcije menija

Obojane zvjezdice za favorite na random mjestima

Uvoz podataka iz Excela

Vizualni prikaz rezultata nekog sporta

Approved

0/5

Committed

21/5

0/3

Done

Stranica s postavkama za izračun kako se boduje i organizira koje natjecanje



Tomislav Maslač

Bluetooth prijenos



Vjeran Hanžek

Bluetooth prijenos



Toni Bakarčić

Brisati grupu koja je trenutno označena, a ne zadnju.



Leon Hrnjak

Omogućiti unos pozicije tima u znanju



Prijava rezultata u sportovima



Luka Jukić

Skratiti prikaz kod uspjeha fakulteta

Dodati RadioButtone za prikaz bodova ili pozicije u rezultatima



Toni Bakarčić

Prikaz pozicija u rezultati u znanju ne radi



Tomislav Božurić

Dodati mogućnost da se pojedinac u znanju natječe individualno



Tomislav Božurić

Sinkronizacija s webom sruši aplikaciju



Leon Hrnjak

Aktivirati formu za vijesti



Leon Hrnjak

Proglas agilnosti

54

- ▶ agilitas (lat.) - svojstvo brzine, okretnosti, hitrosti, lakoće, radinosti
- ▶ Manifest agilnosti (objava, proglas)
 - ▶ Kent Beck, Mike Beedle, Arie van Bennekum, Alistair Cockburn, Ward Cunningham, Martin Fowler, James Grenning, Jim Highsmith, Andrew Hunt, Ron Jeffries, Jon Kern, Brian Marick, Robert C. Martin, Steve Mellor, Ken Schwaber, Jeff Sutherland, Dave Thomas
 - ▶ skijalište Snowbird, Utah, 2001.
 - ▶ *Pojedinci i interakcije* važniji od procesa i alata
 - ▶ *Softver koji radi* važniji od sveobuhvatne dokumentacije
 - ▶ *Suradnja s naručiteljem/korisnikom* važnija od pregovora o ugovoru
 - ▶ *Odziv na promjenu* važniji od praćenja plana

Načela (principi) agilnosti

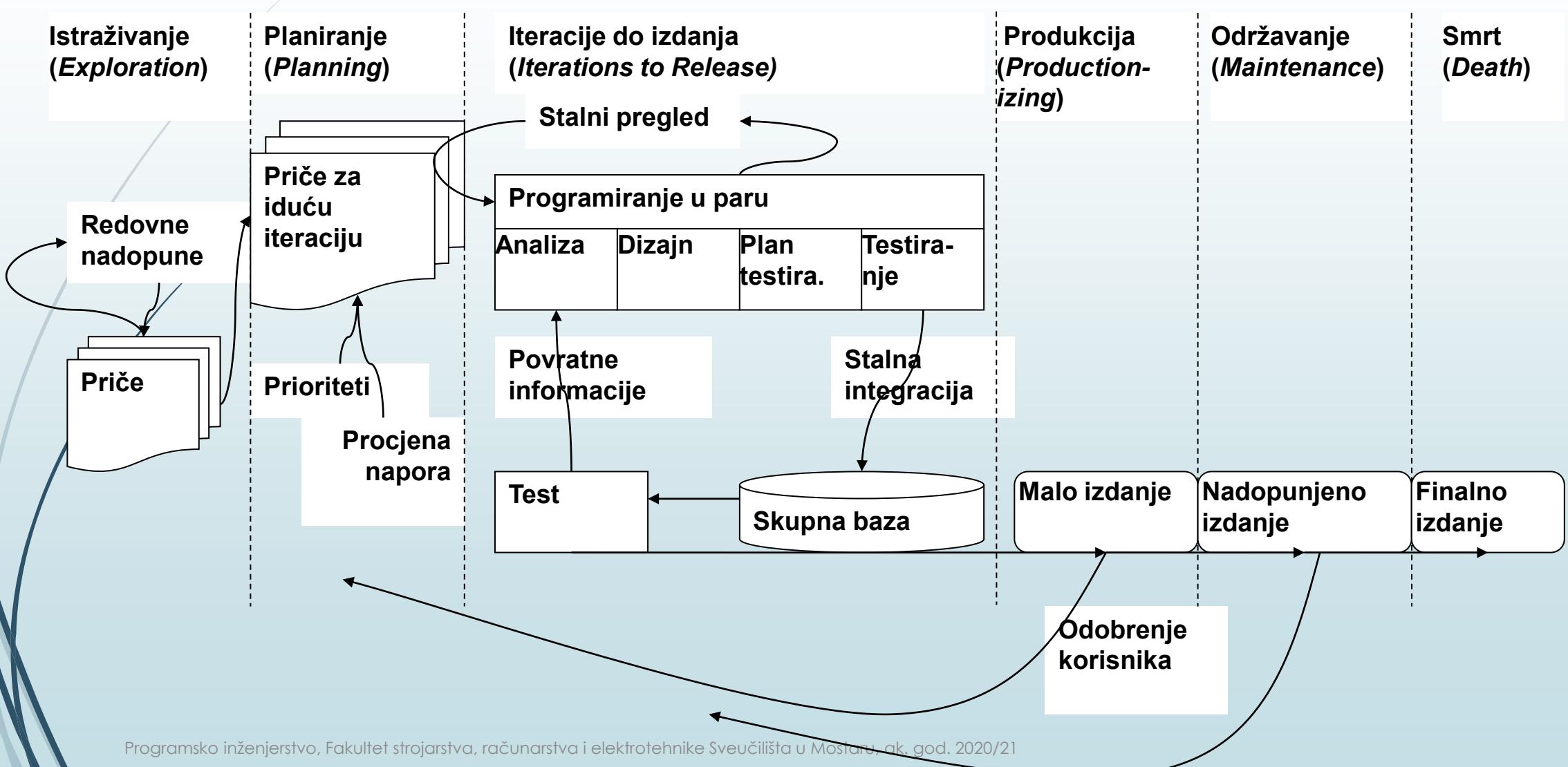
55

- ▶ Zadovoljstvo korisnika ranim i kontinuiranim isporukama softvera
- ▶ Promjene zahtjeva se željno prihvaćaju, čak i u kasnoj fazi razvoja
- ▶ Česta i što ranija isporuka softvera koji radi – 2x/mj do 1x/nekoliko mj
- ▶ Česta (dnevna) suradnja „poslovnjaka“ i razvojnika
- ▶ Motiviranje pojedinaca za rad u projektu – okruženje, povjerenje
- ▶ Usmena komunikacija - najbolja metoda za razmjenu informacija
- ▶ Glavna mjera napretka - softver koji radi
- ▶ Održivi razvoj - sponzori, tim i korisnici - održavati stalni tempo
- ▶ Kontinuirana pažnja na tehničku izvrsnost i dobar dizajn
- ▶ Jednostavnost je nužna
- ▶ Najbolje arhitekture, zahtjevi i dizajn - iz samoorganiziranih timova
- ▶ Tim spoznaje kako postati efektivniji, a zatim se često prilagođava

Ekstremno programiranje

56

► Životni ciklus:



Faze ekstremnog programiranja (1/3)

57

► Istraživanje

- Korisnici bilježe svoje priče na kartice
- Svaka kartica sadrži jednu mogućnost programa.
- Projektni tim se pobliže upoznaje s alatima, tehnologijom i postupcima projekta.
- Radi se prototip sustava za testiranje tehnologije i varijanti arhitekture sustava.
- Faza istraživanja traje nekoliko tjedana do nekoliko mjeseci

► Planiranje

- Postavlja prioritete na korisničke priče (tj. svojstva programskog rješenja)
- Planira se doseg prvog malog izdanja i vrijeme za pojedinu karticu
- Zatim se određuje cijelokupni vremenski raspored.
- Rok za izdavanje prvog malog izdanja obično je unutar dva mjeseca.
- Faza planiranja traje nekoliko dana.

Faze ekstremnog programiranja (2/3)

58

► Iteracije do izdanja (razvoj)

- Uključuje nekoliko iteracija sustava prije prvog izdanja.
- Vremenski raspored iz faze planiranja se razlaže u više iteracija
- Pojedina iteracija traje jedan do četiri tjedna.
- Prva iteracija stvara takav sustav koji obuhvaća cijelu arhitekturu ciljanog sustava.
- Klijent određuje kartice koje će se koristiti pri svakoj narednoj iteraciji.
- Testovi prihvatljivosti izvode se na kraju svake iteracije.
- Na kraju posljednje iteracije, sustav je spremан за produkciju.

► Producija

- Dodatno testiranje i provjera performansi sustava prije isporuke klijentu.
- Razrješenje primjedbi na sustav te odlučivanje da li će se riješiti u ovom izdanju.
- Iteracije trajanja tri do najviše tjedan dana.
- Zakašnjele nove ideje i prijedlozi se dokumentiraju i njihova implementacija odgađa.

► Održavanje

- nakon što je prvo izdanje pušteno u produkciju
- XP projekt mora istovremeno održavati softver u primjeni i proizvoditi nove iteracije
- Zbog toga se brzina implementacije smanjuje
- Održavanje može zahtijevati nove članove projektnog tima i promjenu strukture tima.

► Faza smrti je blizu kada klijent nema više novih kartica s pričama

- Podrazumijeva se da sustav zadovoljava sve zahtjeve (npr. pouzdanost i stabilnost).
- Vrijeme u XP projektu da se konačno napiše sva korisnička dokumentacija budući da više nema promjena na arhitekturi, dizajnu i kodu sustava.
- Smrt može nastupiti i kada sustav ne ispunjava sva korisnička očekivanja, ili ako postane preskup za daljnji razvoj.

Temeljne vrijednosti

60

- ▶ Komunikacija (communication)
 - ▶ Verbalna i elektronička, učestala / stalna, svih dionika
- ▶ Jednostavnost (simplicity)
 - ▶ Najjednostavniji mogući dizajn (KISS), samo što je potrebno i zatraženo
 - ▶ kontinuiranim refaktoriranjem i minimizacijom dokumentacije
- ▶ Povratne informacije (feedback)
 - ▶ Od korisnika i unutar tima – što ranije te češće
- ▶ Hrabrost (courage)
 - ▶ Akcije i (teške, nepopularne) odluke
 - ▶ npr. odbacivanje dijelova (YAGNI), ili veće promjene u kasnoj fazi projekta
- ▶ Uvažavanje (respect)
 - ▶ svatko doprinosi timu; i mali doprinos je vrijedan
 - ▶ izbjegavanje promjena koje bi onesposobile aktualnu verziju ili usporile napredak ostalih

► Priče (korisničke priče) - Stories (User Stories) != Use Case

- kratki opis funkcionalnosti, nekoliko rečenica
- toliko da se može odrediti napor
- „A promise to have a conversation” (Cockburn)

► Tjedni ciklus (Weekly Cycle)

- razvoj u tjednim ciklusima, tjedan započinje sastankom izbora priča
- tjedan ne mora započeti u ponedjeljak

► Kvartalni ciklus (Quarterly Cycle)

- grublje, na dulje staze, razvoj se planira kvartalno („rolling wave“)

► Rezerva (Slack)

- Zadaci niskog prioriteta koji mogu biti odbačeni ako projekt kasni

- ▶ Smještaj ekipe (Sit Together)
 - ▶ Kolocirana ekipa, otvoreni prostor
- ▶ Cjelovitost i zajedništvo ekipe (Whole Team)
 - ▶ Cjelovitost kompetencija, osjećaj pripadnosti
- ▶ Informativno radno okruženje (Informative Workspace)
 - ▶ Ploče, „visible wall graphs“
- ▶ Energičan rad (Energised Work)
 - ▶ Odmorni programeri – produktivnost – ograničeni prekovremeni rad
- ▶ Programiranje u paru (Pair Programming)
 - ▶ Vodič (driver) i promatrač (observer, navigator)
 - ▶ Zamjena uloga ali i partnera

- ▶ Inkrementalni dizajn (Incremental Design)
 - ▶ Nema velikog oblikovanja unaprijed (BDUF – big design up-front)
 - ▶ dizajn kao kontinuirani proces malih koraka – refaktoriranje
- ▶ Test prije programiranja (Test-First Programming)
 - ▶ Testovi trebaju biti napisani prije kodiranja te automatizirani
- ▶ Desetminutna gradnja (Ten-Minute Build)
 - ▶ sustav se mora moći kompilirati i testirati unutar 10 minuta
 - ▶ da bi mogao postići odgovarajuću povratnu informaciju (feedback)
- ▶ Kontinuirana integracija (Continuous Integration)
 - ▶ Svakih nekoliko sati ili nakon promjena → dnevno

- ▶ Klijent, korisnik (customer)
 - ▶ piše priče i određuje prioritete – ne mora nužno biti krajnji korisnik
- ▶ Programer, razvojnik (developer)
 - ▶ kodiranje, pisanje testova, refaktoriranje
- ▶ Tester
 - ▶ Izrada i izvođenje testova, održavanje alata za testiranje
- ▶ Trener (coach)
 - ▶ Savjet, nadzor, kontrola (issue control) # tech lead
- ▶ Upravitelj (manager, big boss)
 - ▶ Upravljanje ekipom, rješavanje problema # team lead
- ▶ Druge uloge
 - ▶ Tracker, Doomsayer, ...
 - ▶ Product Manager, Domain Expert, Interaction Designer, Business Analyst ...

Scrum

65

- ▶ Naziv potječe iz ragbija, gdje se formira skup igrača (engl. scrum) za početak igre nakon prekida
- ▶ jednostavni upravljački okvir za inkrementalni razvoj
 - ▶ što jednostavniji proces, ne bavi se tehnikalijama
- ▶ iteracija = sprint
- ▶ rezultat sprinta – potencijalno isporučivi inkrement proizvoda (engl. shippable)



► Ekipa (Scrum Development Team)

- jedna ili više ekipa od po 7 plus/minus 2 člana
- svestrani članovi (cross-functional)
- samoorganizirajuća ekipa (self-organizing)

► Vlasnik proizvoda (Product Owner)

- zadužen za plan, prioritete, troškove i povrat investicije
- Održava Product Backlog

► Majstor (Scrum Master)

- brine o procesu, koordinira, ali ne donosi poslovne ni tehničke odluke

Scrum - Artefakti

67

► Product Backlog

- neizvršen rad, preostali posao
- lista poželjne funkcionalnosti
- vidljiva svim dionicima
- svatko može dodati elemente

► Product Backlog Item - element

- definira "ŠTO", najčešće kao korisnička priča
- ima kriterij prihvatljivosti, definiciju "dovršenosti"
- sadrži više zadataka
- poslovnu vrijednost odredi Vlasnik
- napor procijeni Ekipa

► Sprint Backlog

- popis zadataka i statusa
- ažuriran tokom sprinta
- vidljiv Ekipi

► Zadatak sprinta (Sprint Task)

- "KAKO" za PBI "ŠTO"
- dan posla ili manje
- preostali napor procjenjuje se dnevno u satima

Životni ciklus Scruma

68

► Prije igre (pre-game)

- podfaze: Planiranje i Dizajn/Arhitektura
- izrađuje se radna lista proizvoda (Product Backlog - PB)
 - u PB se konstantno zapisuju zahtjevi, procjene napora i prioriteti

► Razvoj / "igra" (development / game)

- razvoj iterativnim ciklusima, takozvanim sprintovima
- sprintovi - okvirno jednakog trajanja, 30 dana (prema knjizi)
 - tjedan do tri u praksi
- sprint ima sve faze klasičnog ciklusa (RUP ima mini-vodopad !)
 - zahtjeve, analizu, dizajn, evoluciju, test i isporuku
- tri do osam sprintova dok sustav ne bude spreman za distribuciju

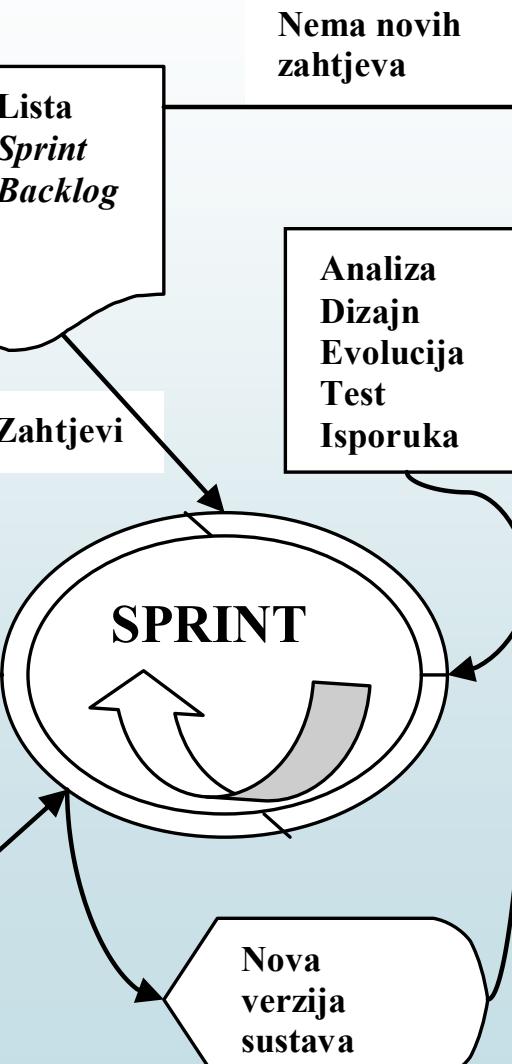
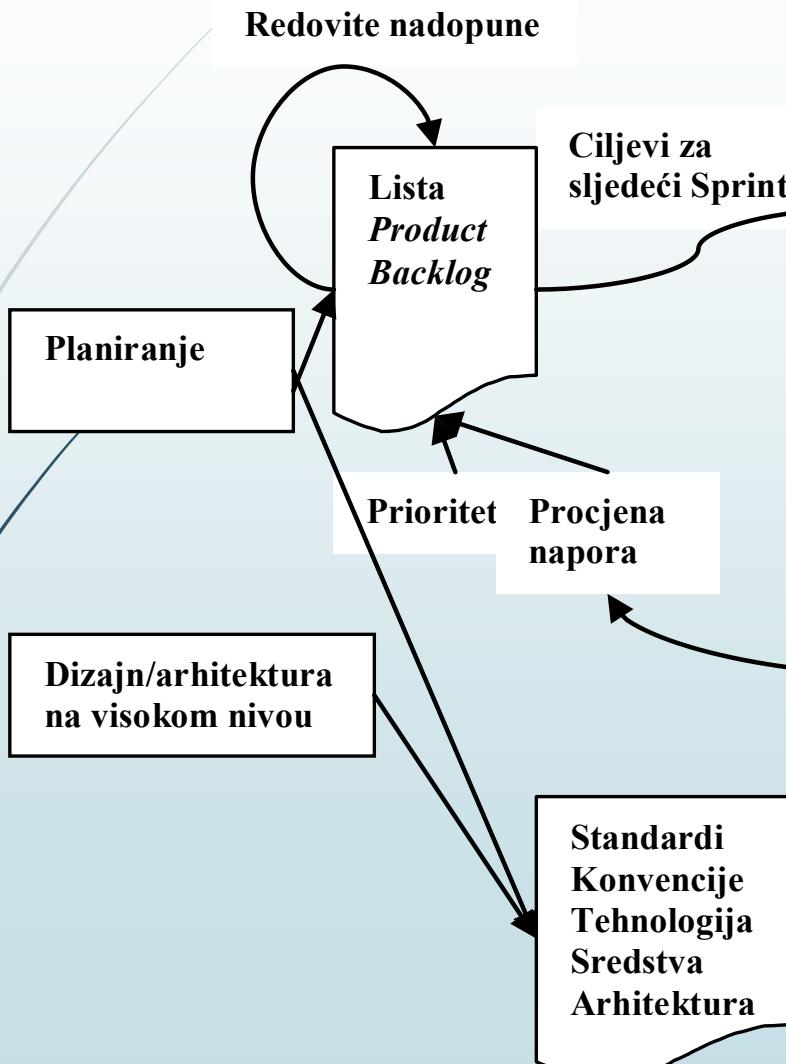
► Poslije igre (post-game)

- priprema sustav za izdanje kroz integraciju, testiranje i druge aktivnosti

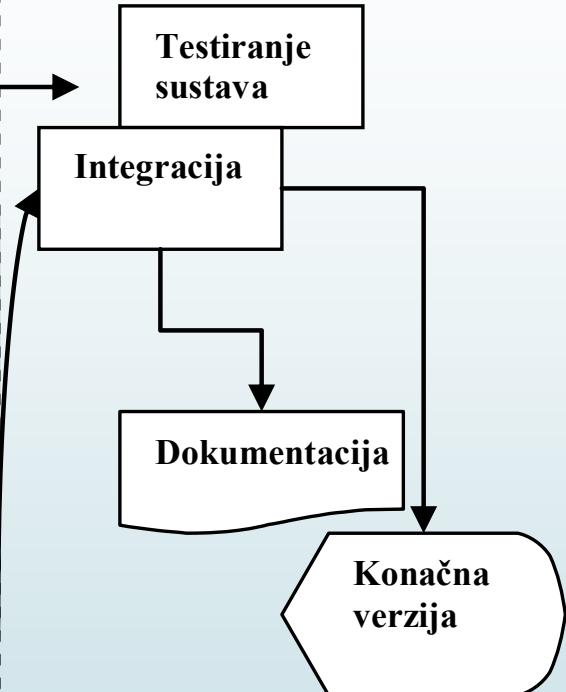
Scrum - Proces

Razvoj (Igra)

Prije igre



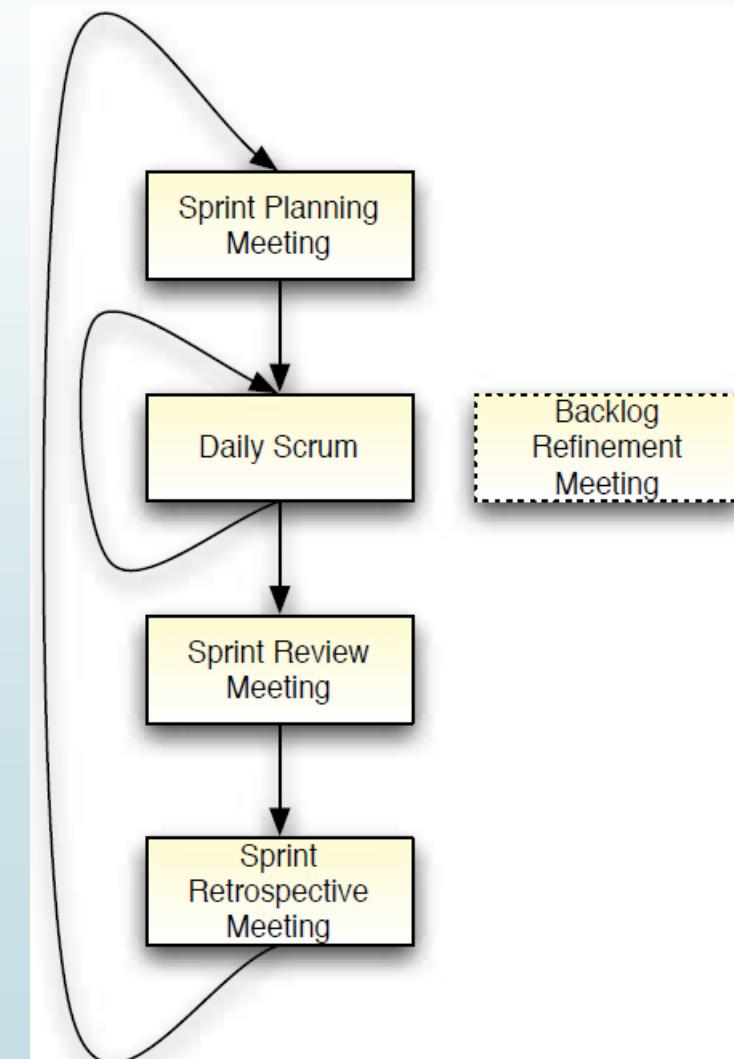
Poslije igre



Scrum - Sastanci

70

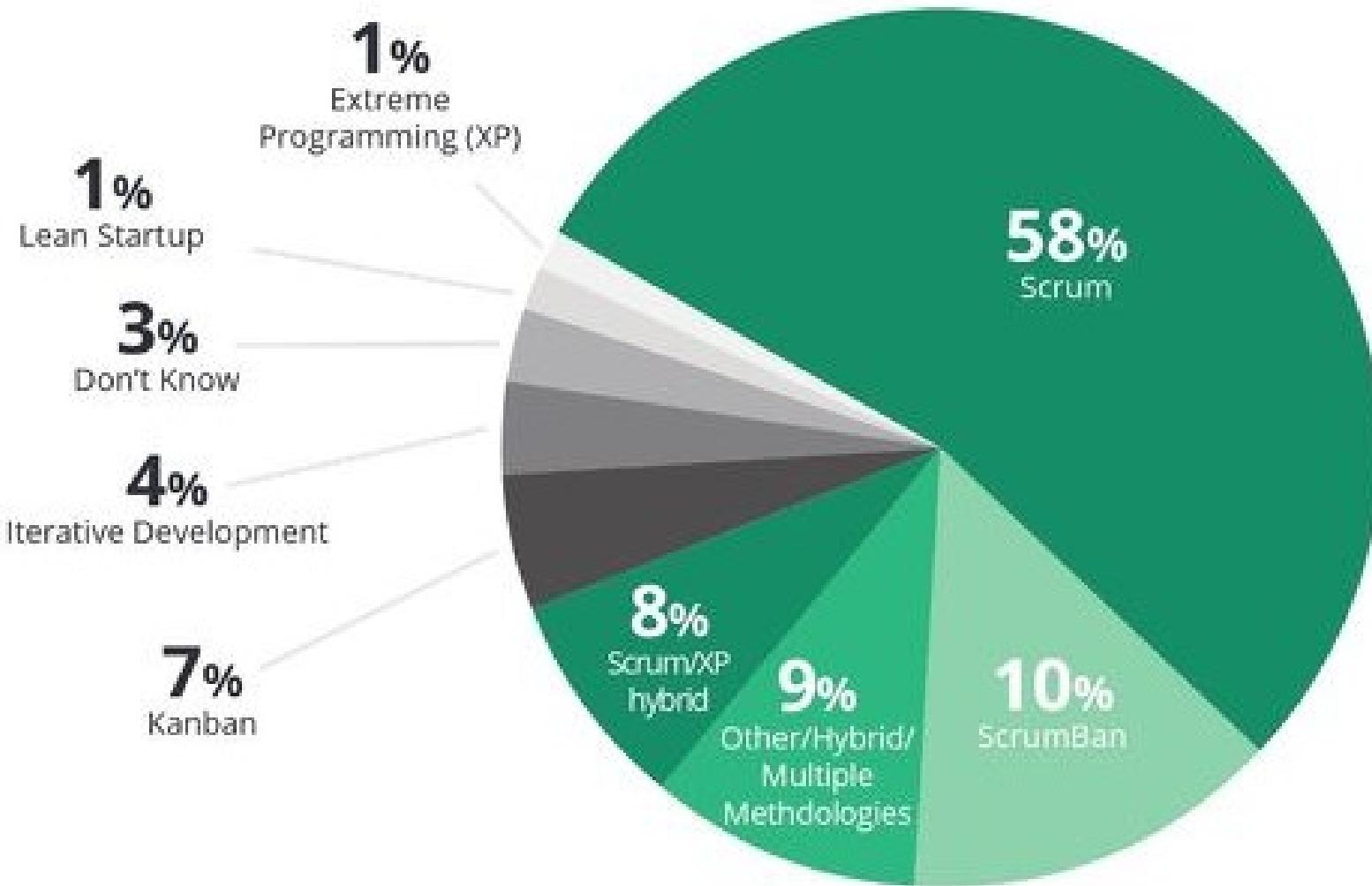
- ▶ Planiranje sprinta
 - ▶ 1. na početku iteracije – koji PB elementi idu u sprint
 - ▶ 2. tim dekomponira PB elemente u listu zadataka
 - ▶ 30d sprint planira se max 8 sati
- ▶ Dnevni Scrum
 - ▶ 15min, članovi tima međusobno
 - ▶ "standup meeting" – dojam užurbanosti
- ▶ Pregled sprinta
 - ▶ demonstracija inkrementa na kraju sprinta
 - ▶ Vlasnik deklarira "dovršeno"
 - ▶ ostalo ide u naredni sprint
- ▶ Retrospektiva sprinta
 - ▶ samoanaliza procesa
- ▶ Pročišćavanje preostalog posla
 - ▶ podjela, procjena, prioriteti ...



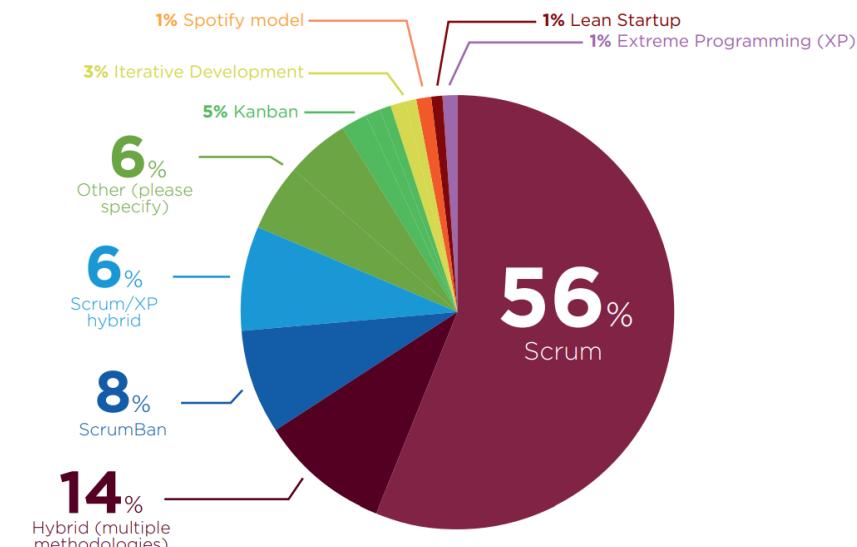
Upotreba agilnih metoda u praksi

71

Izvor: CollabNet VersionOne 14th Annual State of Agile Report
<https://stateofagile.com/#ufh-i-615706098-14th-annual-state-of-agile-report/7027494>



Izvor: CollabNet VersionOne
12th Annual State of Agile Report

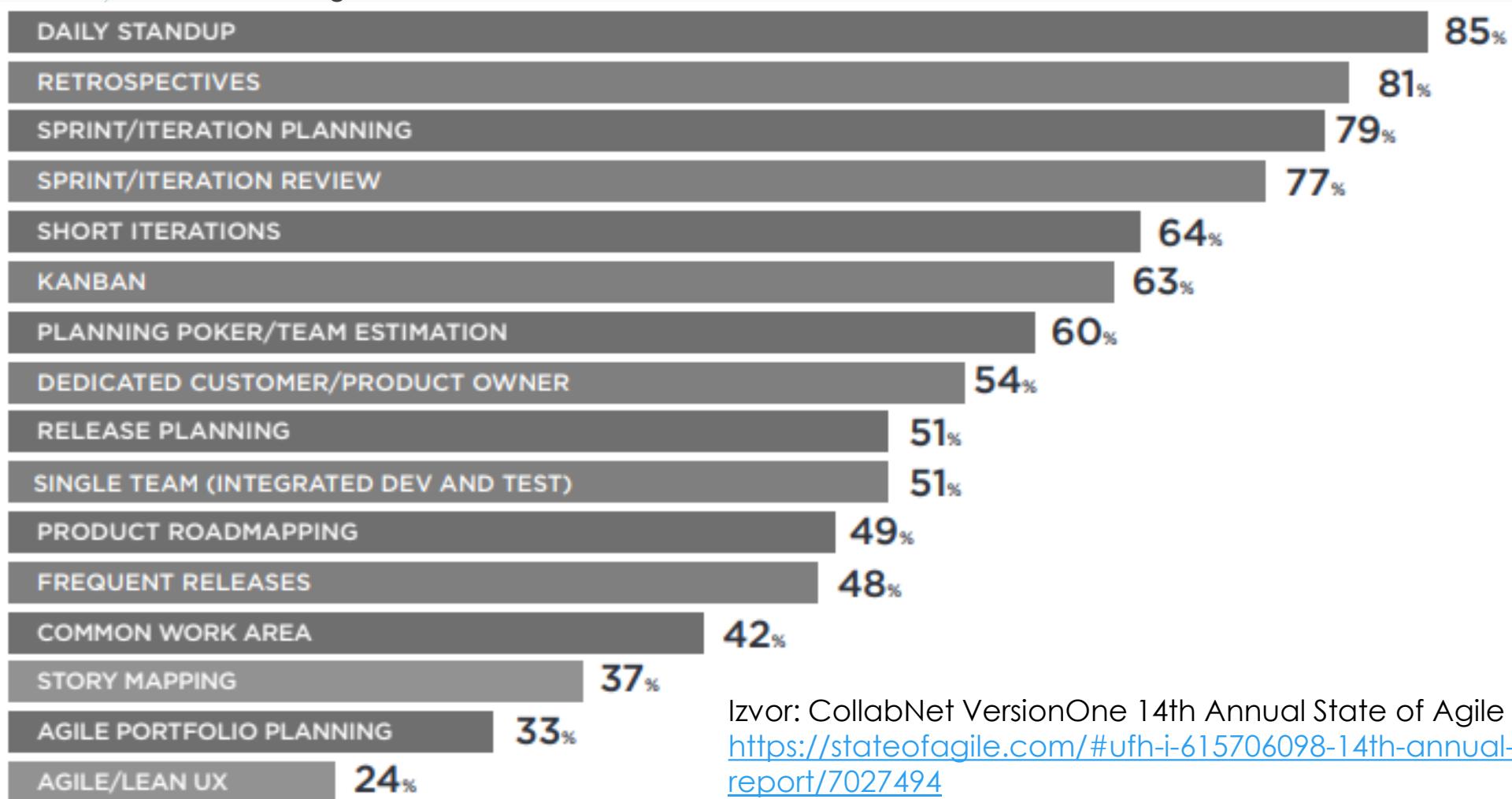


Odabir modela i metodologije

72

► Nema najboljeg pristupa, ovisi o vrsti problema, ekipi

► često prisutno „krojenje po mjeri“ (*method tailoring*) i korištenje dobrih praksi van strogog metodološkog okvira



Izvor: CollabNet VersionOne 14th Annual State of Agile Report
<https://stateofagile.com/#ufh-i-615706098-14th-annual-state-of-agile-report/7027494>

Korištenje inženjerskih praksi

73

► O nekim od navedenih inženjerskih praksi detaljnije naknadno

