



Ured koordinatore za reformu javne uprave  
Канцеларија координатора за реформу јавне управе  
Public Administration Reform Coordinator's Office


**Naziv projekta:**  
**“Izrada i uspostavljanje  
okvira interoperabilnosti i standarda za razmjenu podataka”**

**Vodilje i standardi za  
arhitekturu sistema i razvoj aplikacija**

**Projekat implementira:**  
**Infodom d.o.o.**




**Projekat finansira:**  
**Fond za reformu javne uprave u BiH**


	PROJEKT IZRADA I USPOSTAVLJANJE OKVIRA INTEROPERABILNOSTI I STANDARDA ZA RAZMJENU PODATAKA  <b>VODILJE I STANDARDI ZA          ARHITEKTURU SUSTAVA I RAZVOJ APLIKACIJA</b>	Oznaka: PARCO-IFBIH
		Page: 2 of 88

## Sadržaj


1	Uvod.....	6
1.1	Obim dokumenta.....	6
1.2	Namjena i korištenje dokumenta .....	7
1.3	Povezani dokumenti.....	9
1.4	Definicije i skraćenice.....	10
1.5	Popis slika, tablica i priloga .....	12
2	Strateški pogled.....	13
2.1	Elektronske usluge i javni registri za potrebe građana, privrede i društva .....	13
2.2	Upravljanje efektom u javnoj upravi (eng. Government Performance management).....	14
2.3	Interoperabilnost kao omogućitelj efikasne nove elektronske uprave .....	15
2.4	EU perspektiva i facilitatori interoperabilnosti .....	17
3	Procesni pogled.....	19
3.1	Upravljanje poslovnim procesima .....	19
3.2	Modeliranje poslovnih procesa i procesna arhitektura .....	19
3.3	Metodologija izrade modela poslovnih procesa .....	23
4	Funkcionalni pogled .....	24
4.1	Funkcionalna arhitektura .....	24
4.1.1	Višeslojna arhitektura .....	25
4.1.2	Komponentna arhitektura .....	26
4.1.3	Servisno orijentisana arhitektura .....	26
4.2	Poslovna pravila.....	27
4.3	Podaci (strukture, poruke, dokumenti).....	27
4.4	Sigurnosni mehanizmi (uloge, grupe, protokoli).....	27
4.5	Sučelje (GUI).....	28
4.6	Infrastruktura.....	28
4.7	Općenite smjernice.....	29
5	Podatkovni pogled.....	30
5.1	Modeliranje podataka i podatkovna arhitektura .....	30
5.2	Standardi za opisivanje i razmjenu podataka.....	30
5.2.1	XML .....	31

	PROJEKT IZRADA I USPOSTAVLJANJE OKVIRA INTEROPERABILNOSTI I STANDARDA ZA RAZMJENU PODATAKA  <b>VODILJE I STANDARDI ZA          ARHITEKTURU SUSTAVA I RAZVOJ APLIKACIJA</b>	Oznaka: PARCO-IFBIH
		Page: 3 of 88

5.2.2	XML Shema .....	32
5.2.3	Prateći XML standardi .....	33
5.2.4	Semantičko označavanje tekstualnih HTML dokumenata (Open Linked Data) .....	34
5.3	Zajednička biblioteka jezgrenih komponenti .....	35
5.4	Baze podataka .....	38
6	Integracioni pogled i razmjena podataka/dokumenata/poruka .....	40
6.1	Integracija IT sistema u državnoj upravi .....	40
6.2	Servisne sabirnice javne uprave (GSB – Government Service Bus) .....	42
6.2.1	Funkcije servisnih sabirnica javne uprave .....	42
6.2.2	Načela uvođenja servisne sabirnice .....	45
6.2.3	Nadzor rada integracijske platforme .....	47
6.3	Sigurnost i zaštita podataka .....	48
6.3.1	Nadležnosti i osiguranje zadržavanja vlasništva nad podacima pri realizaciji interoperabilnosti .....	48
6.3.2	Sigurnost informatičkog sistema .....	48
6.3.3	Sigurnost komunikacionih kanala .....	49
6.3.4	Sigurnost i zaštita podataka na upravno-servisnoj sabirnici .....	49
7	Tehnološki pogled - razvoj aplikacionih rješenja .....	51
7.1	Modeliranje i razvoj aplikacionog rješenja .....	51
7.2	Životni ciklus aplikacija .....	51
7.2.1	Upravljanje zahtjevima .....	52
7.2.2	Dizajn i arhitektura .....	52
7.2.3	Razvoj i testiranje .....	52
7.2.4	Isporuka i puštanje u rad (deployment) .....	53
7.2.5	Održavanje .....	54
7.2.6	Optimizacija .....	54
7.3	Nadogradnje i poboljšanja .....	54
7.3.1	Analiza utjecaja .....	55
7.3.2	Dizajn .....	55
7.3.3	Realizacija .....	56
7.3.4	Testiranje .....	56
7.3.5	Uvođenje u rad .....	57
7.4	Načela implementacije servisno orijentisane arhitekture u aplikativnim sistemima .....	57
7.4.1	Sučelja servisa .....	59
7.4.2	Eksplicitne granice servisa .....	60


	PROJEKT IZRADA I USPOSTAVLJANJE OKVIRA INTEROPERABILNOSTI I STANDARDA ZA RAZMJENU PODATAKA  <b>VODILJE I STANDARDI ZA          ARHITEKTURU SUSTAVA I RAZVOJ APLIKACIJA</b>	Oznaka: PARCO-IFBIH
		Page: 4 of 88

7.4.3	Dijeljenje servisnih opisa i shema a ne klasa.....	60
7.4.4	Upravljanje pomoću politika.....	61
7.4.5	Autonomija servisa.....	61
7.4.6	Korištenje formata za prijenos a ne programskih API-eva .....	61
7.4.7	Orjentacija na elektronskedokumente .....	62
7.4.8	Labavo vezanje sistema.....	62
7.4.9	Usklađenost s standardima .....	63
7.4.10	Nezavisnost od proizvođača.....	63
7.4.11	Upravljanje meta-podacima.....	64
7.5	SOA Governance .....	64
7.5.1	Definicija SOA Governancea.....	64
7.5.2	Obim SOA Governancea.....	65
7.5.3	SOA Governance Framework .....	65
7.5.4	SOA Governance Tehnologija.....	67
8	Upravljanje uslugama (eng. service management), sigurnost i drugi nefunkcionalni zahtjevi 68	
8.1	Upravljanje incidentima .....	68
8.2	Upravljanje konfiguracijama .....	69
8.3	Upravljanje dostupnošću .....	69
8.4	Upravljanje kapacitetima .....	70
8.5	Upravljanje kontinuitetom .....	71
9	Zaključak.....	72
	Prilog 1. Katalog tehnoloških standarda .....	74
	Prilog 2. Bibliografija.....	88

	PROJEKT IZRADA I USPOSTAVLJANJE OKVIRA INTEROPERABILNOSTI I STANDARDA ZA RAZMJENU PODATAKA  <b>VODILJE I STANDARDI ZA          ARHITEKTURU SUSTAVA I RAZVOJ APLIKACIJA</b>	Oznaka: PARCO-IFBIH
		Page: 5 of 88

**Historija verzija:**

Datum	Verzija	Opis
21-02-2012	1.0	Inicijalna verzija
3-7-2012	2.0	Ažurirano prema komentarima eksperata (VM BiH)
4-7-2012	2.1	Usklađivanje s izmjenama u K1.2 poglavlje 6
12.9.2012	2.2	Usklađivanje s K1.2 prema komentarima RS
18.10.2012	2.3	Dorade prema komentarima VM BiH

	PROJEKT IZRADA I USPOSTAVLJANJE OKVIRA INTEROPERABILNOSTI I STANDARDA ZA RAZMJENU PODATAKA  <b>VODILJE I STANDARDI ZA          ARHITEKTURU SUSTAVA I RAZVOJ APLIKACIJA</b>	Oznaka: PARCO-IFBIH
		Page: 6 of 88

## 1 Uvod

### 1.1 Obim dokumenta

Ovaj dokument dio je pripremne dokumentacije za izradu Okvira interoperabilnosti u BiH u okviru Komponente 3. Razvijanje standarda za arhitekturu sistema i razvoj aplikacija:

1. Vodilje i standardi za arhitekturu sistema i razvoj aplikacija
2. Nacrt Uputstva za primjenu standarda za arhitekturu sistema i razvoj aplikacija

Dokument je namijenjen Implementacijskom timu i Nadzornom timu Projekta, čiji sadržaj služi kao platforma za izradu Okvira interoperabilnosti za pojedine vlade, a zasniva se na prethodno urađenoj analizi „Najbolje prakse, iskustva i trendovi u BiH i drugim zemljama“ i „Evropskom okviru interoperabilnosti 2.0“ (Evropska komisija 2011), „White Paper on Enterprise Architecture“ (Ministry of Science, Technology and Innovation, Danska, 2003.), SAGA 4.0 i 5.0 (Standards and Architectures for e-Government Applications, Njemačka, 2008.) te “Standardima za opremu i softver” (IDDEEA, 2010.).

Glavni cilj ovog dokumenta je predložiti koncepte, pravila i standarde za arhitekturu informacionih sistema i razvoj aplikacija koji će omogućiti ostvarivanje strateških ciljeva definisanih projektom.

Standardi i preporuke za dizajn arhitekture informacionih sistema i razvoj aplikacija bazirat će se na nekoliko različitih pogleda i modela arhitektura informacionih sistema.

Strateški pogled definiše strateške ciljeve koje se želi postići definisanjem vodilja i standarda za arhitekturu sistema i razvoj aplikacija.


Procesni pogled definiše način modeliranja poslovnih procesa, kao preduslov izgradnji bilo kojeg informatičkog sistema. Naime, jednostavno pretvaranje postojećih manualnih procesa u elektronske neće polučiti maksimalne efekte i optimalno funkcionisanje sistema, pa je potrebno izvršiti i reinženjering poslovnih procesa.

Kroz funkcionalni pogled na arhitekturu informacionog sistema modelira se izrada specifikacije potrebnih funkcionalnosti sistema te odabir njegove arhitekture.

Podatkovni pogled sadrži preporuke za modeliranje, pohranu i razmjenu podataka u informacionom sistemu.

Integracioni pogled bavi se problemima povezivanja različitih aplikacija i drugih komponenti informatičkog sistema u jedinstvenu cjelinu te načinima sigurnog pristupa i razmjene podataka između informacionih sistema.

Tehnološki pogled definiše modeliranje i razvoj aplikacionih rješenja u različitim scenarijima: od vlastitog razvoja aplikacija, kupovine i implementacije gotovih proizvoda do kombinovanih pristupa.

	PROJEKT IZRADA I USPOSTAVLJANJE OKVIRA INTEROPERABILNOSTI I STANDARDA ZA RAZMJENU PODATAKA	Oznaka: PARCO-IFBIH
	<b>VODILJE I STANDARDI ZA          ARHITEKTURU SUSTAVA I RAZVOJ APLIKACIJA</b>	Page: 7 of 88

Poglavlje „Upravljanje uslugama (eng. service management), sigurnost i drugi nefunkcionalni zahtjevi“ bavi se problemima upravljanja životnim ciklusom informacionih sistema i njihovog održavanja na dnevnom nivou.

Prilog 1. ovog dokumenta je Katalog tehničkih standarda, koji definiše tehničke standarde i preporučena oblast njihove primjene.

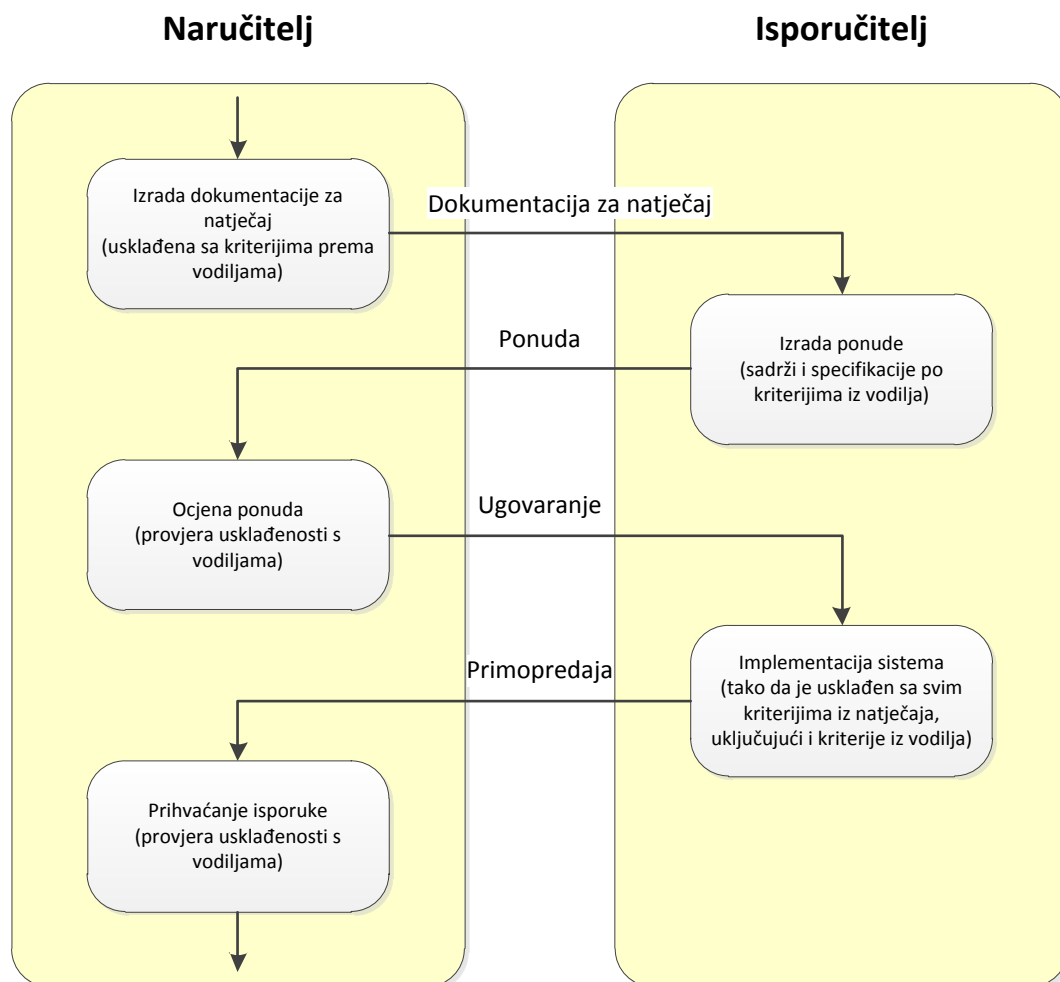
## 1.2 Namjena i korištenje dokumenta

Da bi se osigurala interoperabilnost informatičkih rješenja u državnoj upravi, sva rješenja trebaju slijediti zajednički pristup za sve institucije javne uprave u BiH i biti usklađena sa smjernicama definisanim u ovom dokumentu i povezanim dokumentima iz okvira za interoperabilnost u sljedećim segmentima:

- U pripremi projekata davanjem preporuka i zahtjeva na projekte primjene ICT, davanjem preporuka i zahtjeva na ukupne tehnološke platforme institucija, davanjem preporuka i zahtjeva na projekte uspostave i razvoja pojedinačnih informacionih sistema te zahtjeva na profil zaposlenika za provedbu i implementaciju ICT projekata.
- U javnoj nabavci u vezi sa ICT projektima davanjem metodologije pripreme tenderske dokumentacije, posebno tehničkih specifikacija kod razvoja novih informacionih sistema, tehničkih specifikacija gotovih proizvoda i rješenja, uključujući zahtjeve na servise za razmjenu podataka s vanjskim sistemima, te specifikacije za održavanje i/ili nadogradnju informacionih sistema i rješenja.
- U implementaciji projekata davanjem pregleda preporučenih metodologija i alata za upravljanje projektima, te metodologija i alata za testiranje i prihvatanje rješenja.

U svrhu usklađivanja izlaznih podataka s aktuelnim evropskim inicijativama, konsultovat će se informacije i rezultati odgovarajućih evropskih projekata (npr. OSOR.EU, PEPOL...)

Proces nabave rješenja i usluga na osnovu ovog dokumenta prikazan je na slijedećoj slici.




Slika 1 Korištenje vodilja<sup>1</sup>

Koraci su:

1. Kreiranje konkursne dokumentacije - prilikom izrade konkursne dokumentacije u nju se uključuju i odgovarajući kriteriji iz ovih vodilja
2. Izrada ponude - ponuda uključuje i odgovore za ispunjenost kriterija prema zahtjevu naručitelja
3. Ocjena ponuda - pristigle ponude se usporuđuju, također i prema kriterijima iz vodilja
4. Nakon odabira dobavljača, sklapa se ugovor s dobavljačem i počinje implementacija
5. Nakon obavljene implementacije obavlja se prihvatanje sistema, kako bi se provjerilo da je isporučen sistem u skladu s zahtjevanim funkcionalnostima i kriterijima iz konkursne dokumentacije, što uključuje i sve kriterije iz vodilja

<sup>1</sup> Prema SAGA 4.0, str. 27, Slika 2-4, <http://www.cio.bund.de/saga>



	PROJEKT IZRADA I USPOSTAVLJANJE OKVIRA INTEROPERABILNOSTI I STANDARDA ZA RAZMJENU PODATAKA	Oznaka: PARCO-IFBIH
	<b>VODILJE I STANDARDI ZA          ARHITEKTURU SUSTAVA I RAZVOJ APLIKACIJA</b>	Page: 9 of 88

### 1.3 Povezani dokumenti

U skladu s početnim izvještajem izlazne podatke po ostalim komponentama čini sljedeća dokumentacija:

Komponenta 1. Razvijanje okvira interoperabilnosti za BiH, a koja uključuje:


1. Najbolje prakse, iskustva i trendove interoperabilnosti u BiH i drugim zemljama.
2. Vodilje okvira interoperabilnosti koji sadrže koncept, pravila i standarde.
3. Nacrt Odluke o usvajanju Okvira interoperabilnosti za BiH s pratećim setom dokumenata
4. Nacrt Okvira interoperabilnosti za BiH

Komponenta 2. Stvaranje zajedničkih otvorenih standarda za razmjenu podataka i definicije meta-podataka

1. Lista meta-rječnika podataka, te definicija XML shema/standarda koji ih opisuju
2. Standardi, vodilje i politike za razvoj i održavanje rječnika podataka
3. Inicijalni rječnik podataka u formi XML shema, koji se treba objaviti kao otvoreni standard
4. Nacrt Uputstva za primjenu zajedničkih otvorenih standarda za razmjenu podataka i definicije meta-podataka


Komponenta 4. Razvijanje strategije javnih registara

1. Analiza postojećih parcijalnih elektronskih registara
2. Strateški prijedlozi za optimalnu mobilizaciju i međusobno povezivanje elektronskih registara
3. Lista prioriteta za implementaciju javnih registara
4. Nacrt Strategije javnih registara u BiH
5. Nacrt Uputstva za implementaciju Strategije javnih registara


	PROJEKT IZRADA I USPOSTAVLJANJE OKVIRA INTEROPERABILNOSTI I STANDARDA ZA RAZMJENU PODATAKA  <b>VODILJE I STANDARDI ZA          ARHITEKTURU SUSTAVA I RAZVOJ APLIKACIJA</b>	Oznaka: PARCO-IFBIH
		Page: 10 of 88

## 1.4 Definicije i skraćenice

<b>Pojam</b>	<b>Opis</b>
ABIE	Aggregate Business Information Entity
ACC	Aggregation Core Component
ASBIE	Association Business Information Entity
BBIE	Basic Business Information Entity
BCC	Basic Core Component
BD BiH	Brčko distrikt BiH
BDT	Business Data Type
BIE	Basic Information Entity
BiH	Bosna i Hercegovina
BPM	Business Process Management
CC	Core Components
CCL	Core Components Library
CCT	Core Component Type
CCTS	Core Components Technical Specification
CEN	Comite Européen de Normalisation
CEN CWA	CEN Workshop Agreement
CEN/ISSS	Comite Européen de Normalisation/Information Society Standardisation System
CENELEC	European Committee for Electro technical Standardization
DEN	Dictionary Entry Name
ebCCTS	ebXML Core Components Technical Specification
ebCPP	ebXML Collaboration Protocol Profile and Agreement
ebMS	ebXML Messaging Services
ebRIM	ebXML Registry Information Model
ebXML	electronic business XML
EDI	Electronic Data Interchange
EDIAKT	Austrioni stadard za komunikaciju bez upotrebe papira
EDIFACT	EDI for Administration, Commerce and Transport
eGIF	eGovernment Interoperability Framework
EIF	Evropean Interoperability Framework


	PROJEKT IZRADA I USPOSTAVLJANJE OKVIRA INTEROPERABILNOSTI I STANDARDA ZA RAZMJENU PODATAKA  <b>VODILJE I STANDARDI ZA          ARHITEKTURU SUSTAVA I RAZVOJ APLIKACIJA</b>	Oznaka: PARCO-IFBIH
		Page: 11 of 88

ESB	Enterprise Service Bus
EU	Evropska unija
FBIH	Federacija BiH
GPM	Government Performance Management
GSB	upravna komunikaciona sabirnica (eng. Government Service Bus)
IDDEEA	Agencija za identifikacione dokumente, evidenciju i razmjenu podataka
IDE	Integrirana razvojna okolina (eng. Integrated Development Environment)
IKT	Informaciono komunikaciona tehnologija
ISO	Međunarodna organizacija za standarde (International Organization for Standardization)
ISO	International Organisation for Standardization
ISO/IEC	International Organisation for Standardization/International Electrotechnical Commission
MIG	Message Implementation Guide
NDR	Naming and Design Rules
OIO	Offentlig Information Online
OIOXML	Offentlig Information Online XML
RS	Republika Srpska
SAGA	Standards and Architectures for e-Government Applications
SEMIC.EU	Semantic Interoperability Centre Evrope
SOA	servisno orijentisana arhitektura (eng. Service Oriented Architecture)
UBL	Universal Business Language
UML	Unified Modelling Language
UMM	UN/CEFACT's Modelling Methodology
UN/CEFACT	UN Centre for Trade Facilitation and Electronic Business
VM BiH	Vijeće ministara BiH
XML	EXtensible Markup Language
XSD	XML Schema Definition

	PROJEKT IZRADA I USPOSTAVLJANJE OKVIRA INTEROPERABILNOSTI I STANDARDA ZA RAZMJENU PODATAKA  <b>VODILJE I STANDARDI ZA          ARHITEKTURU SUSTAVA I RAZVOJ APLIKACIJA</b>	Oznaka: PARCO-IFBIH
		Page: 12 of 88

## 1.5 Popis slika, tablica i priloga

Slika 1 Korištenje vodilja.....	8
Slika 2 Generička arhitektura elektronske javne uprave određenog upravnog nivoa .....	14
Slika 3 Proces upravljanja arhitekturom.....	16
Slika 4 Segmentacija Porterovog lanca vrijednosti prilagođenog djelovanju državne uprave	20
Slika 5 Troslojni model arhitekture usluga .....	25
Slika 6 Principi semantičkog weba.....	34
Slika 7 UN/CEFACT CC i BIE - veze .....	37
Slika 8 Integracijska arhitektura IT sistema.....	40
Slika 9 Funkcije servisnih sabirnica javne uprave .....	43
Slika 10 SOA referentni model .....	58
Slika 11 Elementi usluge .....	59
Slika 12. Faze metode prilagodbe SOA referentnog modela .....	66

	PROJEKT IZRADA I USPOSTAVLJANJE OKVIRA INTEROPERABILNOSTI I STANDARDA ZA RAZMJENU PODATAKA  <b>VODILJE I STANDARDI ZA          ARHITEKTURU SUSTAVA I RAZVOJ APLIKACIJA</b>	Oznaka: PARCO-IFBIH
		Page: 13 of 88

## 2 Strateški pogled

Kroz strateški pogled sagledavaju se vizija, misija i strategija informatizacije u javnoj upravi i institucijama javne uprave u Bosni i Hercegovini.

### 2.1 Elektronske usluge i javni registri za potrebe građana, privrede i društva

Prilikom dizajniranja arhitekture IT sistema državne uprave potrebno je osigurati da sistem zadovoljava interese svih potencijalnih zainteresovanih strana: građana, kompanija kao i drugih organa državne uprave koji mogu imati potrebe za određenim podacima.

Glavni ciljevi informatizacije javne uprave koje, u skladu sa Strategijom reforme javne uprave, treba ostvariti jesu:

1. Modernizacija poslovanja
2. Povećanje efektivnosti javne uprave,
3. Povećanje kvalitete usluga javne uprave,
4. Smanjenje troškova javne uprave,
5. Optimizacija IKT integracije javne uprave
6. Povećanje transparentnosti poslovanja
7. Povećanje stepena zaštite i povjerenja u podatke koje vodi

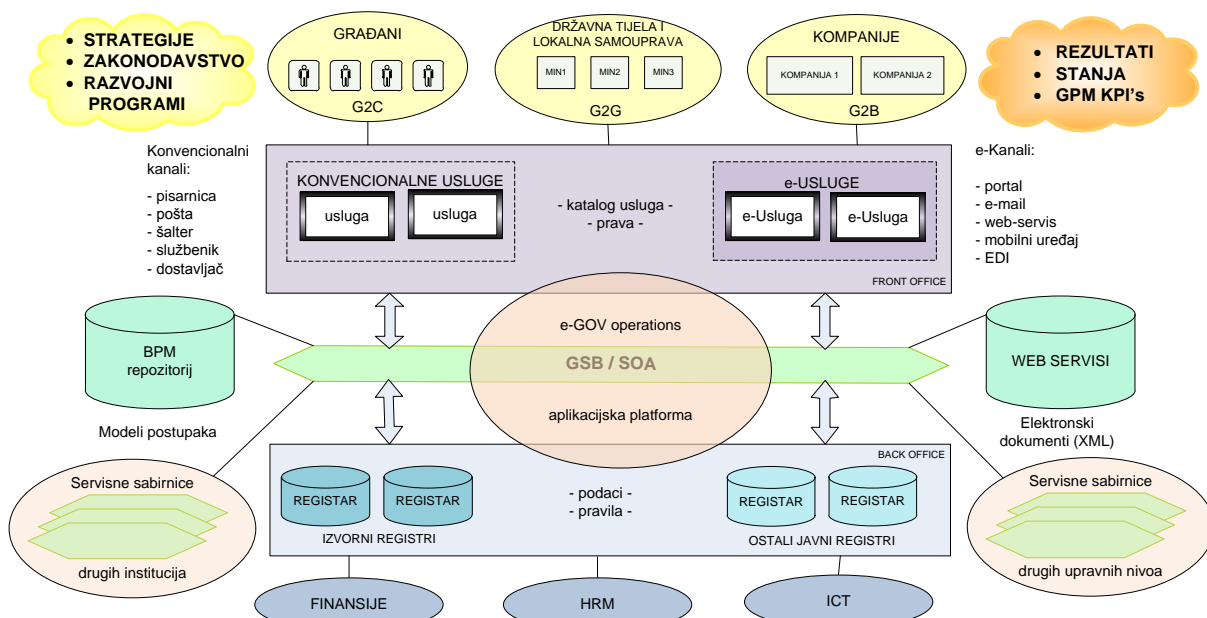
Uspješnost prelaska na elektronsku javnu upravu zasniva se na postupnom prelasku s pružanja usluga konvencionalnim putevima (šalter, pisarnica, pošta...) na elektronsko pružanje usluga (portali, elektronska pošta, razmjena podataka, mobilni uređaji...). Ovaj prelazak ne otklanja potrebu za pružanjem usluge konvencionalnim kanalima (pisarnica, pošta, šalter, službenik, dostavljač...).

Pri tome se mora osigurati da svi korisnici – bez obzira na način pristupa usluzi – uvijek dobiju iste podatke i pod istim uvjetima, a da se podaci uvijek uzimaju sa mjesta koje je odgovorno za njihovo održavanje (kako bi se spriječilo korištenje podataka koji nisu ažurni i izbjegli troškovi njihove dvostruke administracije). Ovo načelo podrazumijeva da javna uprava neće zahtijevati od korisnika (građana i privrednih subjekata) da u postupcima pribavljaju uvjerenja o činjenicama o kojima institucija javne uprave vode registre ili druge službene evidencije. Takva uvjerenja institucije pribavljaju po službenoj dužnosti (u skladu s čl.42. Zakona u upravi BiH, SGBiH 32/02, 102/09; čl. 50 Zakona o upravi u Federaciji Bosne i Hercegovine, Službene novine Federacije BiH, broj 28/97; čl. 75a. Zakona o državnoj upravi Sl. glasnik RS, br. 79/2005, 101/2007 i 95/2010; čl. 52. Zakona o javnoj upravi Brčko Distrikta Službeni glasnik BDBiH 19/07, 2/08 43/08). Vijeće ministara BiH, vlade entiteta i Vlada BD donose poseban akcioni plan za implementaciju navedenog načela i predviđene mjere provode u djelo.

Korisnički usmjerena javna uprava kao pružalac usluga treba težiti ispunjenju očekivanja građana i privrednih subjekata:

- Omogućiti pristup personaliziranim uslugama koje su jednostavne, sigurne i fleksibilne za korištenje.
- Omogućiti dostavu konvencionalnim i elektronskim kanalima (dostava bilo kako, bilo gdje, bilo kada).
- Omogućiti pristup putem jedinstvenih pristupnih točaka (i u slučajevima kada više institucija sudjeluje u pružanje usluge).
- Preuzimati od korisnika samo informacije nužne za pružanje usluge.
- Preuzimati istu informaciju od korisnika samo jedanput.
- Štititi privatnost korisnika.
- Snižavanje troškova građana i privrednih subjekata u komunikaciji s javnom upravom.

Na strateškom nivou, generička arhitektura IT sistema državne uprave (kao i svakog pojedinog organa državne uprave) prikazani su na slici 2.




**Slika 2 Generička arhitektura elektronske javne uprave određenog upravnog nivoa**

Osnova za definisanje arhitekture bilo kojeg sistema su identifikacija usluga koje sistem mora pružiti i korisnika tih usluga.

## 2.2 Upravljanje efektom u javnoj upravi (eng. Government Performance management)

Usluge koje državna uprava pruža definisane su strategijama, zakonodavnim okvirom, (zakonima i podzakonskim aktima) te razvojnim programima.

Korisnici usluga su u pravilu građani, privreda (kompanije) te druga organa javne uprave i lokalne samouprave.

	PROJEKT IZRADA I USPOSTAVLJANJE OKVIRA INTEROPERABILNOSTI I STANDARDA ZA RAZMJENU PODATAKA  <b>VODILJE I STANDARDI ZA          ARHITEKTURU SUSTAVA I RAZVOJ APLIKACIJA</b>	Oznaka: PARCO-IFBIH
		Page: 15 of 88

Arhitektura mora definisati, kanale kojima se usluga pruža korisnicima usluga konvencionalnim i elektronskim putem te način povezivanja usluge sa podacima (izvornim i izvedenim registrima).

Iako pojedini podaci i aplikacije mogu biti implementirani u heterogenim tehnologijama, važno je da na nivou organa kao i na nivou centralne države postoji zajednička integraciona platforma koja omogućava integraciju svih poslovnih procesa i podataka državne uprave u jedinstvenu cjelinu.

Da bi se mogla pratiti efektivnost i uspješnost provođenja strateških ciljeva, potrebno je definisati metodologije praćenja rezultata i stanja sistema te definisati Ključne Pokazatelje Uspješnosti (KPI), koji omogućavaju mjerenje rezultata.

Mjerenje uspješnosti realizuje se kroz harmonizirane indikatore i usporedive prema drugim državama (EU) u skladu sa aktuelnim radnim okvirom Evropske unije za mjerenje informacionog društva („*Benchmarking Digital Europe 2011-2015 a conceptual framework*“ odobrenog od strane *i2010 High level group* Evropske komisije u studenom 2009. godine). Ovaj dokument jest konceptualni okvir za prikupljanje statističkih podataka o informacionom društvu, kao i popis ključnih pokazatelja koji će se koristiti za vrednovanje Evropske Digitalne agende.

Sa stajališta sadržaja mjerenja, upotrebljenih metrika i komparacionih referentnih modela, česta je upotreba „benchmarking“ metode, odnosno modela, te se preporučava koristiti modele Statističkog ureda Evropske unije, (Statistical Office of the European Communities – EVROSTAT), odnosno modele drugih općih uprava Evropske komisije (Directorate General – DG) <sup>2</sup>. Na taj način se dobiva ne samo usporedivost rezultata s drugim zemljama nego i korištenje dobrih praksi i znanja iz drugih država radi vlastitog razvoja!


U okviru „benchmarking“ nalazi se i praćenje razvoja elektronskih usluga javne uprave (e-Government usluga) prema ključnim pokazateljima uspješnosti (eng. *Key Performance Indicators – KPI*) iz predmetnih „benchmarking“ modela.

Kao što je navedeno u Okviru, Vlade upravnog nivoa će odrediti institucije nadležne za sva pitanja interoperabilnosti svog upravnog nivoa, pa tako i praćenje rezultata primjene standarda za interoperabilnost i provođenje mjerenja prema aktuelnom radnom okviru Evropske unije. Nadležne institucije pružat će organizacionu podršku institucijama tog upravnog nivoa u primjeni Okvira, te administrativnu, stručnu i tehničku pomoć radu Koordinacije za interoperabilnost tog upravnog nivoa.

### 2.3 Interoperabilnost kao omogućitelj efikasne nove elektronske uprave

Efikasna elektronska uprava zasniva se na elektronskim uslugama, koje javna uprava pruža građanima, privrednim subjektima i drugim javnim upravama, a koji se zasnivaju na informacionim sistemima izgrađenima na osnovu standardizovanih arhitektura i razmjeni podatak u standardizovanim formatima

<sup>2</sup> Na primjer „*Benchmarking Digital Europe 2011-2015 a conceptual framework*“, Evropska komisija, DG Information Society and Media

	PROJEKT IZRADA I USPOSTAVLJANJE OKVIRA INTEROPERABILNOSTI I STANDARDA ZA RAZMJENU PODATAKA <b>VODILJE I STANDARDI ZA          ARHITEKTURU SUSTAVA I RAZVOJ APLIKACIJA</b>	Oznaka: PARCO-IFBIH
		Page: 16 of 88

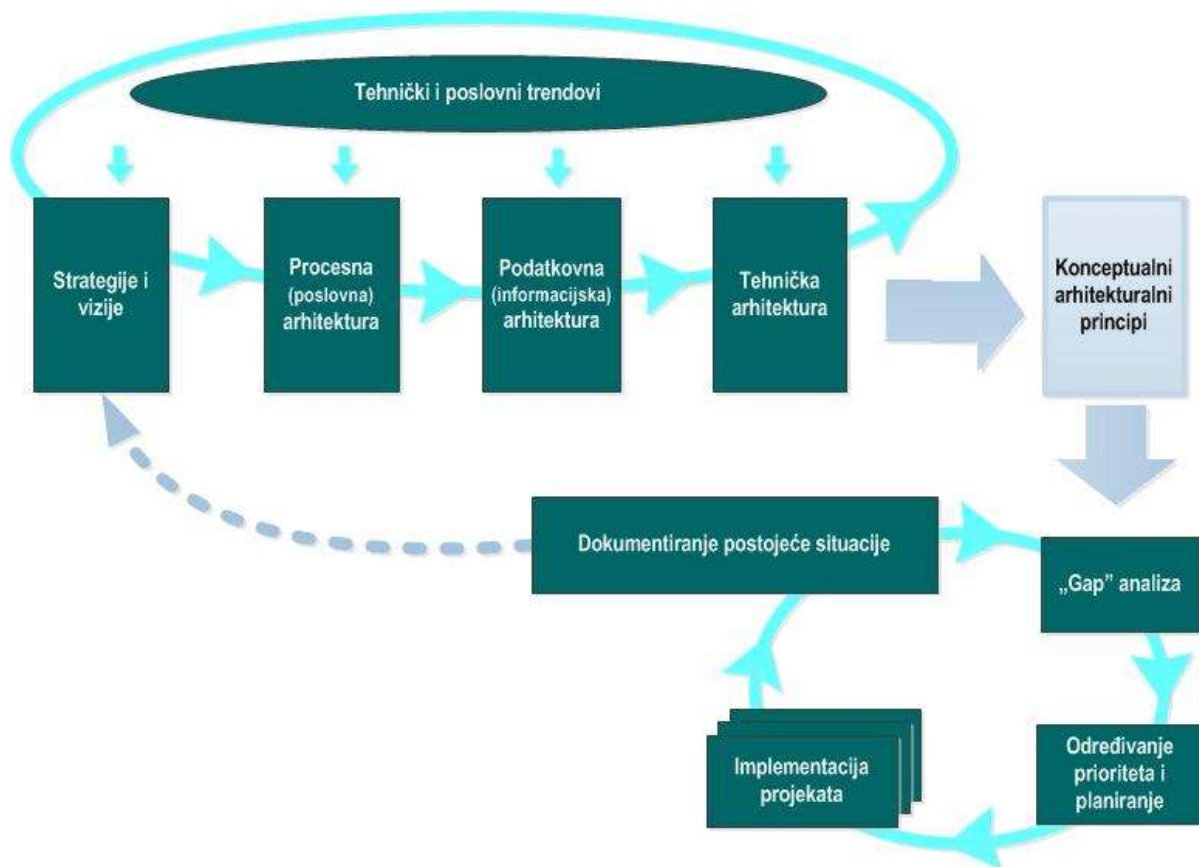
Prilikom modeliranja i uspostave informacionih sistema potrebno je koristiti/biti usklađen sa konceptima sadržanima u ovom dokumentu te referenciranim dokumentima. U oblastima u kojima nije preporučena standard, institucije će primjenjivati sektorske standarde i specifikacije.

Arhitekture informacionih sistema možemo promatrati na nekoliko nivoa:

1. ukupna arhitektura na nivou BiH (povezivanja sistema različitih institucija)
2. ukupna arhitektura organizacije

Strateški gledano, novi informacioni sistem je potrebno pozicionirati u odnosu na oba navedena nivoa.


Rad na kontinuisanom usavršavanju arhitekture elektronske javne uprave posljedica je strateškog opredjeljenja i vizije elektronske javne uprave, a obuhvaća praćenje tehnoloških, poslovnih i drugih trendova, periodične procjene trenutne situacije, planiranje promjene sistema u cilju usavršavanja te implementaciju odgovarajućih projekata. Ovaj proces upravljanja arhitekturom ilustruje sljedeća slika.



Slika 3 Proces upravljanja arhitekturom<sup>3</sup>

<sup>3</sup> Izvor: White Paper on Enterprise Architecture, Ministry of Science, Technology and Innovation, Danska, 2003



	PROJEKT IZRADA I USPOSTAVLJANJE OKVIRA INTEROPERABILNOSTI I STANDARDA ZA RAZMJENU PODATAKA	Oznaka: PARCO-IFBIH
	<b>VODILJE I STANDARDI ZA          ARHITEKTURU SUSTAVA I RAZVOJ APLIKACIJA</b>	Page: 17 of 88

Praćenje aktuelnih tehnoloških i poslovnih trendova u oblasti elektronske javne uprave rezultira potrebom za repozicioniranjem smjera i vizije razvoja ukupne arhitekture, ali i razvojem poslovne, informacione i tehnološke arhitekture. Pri čemu poslovna arhitektura definiše prioritete poslovne procese koje IKT podržava u smislu funkcionalnih i operativnih svojstava. Informaciona infrastruktura definiše organizacijske i informacione (uključujući podatkovne) zahtjeve usklađene s poslovnim zahtjevima. Tehnička arhitektura obuhvaća zahtjeve na tehnička rješenja koja proizlaze iz poslovnih promjena te adresira pozicioniranje pojedinog sistema, te njegovih funkcija i modula, u ukupnoj arhitekturi. Pri tome je nužno poštovati usaglašene principe na kojima se zasniva koncept same arhitekture.

Usaglašeni principi na kojima počiva koncept arhitektura direktno utječe na proces praktične implementacije koji uključuje:

- dokumentiranje postojeće situacije kao polazne tačke za planiranje i kontinuisan operativni rad,
- izradu „gap“analize koja opisuje kolika su odstupanja postojećih rješenja, metoda i organizacije od principa danih konceptom arhitekture
- određivanje prioriteta i planiranje kojim se definišu potrebe za migracijama sa svrhom usklađivanja sa poslovnim ciljevima i principima danima konceptom arhitekture
- implementaciju projekata koji su koordinirani i usmjereni istom općem cilju kroz programsko upravljanje portfeljem projekata.

Prilikom upravljanja informacionim sistemima potrebno je poštovati slijedeće strateške elemente:


- sve strateške pretpostavke navedene u Okviru interoperabilnosti,
- načelo korištenja zajedničkih infrastrukture i resursa interoperabilnosti (rječnika, repozitorija, PKI...),
- načelo ponovnog korištenje postojećih resursa,
- načelo tehnološke neutralnosti i prilagodljivosti korisnicima,
- zaštitu privatnost građana i povjerljivosti informacija privrednih subjekata,
- usklađenosti s pravnim okvirom.

## 2.4 EU perspektiva i facilitatori interoperabilnosti

Uspostava interoperabilnosti zasniva se na načelima korisnički usmjerene uprave u skladu s društvenim i korisničkim potrebama i prioritetima te u odnosu na zahtjeve interoperabilnosti EU.


U skladu sa Strategijom reforme javne uprave, strateška je odrednica BiH da javna uprava mora biti funkcionalna, pouzdana, efikasna, odgovorna, transparentna i usklađena kako bi se kao jedna moderna, korisnicima upućena, javna uprava mogla priključiti "jedinostvenom evropskom upravnom prostoru". Trenutni evropski trend je da su međusobna povezanost IT sistema javnog sektora, razmjena podataka i pružanje usluga zasnovani na interoperabilnosti, radije nego na integraciji IT sistema.

Postojanje zajedničke infrastrukture jedan je od preduslova uspješne implementacije interoperabilnosti. Zato će procesi razvoja i primjene interoperabilnosti biti podržani

	PROJEKT IZRADA I USPOSTAVLJANJE OKVIRA INTEROPERABILNOSTI I STANDARDA ZA RAZMJENU PODATAKA	Oznaka: PARCO-IFBIH
	<b>VODILJE I STANDARDI ZA ARHITEKTURU SUSTAVA I RAZVOJ APLIKACIJA</b>	Page: 18 of 88

informaciono-komunikacionim tehnologijama pomoću zajedničkih resursa i servisa podrške kao facilitatora interoperabilnosti koji su na raspolaganju svim interesnim stranama.

Organi zaduženi za strateško upravljanje arhitekturom informacionih sistema na svojim upravnim nivoima i njihova koordinacija prate trendove razvoje interoperabilnosti u EU i predlažu promjene koje vode ka harmonizaciji i standardizaciji ove oblasti u skladu sa EU praksom.

	PROJEKT IZRADA I USPOSTAVLJANJE OKVIRA INTEROPERABILNOSTI I STANDARDA ZA RAZMJENU PODATAKA	Oznaka: PARCO-IFBIH
	<b>VODILJE I STANDARDI ZA          ARHITEKTURU SUSTAVA I RAZVOJ APLIKACIJA</b>	Page: 19 of 88

### 3 Procesni pogled

Prilikom modeliranja informacionih sistema koji se uspostavljaju potrebno je napraviti modeliranje poslovnih procesa iz poslovnih oblasti u kojima se primjenjuje informacioni sistem. Prilikom toga se koriste prakse iz modeliranja i upravljanja poslovnim procesima, te procesima upravljanja informacionim sistemima.

Razumijevanje poslovnih procesa pretpostavka je za izradu funkcionalne arhitekture sistema. Analiza i kontinuisano provođenje upravljanja poslovnim procesima omogućava nam:

- identifikaciju relevantnih poslovnih procesa
- kreiranje i verifikaciju modela tih procesa
- optimizaciju poslovnih procesa
- implementaciju poslovnih procesa
- nadzor poslovnih procesa
- provođenje kontinuisanog usavršavanja poslovnih procesa

#### 3.1 Upravljanje poslovnim procesima

Upravljanje poslovnim procesima je jedan od načina usklađivanja poslovanja s ciljevima i strategijom organizacije. Zadatak je što efikasnije prilagoditi poslovanje novim utjecajima, kako unutrašnjim (nova strategija), tako i vanjskim (npr. promjene u zakonskim regulativama, tehnologijama).


Konačni proizvod usklađivanja je centralni repozitorij poslovnih procesa dostupan različitim nivoima managementsa i integrisan s drugim repozitorijima. Takav repozitorij ima višestruku vrijednost jer predstavlja bazu znanja i dokumentira smjernice, uspostavljene prakse i procedure.

Kod definisanja poslovnih procesa moraju se obraditi sljedeće karakteristike:

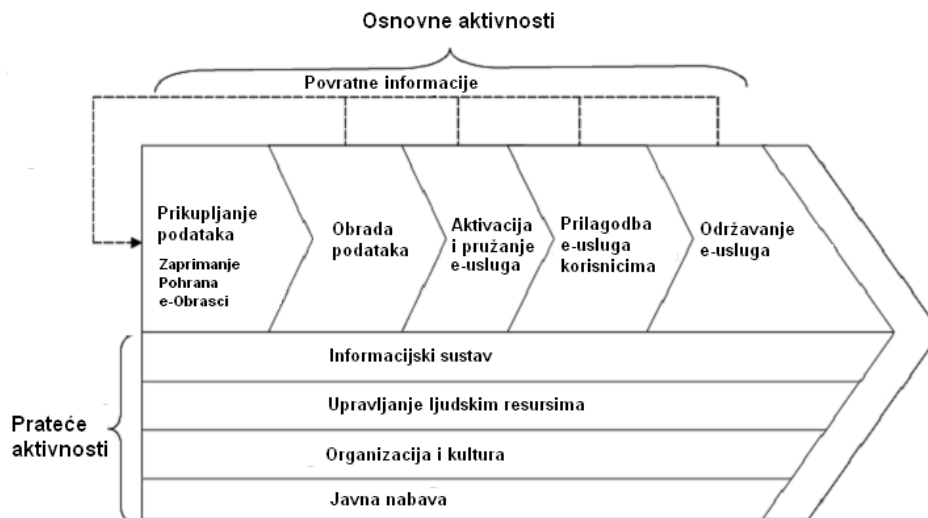
- Ciljevi i stepen usklađenosti s regulativama
- Analiza postojećih procesa („as-is“)
- Model novog stanja („to-be“)
- Pravila izvršavanja
- Struktura organizacije i resursi
- Tehnologija
- Integracije s drugim procesima i sistemima
- Indikatori performansi
- Odgovorna osoba (vlasnik procesa)

#### 3.2 Modeliranje poslovnih procesa i procesna arhitektura

Generička procesna arhitektura uključuje sve poslovne procese institucije javne uprave (upravljачke, izvršne i informacione). Oni se sastoje od niza koraka koje je potrebno ostvariti da bi građanin ili privredni subjekt dobio od države određenu uslugu ili ostvario neko pravo,

	PROJEKT IZRADA I USPOSTAVLJANJE OKVIRA INTEROPERABILNOSTI I STANDARDA ZA RAZMJENU PODATAKA  <b>VODILJE I STANDARDI ZA          ARHITEKTURU SUSTAVA I RAZVOJ APLIKACIJA</b>	Oznaka: PARCO-IFBIH
		Page: 20 of 88

odnosno da bi institucija provela neku svoju funkciju čime se povećava vrijednost usluga javne uprave građanima i privredi. Pri analizi vlastite sposobnosti djelovanja organizacije uobičajeno je korištenje metodologije Porterove analize lanca vrijednosti. Lanac vrijednosti identificira osnovne i prateće procese i aktivnosti koje se odvijaju u institucijama javne uprave, a koji su neophodni za kvalitetnu isporuku usluga građanima i privredi. Donja slika prikazuje segmentaciju Porterovog lanca vrijednosti prilagođenog djelovanju državne uprave kakva je usvojena u okviru eGovernment Economics Project (eGEP) Evropske komisije, a koja je objavljena u Završnoj studiji eGovernment Economics Project (eGEP).




Slika 4 Segmentacija Porterovog lanca vrijednosti prilagođenog djelovanju državne uprave<sup>4</sup>

Osnovne aktivnosti javne uprave jesu prikupljanje podataka (uključujući zaprimanje i pohranu podataka), obrada podataka, aktivacija, pružanje i prilagodba e-usluga korisnicima, te održavanje. Prateće aktivnosti bez kojih ukupno funkcionisanje kao i provođenje osnovnih aktivnosti nije moguće uključuju upravljanje informacionim sistemom, ljudskim resursima, organizacijom i organizacijskom kulturom te provođenje javne nabavke.

Prvi korak u efikasnom upravljanju poslovnim procesima je identifikacija i modeliranje poslovnih procesa organizacije. Modeliranje poslovnih procesa je kreiranje grafičkih modela odnosno dijagrama, kada se pomoću više vrsta objekata definiše proces u obliku procesne mape. Procesne mape koje se koriste u modeliranju su više od jednostavnih dijagrama toka. One su grafičke prezentacije poslovnih procesa. Opisuju korak po korak dijelove zadatka koji trebaju biti izvršeni određenim redoslijedom, kako bi se izvršio jedan proces od inicijalizacije do završetka. Specifikacija obaveznih i preporučenih metodologija i alata za upravljanje poslovnim procesima sadržana je u Prilogu 1.: "Katalog tehnoloških standarda" ovog dokumenta.

<sup>4</sup> Izvor: eGovernment Economics Project (eGEP): Expenditure Study, Final Version. Brussels: eGovernment Unit, DG Information Society, European Commission

	PROJEKT IZRADA I USPOSTAVLJANJE OKVIRA INTEROPERABILNOSTI I STANDARDA ZA RAZMJENU PODATAKA	Oznaka: PARCO-IFBIH
	<b>VODILJE I STANDARDI ZA          ARHITEKTURU SUSTAVA I RAZVOJ APLIKACIJA</b>	Page: 21 of 88


Modeli poslovnih procesa pohranjuju se u bazi podataka (repozitorij), koja garantira konzistenciju modela, ponovnu upotrebu i jednostavno održavanje promjena, kao što je opisano u poglavlju 6.5. Okvira interoperabilnosti BiH.

Poslovni procesi su dinamički model odvijanja poslovanja i kao takvi u skladu s potrebama i situaciji u okolini se mijenjaju, unapređuju. Kako jednom dizajnirani dijagrami poslovnih procesa ne bi predstavljali samo stanje procesa u jednom trenutku vremena, potrebno ih je redovno ažurirati.

Redovito ažuriranje modela poslovnih procesa osigurava kvalitetnu dokumentovanost poslovnih procesa, čime su u svakom trenutku omogućene i dostupne aktuelne informacije o toku poslovnih procesima, odgovornostima i rezultatima poslovnih procesa.

Ovaj dokument opisuje i daje preporuke za arhitekture sistema i razvoj aplikacija kroz pet pogleda:

1. **Pogled proizvoda/ usluga** sadrži sve materijalne i nematerijalne ulaze i izlaze u poslovnom procesu te opisuje proizvode/usluge zbog kojih se poslovni procesi i izvršavaju (njihov su rezultat).
2. Informacioni objekti (entiteti) i njihovi atributi, kao i odnosi između informacionih objekata opisuju se u **podatkovnom pogledu**. S obzirom da događaji predstavljaju status procesa u određenom trenutku, oni se također pridodaju u podatkovnom pogledu. Ovaj pogled sličan je dizajnu modela podataka (npr. ER sheme). Osim klasičnih modela relacionih baza podataka, to mogu biti i XML i druge standardne strukture podataka.
3. Transakcije koje oblikuju efekat i statične veze između njih su prikazani u **funkcijskom pogledu**. Aplikacije i IT sistemi su također uključeni u funkcioni pogled zato što određuju aplikativno podržana procesna pravila za obavljanje aktivnosti. Funkcioni pogled uobičajeno je hijerarhioni, te se razlažu funkcije prema prioritetu i redosljedu izvršavanja u poslovnom procesu.
4. Organizacioni elementi i njihove veze oblikuju organizacionu strukturu i opisuju se u **organizacijskom pogledu**. Kao dodatak ljudskim resursima u ovom pogledu moguće je prikazati i operativne resurse. Ovi će se organizacioni elementi kasnije koristiti kao sudionici (pokretači, kontrolori, korisnici, izvršitelji) određenih procesnih koraka. Njihovo opisivanje na jednom mjestu dat će modelima konzistenciju i jednostavnost kasnijeg održavanja modela.

	PROJEKT IZRADA I USPOSTAVLJANJE OKVIRA INTEROPERABILNOSTI I STANDARDA ZA RAZMJENU PODATAKA  <b>VODILJE I STANDARDI ZA          ARHITEKTURU SUSTAVA I RAZVOJ APLIKACIJA</b>	Oznaka: PARCO-IFBIH
		Page: 22 of 88

Usprkos činjenici da prva četiri pogleda, koji su osnovi statički, imaju utjecaja na procesni tok, ipak ne prikazuju eksplicitno tok procesa. Zato, informacije dobivene iz statičnih pogleda moraju biti ujedinjene u peti dinamični procesni pogled.

- 5. Procesni pogled** prikazuje veze između statičnih objekata u podatkovnom, funkcijskom, proizvodno/ uslužnom i organizacijskom pogledu, stvarajući dinamičan, kronološki tok procesa.


Cjeloviti poslovni procesi (od početka do kraja ili „end-to-end“) definišu se po novoima, za koje su korištena dvije osnovne vrste dijagrama: Value Added Chain model i Event-driven Process Chain modela, kojim se općenitija nivo informacija raščlanjuje u detaljniji prikaz.

Informacije na **Value Added Chain** nivou opisuju generalne procesne korake (pod-procese), organizacionu odgovornost za pod-procese i mogu biti dodatno opisani informacijama poput informatičkih sistema koji ih podržavaju, identifikovanih potencijalnih rizika ili podataka koji se upotrebljavaju ili nastaju u procesu. Navedeni modeli biti će ažurirani ukoliko je glavni poslovni proces značajno izmijenjen, odnosno promijenjen je tok glavnih podprocesa. Ovakav prikaz važan je jer jasno pokazuje i lanac nastajanja vrijednosti u poslovnim procesima (kako mu i ime govori).

Svaki pod-proces dodatno se raščlanjuje kako bi se detaljnije opisao redoslijed procesnih koraka, organizacionih jedinica koje ih izvršavaju i njihovih rezultata koristeći **Event-driven Process Chain (EPC)**. Budući da EPC modeli sadrže mnoštvo detalja o procesima, na ovom nivou izvršiti će se većina promjena nad samim modelima poslovnih procesa. Ovakav opis posebno je značajan u poslovnim procesima koji su intenzivno događajima pogonjeni (event-driven). Kasnije se ovakav model nadovezuje na tehničku realizaciju koja je događajno-orijentisana i projektovana (EDA), čime su aplikacioni sistemi značajno sličniji odvijanju u stvarnosti, te su tako razumljiviji i bliži krajnjim korisnicima.

Primjena predloženog modela za ažuriranje poslovnih procesa omogućit će prije svega standardizaciju, a zatim i postizanje najvažnijih efekata, a to je uspostava aktuelnog i realnog poslovnog modela za povećanje efektivnosti te racionalno korištenje resursa sudionika poslovnih procesa, što uključuje slijedeću procesnu i informacionu infrastrukturu:

- Jednoznačno definisanje događaja – pokretača procesa,
- Definisane procesa,
- Definisane aktivnosti subjekata i njihovog slijeda,
- Definisane međudnosa subjekata i aktivacija djelovanja pojedinih subjekata Koordinacije,
- Definisane resursa i uspostava parametara za njihovu klasifikaciju po vrstama,
- Definisane parametara – elemenata za praćenje efektivnosti procesa,
- Definisane čimbenika koji su bitni da bi se stečeno znanje moglo koristiti za unapređivanje sistema.

	<p>PROJEKT IZRADA I USPOSTAVLJANJE OKVIRA INTEROPERABILNOSTI I STANDARDA ZA RAZMJENU PODATAKA</p>	<p>Oznaka: PARCO-IFBIH</p>
	<p>VODILJE I STANDARDI ZA ARHITEKTURU SUSTAVA I RAZVOJ APLIKACIJA</p>	<p>Page: 23 of 88</p>

### 3.3 Metodologija izrade modela poslovnih procesa

**Analiza propisa** – analiza zakona, upravnih i pravnih akata kao skupa pravila u poslovnim procesima.

**Anketiranje i razgovor sa sudionicima poslovanja** – dobavljanje informacija iz prve ruke i uvid u realnu praksu i konkretne probleme.


**Prikupljanje funkcionalnih podataka** – opisi zadataka, analiza tipova dokumenata, načina komunikacije (protokoli), formata poruka, utvrđivanje odgovornosti i prava pristupa, baza pravila.

**Prikupljanje nefunkcionalnih podataka** – popisivanje sudionika i resursa, analiza postojeće tehnologije, tipična vremena izvršavanja zadatka

**Modeliranje i simulacije** – izrada modela procesa na svim nivoima i definisanje podataka za simulacije.

**Monitoring** – prikupljanje statistike izvršavanja

Informacione tehnologije koje se koriste u ovim metodama moraju poštivati općenite odredbe i standarde vezane uz interoperabilnost i otvorenost specificirane u Prilogu 1 ovoga dokumenta.

	<p>PROJEKT IZRADA I USPOSTAVLJANJE OKVIRA INTEROPERABILNOSTI I STANDARDA ZA RAZMJENU PODATAKA</p>	<p>Oznaka: PARCO-IFBIH</p>
	<p><b>VODILJE I STANDARDI ZA ARHITEKTURU SUSTAVA I RAZVOJ APLIKACIJA</b></p>	<p>Page: 24 of 88</p>

## 4 Funkcionalni pogled

Kroz funkcionalni pogled na arhitekturu informacionog sistema modelira se rad potrebnih funkcionalnosti sistema kroz korake kako slijedi.

Definisanje funkcionalnosti sistema mora obuhvatiti razrađene korisničke zahtjeve specifikaciju funkcionalnosti i arhitekturu sistema.

**Korisničke zahtjeve** je potrebno detaljno analizirati i uskladiti s realnim stanjem sistema i obimom projekta. Analizom se mogu utvrditi kontradikcije u zahtjevima ili dogovorenom obimu, nepotpunosti ili nerelana očekivanja s obzirom na stanje sistema. Ponekad je potrebno i opisati period uvođenja novog sistema, edukaciju korisnika i sl.

**Funkcionalna specifikacija** mora definisati sve funkcije i pravila sistema koje Korisnik traži. Radi toga nije potrebno ulaziti u tehničke detalje, već samo naglasiti sve tačke koje utječu na poslovanje, a to su:

- Arhitektura
- Poslovna pravila
- Podaci (strukture, poruke, dokumenti)
- Sigurnosni mehanizmi (uloge, grupe, protokoli)
- Sučelje (GUI)
- Infrastruktura

### 4.1 Funkcionalna arhitektura

Važna stavka prilikom dizajna/redizajna sistema je odabir odgovarajuće arhitekture informacionog sistema.

Najčešće korištene arhitekture uključuju:

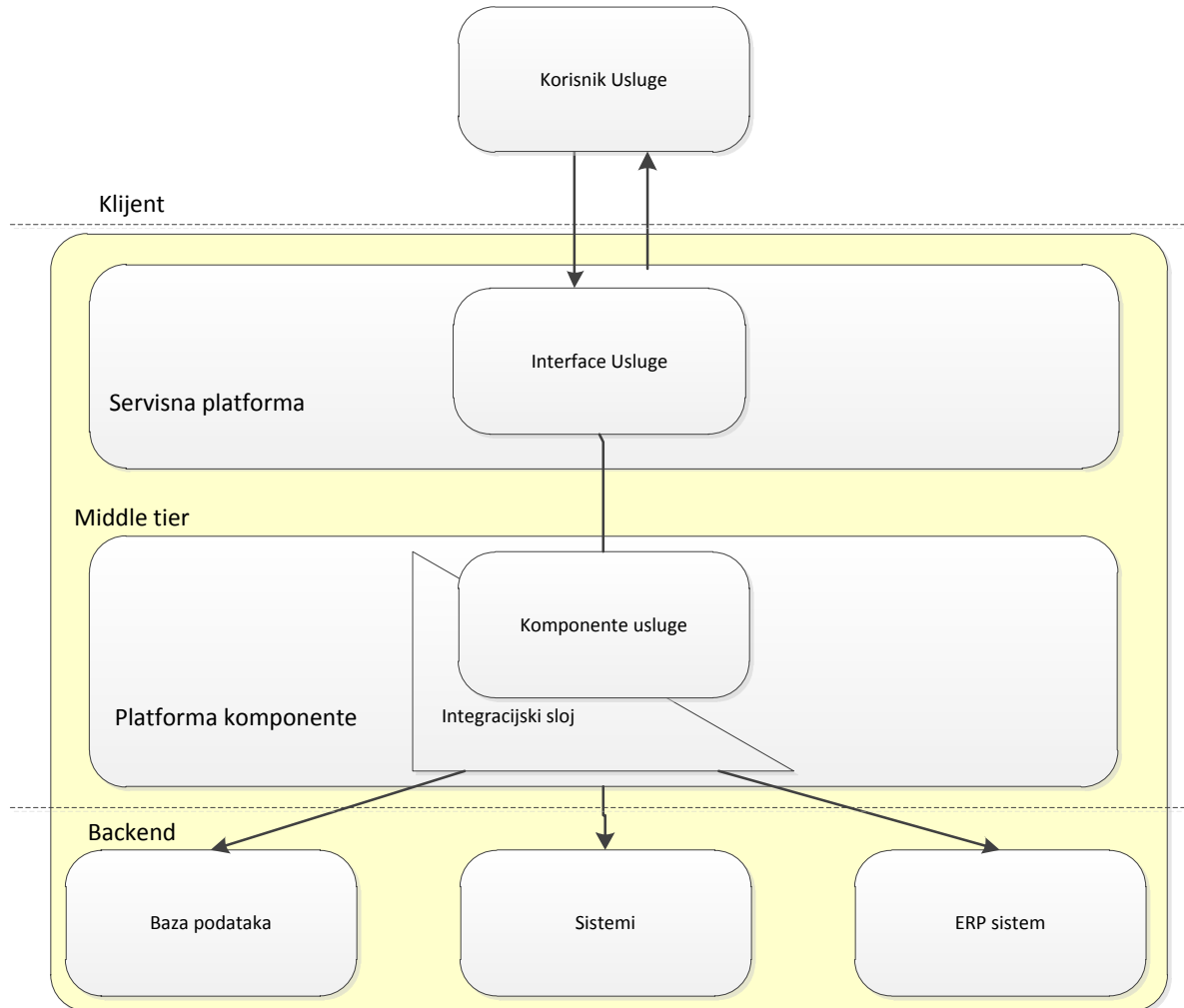
- višeslojna
- komponentna
- SOA

Odabir arhitekture ovisi o zahtjevima i obimu projekta. Na primjer, nekada servisna arhitektura ne odgovara zahtjevima odziva sistema, a nekada komponentna aritektura donosi premali nivo integracije.

Vrlo često konačna arhitektura predstavlja kombinaciju više drugih arhitekture. Međutim, prednost se daje servisnoj arhitekturi zbog njeze interoperabilnosti i fleksibilnosti.



#### 4.1.1 Višeslojna arhitektura




Slika 5 Troslojni model arhitekture usluga<sup>5</sup>

Višeslojna arhitektura se ne koncentrira na funkcionalne cjeline, već na logičke cjeline (slojeve) vezane za funkcionalne zahtjeve (prikaz, pravila, spremanje podatka). Za svaki taj sloj se definiše apstraktna funkcionalnost i sučelje kojim će slojevi razmjenjivati poruke.

Na ovaj način se razdvajaju pravila funkcioniranja svakog sloja pa je i razvoj jednostavniji, a moguće je i lakše zamijeniti cijeli sloj ako oba podržavaju isto sučelje. Problemi nastaju jedino kod izmjene sučelja između slojeva pa se na to treba obratiti najviše pažnje.

Tipična višeslojna arhitektura je troslojna arhitektura, tzv. MVC (Model-View-Controller) koja razdvaja korisničko sučelje (View) od poslovne logike (Controller) i modela podataka (Model).

<sup>5</sup> Prema SAGA 4.0, str. 74, Slika 6.4, <http://www.cio.bund.de/saga>

	PROJEKT IZRADA I USPOSTAVLJANJE OKVIRA INTEROPERABILNOSTI I STANDARDA ZA RAZMJENU PODATAKA  <b>VODILJE I STANDARDI ZA          ARHITEKTURU SUSTAVA I RAZVOJ APLIKACIJA</b>	Oznaka: PARCO-IFBIH
		Page: 26 of 88

Moguće je koristiti i dodatne slojeve za npr. apstrakciju pristupa.podacima (JDBC) ili sigurnosne mehanizme (SSO).

Prednosti ovakve arhitekture su:

- Manja ovisnosti o postojećim tehnologijama i platformama
- Brži razvoj zbog modularnosti
- Prilagođavanje korisniku (prikaz na različitim uređajima, integracija s drugim aplikacijama)

#### 4.1.2 Komponentna arhitektura

Ova arhitektura je pogodna za sisteme s malim potrebama integracije i malim brojem zastupljenih tehnologija. Zasniva se na modulima koji koriste i koordiniraju rad velikog broja komponenti.

Komponenta je softver koji obavlja čvrsto definisanu funkcionalnost uz određene parametre (sučelje). Svrha komponente je da se njena funkcionalnost može koristiti u različitim poslovnim scenarijima. Zato je cilj razvoja komponente da parametri, koji su jedini vidljivi korisnicima, što bolje definišu zadanu funkcionalnost.

U modeliranju poslovnih procesa komponenta predstavlja funkcionalnost jednog zadatka koji može, i ne mora, biti podržan ljudskom interakcijom.

Loše strana ovakve arhitekture:

- Otežano održavanje zbog fizičke povezanost komponenti
- Vežanost za tehnologiju i platformu
- Otežano dijeljenje komponente s drugim sistemima

Distribuisanost komponente je najčešće vezana za platformu/tehnologiju. U slučaju da se pojavi potreba za prenošenjem te funkcionalnosti na drugu platformu, postavlja se pitanja da li komponentu treba prebaciti u servis.


#### 4.1.3 Servisno orijentisana arhitektura

Jedna od definicija Servisno orijentisane arhitekture (SOA) je: *“SOA je arhitekturni stil za izgradnju enterprise rješenja zasnovanih na servisima. Specifično, SOA se bavi izgradnjom neovisnih poslovno orijentisanih servisa koji se mogu kombinirati u smislene poslovne procese i rješenja visokog nivoa na nivou cijelog enterprise sistema.*

Dakle, ključna značajka SOA arhitekture je da je ona **Arhitekturni stil** (u smislu *IT arhitekture*).

SOA nudi prednost da je opis IT okruženja iskazan na višem nivou apstrakcije: pomoću glavnih komponenti, njihovih odnosa i podataka koje razmjenjuju. Na taj način gradi se plan za izgradnju IT okruženja koje zadovoljava zahtjeve postavljene na njega, sada i u budućnosti

Doseg SOA arhitekture je cijeli enterprise. U poslovnom sektoru cijelu poduzeće, a u javnom sektoru državno organo ili cijela država.

	PROJEKT IZRADA I USPOSTAVLJANJE OKVIRA INTEROPERABILNOSTI I STANDARDA ZA RAZMJENU PODATAKA  <b>VODILJE I STANDARDI ZA          ARHITEKTURU SUSTAVA I RAZVOJ APLIKACIJA</b>	Oznaka: PARCO-IFBIH
		Page: 27 of 88

Gradivni elementi arhitekture su servisi. Kombiniranjem servisa ostvaruju se poslovni procesi, pri čemu je ključna podređenost poslovanju odnosno implementaciji poslovnih procesa. Servisi imaju životni ciklus, koji traje od osmišljanja samog servisa do njegovog povlačenja iz upotrebe

Sistemi koji koriste različite tehnologije na različitim platformama moraju funkcionalnosti definisati na nivou sučelja, a implementaciju prepustiti pružatelju usluge („servis provider“).

Servis kao softverska cjelina predstavlja izlaganje funkcionalnosti komponente putem standardnih komunikacionih protokola (SOAP). Povrh toga, protokoli omogućavaju i popisivanje i identifikaciju servisa u jedinstvenom registru (UDDI, WSDL). Takva arhitektura omogućava definisanje još složenijih rješenja međusobnom orkestracijom servisa (ESB).

U modeliranju poslovnih procesa, servis predstavlja automatizirani zadatak. Korištenje servisa u poslovnim procesima potpuno je transparentno za sve nivoe procesa.

Servisna arhitektura omogućava:

- Zamjenu tehnologije i platforme (modernijom ili potpuno različitom)
- Zamjenu samog pružatelja usluge
- Dijeljenje usluga s drugim sistemima
- Pouzdanost korištenjem distribuirane arhitekture

Mane servisne arhitekture na koje se mora posebno voditi računa kod planiranja su:

- Potencijalno duga vremena odziva servisa
- Složenost arhitekture koja angažira resurse različitih odjela
- Sigurnosni mehanizmi moraju biti implementirani na različitim tehnologijama u distribuiranom sistemu
- Otvorenost servisa nekada dovodi do neplaniranih scenarija (opterećenje)

## 4.2 Poslovna pravila

Funkcionalnost svake od komponenti sistema, a zavisno od odabranog arhitekturnog modela mora osiguravati sve potrebne funkcionalnosti a u skladu s poslovnim pravilima koja su definisana kroz procesni pogled iz poglavlja 3 ovog dokumenta.


Osim samih potrebnih funkcionalnosti poslovnim pravilima također su definisane i interakcije između određenih komponenti sistema.

## 4.3 Podaci (strukture, poruke, dokumenti)

Arhitekturni sloj koji osigurava pohranu – persistentnost podataka realizuje se prema pristupima i standardima kako je to opisano u poglavlju 5.

## 4.4 Sigurnosni mehanizmi (uloge, grupe, protokoli)

Funkcionalna arhitektura sistema mora biti usklađena sa svim sigurnosnim aspektima, a kako je to specificirano u poglavlju 6.3 ovog dokumenta.

	PROJEKT IZRADA I USPOSTAVLJANJE OKVIRA INTEROPERABILNOSTI I STANDARDA ZA RAZMJENU PODATAKA  <b>VODILJE I STANDARDI ZA          ARHITEKTURU SUSTAVA I RAZVOJ APLIKACIJA</b>	Oznaka: PARCO-IFBIH
		Page: 28 of 88

## 4.5 Sučelje (GUI)

Korisničko sučelje predstavlja eksterni pogled onoga što će krajnji korisnik vidjeti na ekranu, kao i komplet pravila koje će se koristiti da se u potpunosti postignu jednostavnost i integracija. Obavezno je da korisničko sučelje podržava na sva tri službena jezika u BiH (srpski, hrvatski, bosanski) i dva alfabeta (srpski ćirilčni i latinični).

Korisnik treba imati isti "izgled i osjećaj" u svim segmnetima interakcije sa sistemom, a potezi krajnjeg korisnika trebaju imati jednako značenje kroz cijeli sistem. Korisničko sučelje treba biti standardizovano i dogovoreno između naručioca i implementatora tokom faze dizajna te dokumentirano u okviru projektne dokumentacije. Primarni cilj standardizacije sučelja jest da sučelje bude intuitivno, lako za korištenje kao i da ne zbunjuje korisnika prilikom njegove interakcije s aplikacijom.

Elementi standarda korisničkog sučelja uključuju:


- Detaljan opis organizacije i prikaza prozora korisničkog sučelja (logotipi, naslovi, kontrole prozora: zatvaranje, promjena veličine, traka za pomicanje, traka s izbornicima, traka s alatima, korištenje komandnih tipki...)
- Detaljan opis radne oblasti prozora
- Detaljan opis navigacije i izbornika (tipovi izbornika, elementi izbornika,
- Detaljan opis implementacije toka procesa
- Opise svih korištenih prikaza i prezentacija podataka u aplikaciji (tabelarnih prikaza, načina filtriranja, sortiranja, listi, kontrola, greški, upozorenja, upita...)
- Korištene tipografske konvencije (upotreba fontova, veličine znakova, boja, zvuka,...)

## 4.6 Infrastruktura

Prilikom specifikacije arhitektura i funkcionalnosti potrebno je odrediti sve potrebne infrastrukturne komponente i metodologije koje se primjenjuju.

Specifikacijom odgovarajuće infrastrukture potrebno je osigurati:


- selekciju odgovarajućih infrastrukturnih komponenti a prema funkcionalnim zahtjevima za rad sistema
- osiguravanje potrebne skalabilnosti sistema i infrastrukture
- maksimalnu raspoloživost sistema
- fizičku zaštitu sistema
- primjenu zaštite svih komponenti sistema u ovisnosti od potreba za njihovim šticećenjem
- klasifikacija sistema i njihovih komponenti u posebne sigurnosne zone
- uspostava sistema za upravljanje pristupom na sisteme i upravljanje pristupom s mreže – udaljeni pristup (korištenje odgovarajućih sigurnosnih koncepata i protokola prema poglavlju 6.3)
- osiguravanje jednostavnog korištenja cjelokupne infrastrukture, efikasnog održavanja infrastrukture i svih komponenti sistema koje se na nju oslanjaju

	<p>PROJEKT IZRADA I USPOSTAVLJANJE OKVIRA INTEROPERABILNOSTI I STANDARDA ZA RAZMJENU PODATAKA</p>	<p>Oznaka: PARCO-IFBIH</p>
	<p>VODILJE I STANDARDI ZA ARHITEKTURU SUSTAVA I RAZVOJ APLIKACIJA</p>	<p>Page: 29 of 88</p>

## 4.7 Općenite smjernice

Osim specifičnih funkcionalnih zahtjeva, kod određivanja arhitekture se moraju poštivati i određena nefunkcionalna pravila:

- Proširivost – Sposobnost dodavanja novih funkcionalnosti ili promjene postojećih tako da se ne naruši rad sistema. Ovo se posebno odnosi na aplikacije koje su osjetiljve na izmjene zakona.
- Fleksibilnost – Sposobnost da se izmijeni arhitektura sistema bez narušavanje funkcionalnosti. Na ovaj način se postiže ispunjavanje nefunkcionalnih zahtjeva koji se mogu vremenom pojaviti (nova mrežna topologija, virtualizacija, sigurnosni mehanizmi).
- Interoperabilnost – standardizovanje dokumenta, formata i struktura podataka na nivou sistema.
- Otvorenost – Korištenje otvorenih standarda uvelike olakšava proširivosti i integrisanje sistema. Odnosi se i na formate podataka i na programska sučelja (API). Uvjet je da otvoreni standard bude vrlo dobro dokumentiran odn. implementiran.
- Performanse – Odziv komponenti sistema i ostali parametri moraju zadovoljavati zahtijevane standarde. Poželjno je da ti parametri budu definisani na samom početku.
- Sigurnost – Pristup podacima i komponentama sistema se mora uskladiti s internim zahtjevima i s eventualnim zakonskim standardima.
- Skalabilnost – Potrebno je omogućiti da se sistem može prilagoditi povećanim parametrima sistema od onih početnih (broj korisnika, opterećenje). Jedan od načina je da se aplikacija izvede u distribuiranoj arhitekturi.
- Pouzdanost – Potrebno je održati maksimalnu moguću dostupnost aplikacije prema zadanim zahtjevima i raspoloživim resursima. Distribuirana arhitektura omogućava load-balancing, a poželjno je i eliminirati kritične komponente („single point of failure“).
- Ažuriranje – Proces ažuriranja komponenti ne smije biti presložen tj. mora biti prilagođen zaduženom osoblju. Posebno treba obratiti pažnju na kompatibilnost komponenti.
- Ponovno korištenje komponenti – Komponente koje se mogu ponovno iskoristiti podižu nivo pouzdanosti jer su istestirane na više sistema, te se smanjuje krivulja učenja i kod razvoja i kod održavanja. Da bi se ovo omogućilo, funkcionalnost komponente moraju biti čvrsto definisana.

	<p style="text-align: center;">PROJEKT IZRADA I USPOSTAVLJANJE OKVIRA INTEROPERABILNOSTI I STANDARDA ZA RAZMJENU PODATAKA</p>	<p>Oznaka: PARCO-IFBIH</p>
	<p style="text-align: center;"><b>VODILJE I STANDARDI ZA ARHITEKTURU SUSTAVA I RAZVOJ APLIKACIJA</b></p>	<p>Page: 30 of 88</p>

## 5 Podatkovni pogled

### 5.1 Modeliranje podataka i podatkovna arhitektura

Da bi se postigla semantička interoperabilnost, potrebno je definisati uniformno značenje (semantiku) i sintaksu podataka koji se razmjenjuju. No, da bi se mogla definisati semantika podataka, svaki podatak mora biti povezan i sa poslovnim kontekstom u kojem se pojavljuje.

Za modeliranje podataka u tako složenim sistemima kao što je državna uprava nužno je korištenje modela koji omogućava povezivanje pogleda na poslovnu domenu (poslovne potrebe i zahtjeve), procesnu domenu (odvijanje poslovnih transakcija i dokumente koje te transakcije uključuju) i podatkovnu domenu (podatkovne objekte koji se razmjenjuju).

UML je općeprihvaćen jezik koji omogućava modeliranje podataka u kompleksnim poslovnim scenarijima.

UML Use Case dijagram opisuje poslovni kontekst pojedine poslovne transakcije. On definiše aktere pojedine poslovne transakcije i slučajeve u kojima se ona koristi.

Tok odvijanja same transakcije opisuje UML Activity dijagram. UML Activity dijagram opisuje pojedine korake u poslovnoj transakciji, aktere/sisteme koji su za te korake nadležni te podatke/dokumente koji se razmjenjuju između pojedinih koraka.

Za modeliranje podataka koristi se UML Class dijagram. UML Class dijagram opisuje podatke koje sadrže pojedini podatkovni entiteti te definiše veze (relacije) između entiteta.


Na osnovi UML modela podataka mogu se kreirati bilo XML sheme za razmjenu podataka bilo modeli baze podataka.

Za definisanje modela podataka u relacionim bazama mogu se koristiti i ER-dijagrami.

### 5.2 Standardi za opisivanje i razmjenu podataka

Formati podatka moraju omogućavati pohranu i razmjenu podataka na svim jezicima i pismima koji se koriste u BiH.

Unicode® je standard za razmjenu podataka usmjeren na prikaz slova na način neovisan o jeziku, računalnom programu ili računalnoj platformi. Da bi se postigla kompatibilnost sa starijim računalnim sistemima, postoji više različitih načina kodiranja Unicode znakova. Najznačajniji su UTF-8 (8-bitno kodiranje, s promjenjivim brojem bajtova po znaku) te UCS2 (16-bitno kodiranje, s nepromjenjivim brojem bajtova po znaku, ali podržava samo Basic Multilingual Plane)

	PROJEKT IZRADA I USPOSTAVLJANJE OKVIRA INTEROPERABILNOSTI I STANDARDA ZA RAZMJENU PODATAKA	Oznaka: PARCO-IFBIH
	VODILJE I STANDARDI ZA ARHITEKTURU SUSTAVA I RAZVOJ APLIKACIJA	Page: 31 of 88

**Za pohranu i razmjenu tekstualnih podataka preporuča se korištenje isključivo Unicode formata zapisa znakova. Preporuča se UTF-8 kodiranje radi očuvanja kompatibilnosti sa postojećim (legacy) sistemima.**

**Za pohranu binarnih podataka (slika, zvuk, video zapisi) i kompresiju se preporuča korištenje otvorenih formata.**

### 5.2.1 XML


XML je kratica za EXtensible Markup Language - odnosno jezik za označavanje podataka. Ideja je bila stvoriti jedan jezik koji će biti jednostavno čitljiv i ljudima i računalnim programima. Princip realizacije je vrlo jednostavan: odgovarajući sadržaj treba se uokviriti odgovarajućim oznakama koje ga opisuju i imaju poznato, ili lako shvatljivo značenje. Format oznaka u XMLu vrlo je sličan formatu oznaka u npr. HTML jeziku. Danas je XML jezik vrlo raširen i koristi se za različite namjene: odvajanje podataka od prezentacije, razmjenu podataka, pohranu podataka, povećavanje dostupnosti podataka i izradu novih specijaliziranih jezika za označavanje. XML je standardizovani jezik i za njegovu standardizaciju brine se World Wide Web Consortium.

XML dokument je obična tekstualna datoteka koja je čitljiva gotovo na svakoj platformi. XML omogućuje svakoj strani opisati podatke koje ima, odnosno one koje im trebaju odgovarajućim oznakama koje su čitljive i čovjeku i koje dosta dobro opisuju podatak koji se unutar određene oznake nalazi.

XML ima brojne prednosti pred tradicionalnim tekstualnim i binarnim formatima za razmjenu sadržaja:

- jednostavno je čitljiv i čovjeku u običnom tekstualnom editoru i računalu
- XML dokument je obična tekstualna datoteka čitljiva na svakoj platformi koja može čitati tekstualne podatke. To ga čini neosjetljivim na tehnološke promjene jer bez obzira na napredak tehnologije, tekstualni podaci će još jako dugo ostati nešto što će svaki kompjuterski sistem trebati moći pročitati
- podržava Unicode i omogućuje prikaz teksta na svim danas poznatim jezicima
- format je samodokumentirajući. Oznake opisuju sadržaj koji se nalazi unutar njih.
- ima stroga sintaksna pravila tako da je jednostavno kontrolirati ispravnost nastalog dokumenta. Računalni programi koji obrađuju dokument zbog toga mogu jednostavno obrađivati XML sadržaj
- međunarodno prihvaćen standard. Mnogi proizvođači programa su ga prihvatili i koriste u svojim proizvodima
- hijerarhijska struktura je pogodna za opisivanje mnogih sadržaja (ali ne i svih!)

Kod tekstualnih datoteka, format tih datoteka obično je bio pozicioni (svi podaci u jednom retku, a svaki podatak ima određenu duljinu) ili delimiterski (podatci razdvojeni posebnim znakom - delimiterom npr. zarez ili tačka-zarez). Problem kod toga je što je za svaku posebnu razmjenu potrebno raditi novi format prema dogovoru strana u razmjeni. Osim toga ti formati su neintuitivni. Bez dokumentacije toga formata nije moći niti naslutiti kakvi podaci se u njoj nalaze. S druge strane binarni formati, mada efektoviti, ne omogućavaju pristup podacima bez

	PROJEKT IZRADA I USPOSTAVLJANJE OKVIRA INTEROPERABILNOSTI I STANDARDA ZA RAZMJENU PODATAKA	Oznaka: PARCO-IFBIH
	<b>VODILJE I STANDARDI ZA          ARHITEKTURU SUSTAVA I RAZVOJ APLIKACIJA</b>	Page: 32 of 88

posebnog programa koji razumije te podatke, a također ne pružaju nikakve dodatne informacije o sadržaju podataka bez dokumentacije formata.

Danas sve više računalnih programa umjesto zatvorenih binarnih formata (npr. doc, xls,...) za pohranu podataka koristi XML format. Zbog svoje fleksibilnosti, otvorenosti, neovisnosti o tehnologiji i mogućnosti definisanja korisničkih gramatika (shema) XML je postao *lingua franca* za razmjenu podataka između različitih informatičkih sistema.

XML 1.0 je kreirana 1998. godine.. Verzija XML 1.0 bazira se na filozofiji da je zabranjeno sve što nije dozvoljeno. Verzija XML 1.1 inicijalno je objavljena 2004. godine i podržava širi skup Unicode znakova nego verzija 1.0. Njezin pristup je da je dozvoljeno sve što nije zabranjeno. Na takav način omogućuje se korištenje svih budućih Unicode znakova koji će se bilo kada u budućnosti definisati, ali je to dovelo do problema u kompatibilnosti pojedinih računalnih programa, tako da nije u širokoj upotrebi.

**Preporuka je da se zbog svoje raširenosti, kompatibilnosti i podrške u gotovo svim alatima u razmjeni podataka koristi verzija XML 1.0 jer je zadovoljavajuća za većinu korisnika.**

### 5.2.2 XML Shema

Korištenje XML-a kao formata za razmjenu podataka ne osigurava nužno semantičku interoperabilnost. Semantička interoperabilnost osigurava se tek jednoznačnom semantikom XML elemenata, koja se može postići npr. definisanjem odgovarajuće gramatike izražene kao XML sheme.

Postoji nekoliko tehnologija za definisanje i validaciju gramatika XML dokumenata. Te tehnologije nazivaju se XML sheme i omogućavaju formalan način zapisivanja dopuštenih naziva i poretka XML elemenata u nekom dokumentu, a time i definisanje njihove semantike.


Najšire korištena tehnologija za definiciju XML shema je XSD shema definisana od strane W3C konzorcija koji definiše standarde za razmjenu podataka na internetu. Pored definisanja sintakse i strukture XML dokumenta, XSD shema omogućava i definiciju tipova podataka pojedinih elemenata i atributa te kontrolu njihovih vrijednosti.

**Preporučuje se da se W3C XSD Shema koristi kao tehnologija za definisanje gramatika XML dokumenata koji se razmjenjuju.**

Pored definisanja semantike, XML shema omogućava i provjeru da li pojedini XML dokument odgovara unaprijed definisanim pravilima, što značajno olakšava kontrolu ispravnosti razmjenjenih podataka.

Postoje brojni međunarodni i industrioni standardi koji definišu semantiku XML elemenata za pojedine oblasti. Kod bilo kakve razmjene podataka treba uvijek nastojati koristiti neki od postojećih standarda, a tek ako takav ne postoji za danu oblast ili nije adekvatan može se pribjeći definisanju vlastite sheme.



	PROJEKT IZRADA I USPOSTAVLJANJE OKVIRA INTEROPERABILNOSTI I STANDARDA ZA RAZMJENU PODATAKA	Oznaka: PARCO-IFBIH
	<b>VODILJE I STANDARDI ZA          ARHITEKTURU SUSTAVA I RAZVOJ APLIKACIJA</b>	Page: 33 of 88

Naime, osim konzistentnog definisanja same semantike pojedinih XML elemenata u te je standarde ugrađena velika količina znanja koje može značajno pomoći da razmjena podataka bude kvalitetnija i predvidi neke zahtjeve koji će se pojaviti tek u budućnosti. Nadalje, korištenje standarda u razmjeni podataka povećava mogućnosti korištenja već gotovih programskih rješenja umjesto razvoja vlastitih.

**Preporučuje se da svi XML dokumenti koji se razmjenjuju budu zasnovani na odgovarajućim XML shemama.**

Da bi se omogućili korištenje različitih XML shema u jednom dokumentu, koristi se tehnologija *XML Namespaces* koja omogućava definiciju kojoj pojedinoj XML shemi pripada svaki pojedini XML element ili atribut.

**Preporučuje se da svi XML dokumenti koji se razmjenjuju koriste XML Namespace da bi osigurali i precizno definisali semantiku pojedinih XML elemenata i njihovih vrijednosti.**

Budući da sam XML omogućava puno različitih načina zapisa istih podataka (npr. smještanje podataka u elementima ili atributima, način dodjele odgovarajućih imena itd). važno je definisati i zajedničke smjernice za dizajn XML shema, takozvani „Naming and design rules“. Te smjernice definišu način prevođenja poslovne logike u XML sintaksu definisanu XML shemom.

Isporuke Komponente 2 definišu način izgradnje i početni zajednički rječnik jezgrenik komponenti.

**Preporučuje se kod izrade XML shema koristiti pravila (*Naming and design rules*) definisana u sklopu Komponente 2**


Važan aspekt uspostave interoperabilnosti su i kontrolirani rječnici (šifarnici) za vrijednosti pojedinih XML elemenata koji definišu njihovu semantiku, budući da vrijednost pojedine oznake sama po sebi ne nosi nikakvo značenje (npr. „HR“ može biti oznaka države, odjela za ljudske resurse ili nečeg drugog).

### 5.2.3 Prateći XML standardi

Pored samog XML-a, razvijen je veliki broj standarda i tehnologija za različite vrste manipulacije s XML dokumentima i njihovim sadržajem.

Preporuča se korištenje sljedećih pratećih standarda i tehnologija:

- **XPath** je jezik koji omogućuje jednostavnije pretraživanje sadržaja u XML dokumentu.
- **XQuery** je upitni jezik za pretraživanje XML dokumenta. On je za XML isto ono što je SQL za relacijske baze podataka
- **XSLT** je jezik koji omogućuje transformacije XML dokumenata iz jednog formata u drugi (npr. iz XMLa u XHTML)
- **XML Document Object Model (DOM)** je sučelje koje omogućuje računalnim programima pristup i ažuriranje sadržaja i strukture XML dokumenta
- **SOAP** je jednostavan protokol baziran na XMLu koji omogućuje aplikacijama razmjenu podataka u tekstualnom obliku preko HTTP protokola

	PROJEKT IZRADA I USPOSTAVLJANJE OKVIRA INTEROPERABILNOSTI I STANDARDA ZA RAZMJENU PODATAKA  <b>VODILJE I STANDARDI ZA          ARHITEKTURU SUSTAVA I RAZVOJ APLIKACIJA</b>	Oznaka: PARCO-IFBIH
		Page: 34 of 88

- **Web Services Description Language (WSDL)** je jezik baziran na XMLu koji omogućuje opis web servisa i sučelja za njihovo korištenje
- **XML Signature** definiše sintaksu i pravila za kreiranje digitalnog potpisa XML sadržaja
- **XML Encryption** definiše sintaksu i pravila za šifriranje XML sadržaja

#### 5.2.4 Semantičko označavanje tekstualnih HTML dokumenata (Open Linked Data)

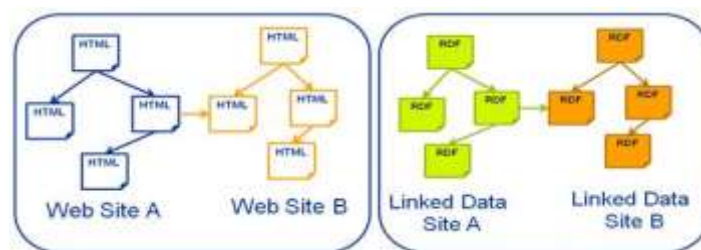
Semantički web je inicijativa kreatora HTML jezika Tim Berners Lee-a definisana kao: "Semantički web je ekstenzija postojećeg weba u kojem informacije imaju precizno definisano značenje, omogućavajući računalima i ljudima lakšu međusobnu suradnju". (Tim Berners Lee , Scientific American, May 2000)

Dakle, osnovni cilj semantičkog weba je omogućiti računalima (odnosno softverskim agentima) bolje razumijevanje relacija između različitih dijelova sadržaja i pretvaranja weba u mrežu podataka (za razliku od mreže dokumenata namijenjenih ljudima, kao što je to sada).

Semantički web je potpuno interoperabilan sa postojećim HTML dokumentima od kojih se sastoji „današnji“ web i definiše četiri osnovna principa:


- Korištenje URI-a kao oznaka za označavanje koncepata
- Korištenje HTTP URI-a koje je lagano dereferencirati od strane korisnika i koji pružaju dodatne informacije o značenju samog URI-a
- Pružanje svih podataka u obliku čitljivom korisnicima (*human readable* – HTML) i u obliku čitljivom računalima (*machine readable* – RDF, RDF/a, mikroformati)
- Povezivanje podataka s drugim povezanim podacima gdje god je to moguće (tj. korištenje informacije na njezinom izvoru umjesto duplikacije).

Principi semantičkog weba prikazani su na sljedećoj slici:



Slika 6 Principi semantičkog weba

Povezana inicijativa nazvana „Open Linked Data“ nadogradnja je ideje semantičkog weba, pružajući različite podatke i identifikatore za koncepte koji su dostupni tehnologijama semantičkog weba. Omogućava kreiranje potpuno novog oblika aplikacija (tzv. „*mashups*“), u kojima se povezuju podaci iz različitih izvora i na taj način im se daje novo značenje i pruža dodatna vrijednost korisnicima.

	PROJEKT IZRADA I USPOSTAVLJANJE OKVIRA INTEROPERABILNOSTI I STANDARDA ZA RAZMJENU PODATAKA	Oznaka: PARCO-IFBIH
	<b>VODILJE I STANDARDI ZA          ARHITEKTURU SUSTAVA I RAZVOJ APLIKACIJA</b>	Page: 35 of 88

Inicijativa je naišla na znatan odziv tokom zadnjih godina i razvija se vrlo brzo uz pomoć pojedinih vlada zemalja Evropske unije (npr. Velike Britanije – data.gov.uk) kao i Evropske komisije.

Na taj se način omogućava puno lakše otkrivanje informacija jer se na podatke mogu primjeniti formalna pravila, a time i izvoditi zaključci. Takav pristup omogućava i puno bilje upravljanje znanjem i njegovo korištenje, jer sistem može koristiti ne samo znanje pohranjeno u pojedinom dokumentu, nego i povezano znanje svih drugih izvora i znanje samih korisnika i to znanje ponovno ugraditi u sadržaj i time ga dodatno obogatiti.

To je posebno važno kod sistema za pretraživanje sadržaja, jer se tražilici omogućava dublje razumijevanje sadržaja dokumenta (nego što je to moguće jednostavnom pretragom teksta) i posebno relacija između pojedinih pojmova, pa ona korisnicima može ponuditi znatno preciznije odgovore.

Inicijativu su prihvatile i vodeće tražilice na Internetu kao što su Bing, Google, Yahoo! i Yandex te su kreirale web stranicu „Schema.org“. Shema.org pruža skup XML shema koje definišu oznake pojedinih metapodataka na način razumljiv internetskim tražilicama. Uključivanjem tih oznaka u dokumente one mogu steći bolji uvid u sadržaj stranica koje koriste te metapodatke i tako ponuditi korisnicima preciznije odgovore na njihove upite.

Naime, mnogi dokumenti generiraju se iz strukturiranih podataka. Uključivanjem strukturiranog sadržaja u HTML dokumente gubi se značenje izvornih podataka, te oni nisu više iskoristivi tražilicama. Uključivanjem tih podataka na način čitljiv računalima (a koji je posve nevidljiv ljudskim korisnicima) strukturirani sadržaj se zadržava i postaje dostupan tražilici, pa i njezini odgovori mogu biti puno precizniji.


**Preporuka je da se principi semantičkog weba i Open Linked Data inicijative ugrade u HTML tekstualne dokumente gdje god je to moguće i koriste za identifikaciju i povezivanje metapodataka, a posebno imenovanih entiteta (osoba, institucija, tvrtki, zemljopisnih pojmova), te se dokumenti tako učine dostupnijim tražilicama i drugim inteligentnim agentima.**

### 5.3 Zajednička biblioteka jezgrenih komponenti

XSD Shema može i uključivati druge XML sheme, te na taj način omogućava stvaranje interoperabilnih modula ili biblioteka za opisivanje pojedinih vrsta podataka. Jedan od najvažnijih načina definicije takovih biblioteka je korištenje UN/CEFACT Core Components tehnologije.

*Jezgrena komponenta* je semantički građevni blok koji se može koristiti za bilo koji aspekt modeliranja podataka i informacija i njihove razmjene. One omogućavaju kreiranje modela poslovnih procesa i poslovnih dokumenata koji se mogu lako povezivati i međusobno razmjenjivati informacije.

Poslovni dokumenti se modeliraju na način da entitet koji je vlasnik podatka definiše shemu poslovnog dokumenta, ali pri tome kao gradive blokove koristi isključivo komponente iz zajedničke biblioteka jezgrenih komponenti, što osigurava definisanu zajedničku sintaksu i semantiku razmjenjenih podataka.

	PROJEKT IZRADA I USPOSTAVLJANJE OKVIRA INTEROPERABILNOSTI I STANDARDA ZA RAZMJENU PODATAKA	Oznaka: PARCO-IFBIH
	<b>VODILJE I STANDARDI ZA          ARHITEKTURU SUSTAVA I RAZVOJ APLIKACIJA</b>	Page: 36 of 88

Osnovni tipovi *Jezgrenih komponenti* su:

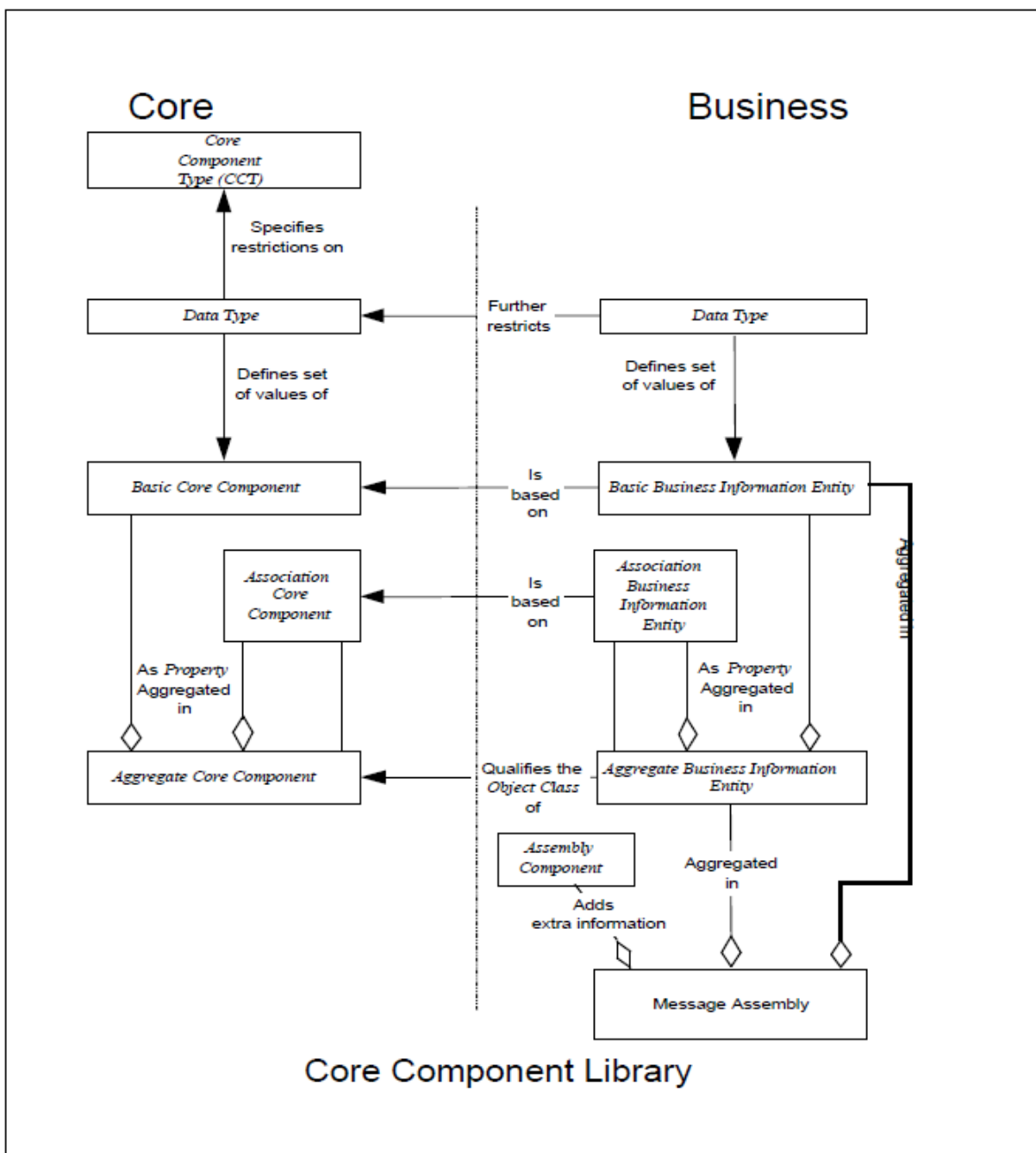
- **Osnovna jezgrena komponenta** (*Basic Core Component*) - jezgrena komponenta koja izgrađuje pojedinačnu poslovnu karakteristiku specifične Složene jezgrene komponente koja reprezentira objektu klasu. Ima jedinstvenu poslovno-semantičku definiciju.
- **Asocijacijska jezgrena komponenta** (*Association Core Component*) - jezgrena komponenta koja predstavlja kompleksnu poslovnu karakteristiku specifične Složene jezgrene komponente koja reprezentira objektu klasu.
- **Složena jezgrena komponenta** (*Aggregation Core Component*) -skup povezanih dijelova poslovnih informacija koje nose jedinstveno poslovno značenje, zavisno od bilo kakvog poslovnog konteksta
- **Tip jezgrene komponente** (*Core Component Type*) - jezgrena komponenta koja se sastoji od jedne i samo jedne sadržajne komponente koja nosi stvarni sadržaj plus jedne ili više dodatnih komponenti koje daju dodatnu definiciju sadržajnoj komponenti
- **Osnovni podatkovni tip** (*Core Data Type*).

Kad se osnovne jezgrene komponente stave u poslovni kontekst one predstavljaju podlogu na kojoj se izgrađuju Poslovni informacioni entiteti i na taj način se omogućava jedinstvenost poslovne semantike. Osnovne jezgrene komponente služe kao kontrolirani vokabular za izgradnju poslovnih informacionih entiteta.

Razlikujemo slijedeće kategorije poslovnih informacionih entiteta:

- **Osnovni poslovni informacioni entitet** (Basic Business Information Entity) - reprezentira pojedinačnu poslovnu karakteristiku specifične objektne klase u specifičnom poslovnom kontekstu
- **Asocijativni poslovni informacioni entitet** (Association Business Information Entity) - predstavlja kompleksnu poslovnu karakteristiku specifične objektne klase u specifičnom poslovnom kontekstu
- **Složeni poslovni informacioni entitet** (Aggregate Business Information Entity) - skup povezanih dijelova poslovnih informacija koji zajedno nose jedinstveno poslovno značenje u specifičnom poslovnom kontekstu
- **Poslovni podatkovni tip** (Business Data Type – BDT)


Veza *Poslovnih Informacionih Entiteta* i *Jezgrenih komponenti* prikazana je na slijedećoj slici.



Slika 7 UN/CEFACT CC i BIE - veze<sup>6</sup>

Komponenta 2 definiše specifikaciju modela jezgrenih komponenti, inicijalnu verziju biblioteke jezgrenih komponenti razvijenu za potrebe Bosne i Hercegovine te metodologiju za pronalaženje postojećih i definisanje novih jezgrenih komponenti.

<sup>6</sup> Prema UN/CEFACT CCTS V2.01, str. 18, Slika 4-2

	PROJEKT IZRADA I USPOSTAVLJANJE OKVIRA INTEROPERABILNOSTI I STANDARDA ZA RAZMJENU PODATAKA  VODILJE I STANDARDI ZA ARHITEKTURU SUSTAVA I RAZVOJ APLIKACIJA	Oznaka: PARCO-IFBIH
		Page: 38 of 88

**Preporuča se da gdje god je to moguće XML Sheme podataka koje se razmjenjuju ju državnoj upravi budu zasnovane za zajedničkoj biblioteci jezgrenih komponenti.**

Obzirom na složenost državne strukture u BiH te nepostojanje već razvijenih modela e-Poslovanja, pristup koji osigurava najveće šanse za uspostavu semantičke interoperabilnosti na nivou države je izgradnja zajedničke biblioteke jezgrenih komponenti zasnovane na UN/CEFACT biblioteci jezgrenih komponenti (CCL).

UN/CEFACT biblioteka jezgrenih komponenti (CCL) trenutno je u verziji 11A. Biblioteka se konstantno održava i razvija (izlaze dvije verzije godišnje), tako da osigurava i konstantni napredak samog modela.

**Preporuča se da biblioteka jezgrenih komponenti bude zasnovana na zadnjoj (najnovijoj) raspoloživoj biblioteci jezgrenih komponenti koja je dostupna u trenutku izrade biblioteke.**

## 5.4 Baze podataka


Preporuča se korištenje server baziranih, relacionih baza podataka

Korištenje ovakve tehnologije omogućava ostvarenje ovih prednosti:

- **Fleksibilnost.** Baze podataka u ovoj tehnologiji mogu podržati rješenje bilo kog problema upravljanja podacima, konfiguracija i upravljanje bazama je jednostavnije jer se odvija kroz definisana programerska sučelja (Application Programming Interfaces – API)
- **Performanse.** Baze u ovoj konfiguraciji mogu iskoristiti sve mogućnosti hardver platforme na kojoj se implementiraju, a prema definisanim performansnim zahtjevima.
- **Skalabilnost.** Korištenje baza podataka u ovoj konfiguraciji omogućava brzu prilagodbu promjenama za obimom bilo da se tiče korisnika ili podataka, odnosno zahtijevanim performansama sistema.
- **Integritet** – Korištenjem ove tehnologije osiguran je integritet podataka.

Prilikom postupka odabira baze podataka potrebno je provjeriti slijedeće korake:

1. **Sadrži li vaša baza podataka podatke od kritične važnosti ?** To znači da se bili kakav gubitak podataka, kao niti ispad sistema ne može tolerirati.
2. **Je li vam potrebna 24x7 raspoloživost baze podataka?** Vrlo malo baza podataka bi trebao biti u ovoj kategoriji. Puno je lakše zadovoljiti uvjet da bi se bilo operativan 95% vremena nego 100% vremena jer visoka dostupnost sistema znatno poskupljuje njegovo održavanje. Ako vam je potrebna, dobro argumentirajte zahtjeve.
3. **Kakvi su zahtjevi na backup sistema?** Dva osnovna načina backupa su hot (baza je dostupna korisnicima za vrijeme backupa) i cold (baza nije dostupna korisnicima za

	PROJEKT IZRADA I USPOSTAVLJANJE OKVIRA INTEROPERABILNOSTI I STANDARDA ZA RAZMJENU PODATAKA	Oznaka: PARCO-IFBIH
	<b>VODILJE I STANDARDI ZA          ARHITEKTURU SUSTAVA I RAZVOJ APLIKACIJA</b>	Page: 39 of 88

vrijeme backupa) backup. Provjerite zahtjeve za tim da li je potrebno da baza bude stalno dostupna korisnicima.

4. **Koliko je velika vaša baza podataka?** Potrebno je realno procijeniti potrebe za veličinom baze podataka. To uključuje sve indekse, tablice, backupove i fileove za replikaciju. Prilikom obavljanja procjene često dolazi do pogrešaka, tako da dobra praksa pokazuje da se inicijalna procjena množi s faktorom 10 što obično ispunjava zahtjeve za prostorom za prve dvije godine rada sistema.
5. **Kakva je očekivana stopa rasta veličine baze podataka?** To utječe na skalabilnost baze podataka, te hardvera i softvera.
6. **Koje vrste podataka će biti sadržane u bazi?** (Binarni, veliki objekti?)
7. **Trebate li replikaciju baze?** Da li će ova baza podataka biti na jednom serveru ili će biti raspoređena na nekoliko mjesta.
8. **Koliko istodobnih korisnika baze podataka mora biti podržano?**
9. **Koja je vrsta podrške potrebna uz bazu podataka?** Kategorije podrške mogu biti: podrška prilikom dizajna ER (Entity Relationship) shema, razvoj aplikacija, održavanje poslužitelja, administriranje hardvera i operativnog sistema, instalacija zakrpa i nadogradnji.
10. **Zahtjeva li se podrška rada s određenim platformama ?** U određenim slučajevima može se zahtijevati da baza podataka mora raditi na određenoj platformi, te se onda takvi zahtjevi stavljaju kao jedan od kriterij baze podataka koja se implementira.
11. **Vremenska dinamika uspostave baze podataka?**
12. **Očekivani period rada za bazu podataka – npr. dostupnost: 1 godina, 5 godina, 10 godina?** Za baze podataka s dugim vremenom života, potrebno je osigurati kontinuitet podrške za bazu.

Prema rezultatima ove liste potrebno je specificirati zahtjeve na konkretan sistem baze podataka koji se implementira.

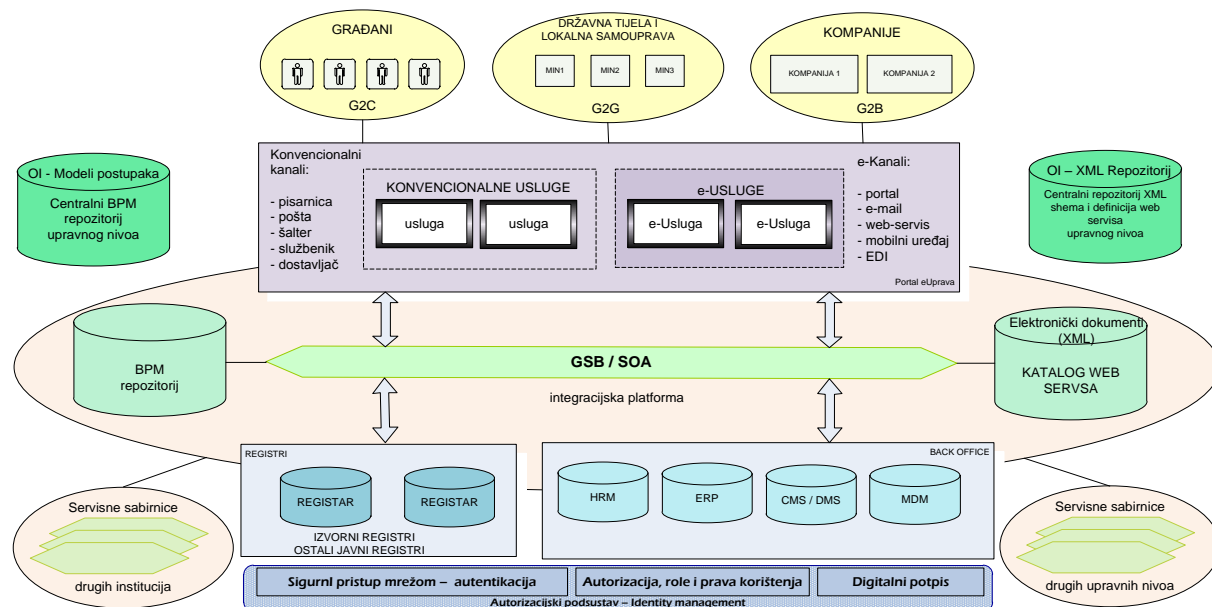
## 6 Integracioni pogled i razmjena podataka/dokumenata/poruka

### 6.1 Integracija IT sistema u državnoj upravi

Prilikom dizajniranja arhitekture IT sistema državne uprave potrebno je osigurati da sistem zadovoljava interese svih potencijalnih zainteresovanih strana: građana, kompanija kao i drugih organa javne uprave koji mogu imati potrebe za određenim podacima.

Pri tome se mora osigurati da svi korisnici – bez obzira na način pristupa usluzi – uvijek dobiju iste podatke i pod istim uvjetima, a da se podaci uvijek uzimaju sa mjesta koje je odgovorno za njihovo održavanje (kako bi se sprječilo korištenje podataka koji nisu ažurni i izbjegli troškovi njihove dvostruke administracije).

Da bi se to omogućilo, potrebno je na nivou pojedinih organa državne uprave kao i na nivou upravnog nivoa definisati integracijsku arhitekturu kao što je to prikazano na slici ispod:




**Slika 8 Integracijska arhitektura IT sistema**

Detaljni opis poslova, koji uključuje izgradnju integracijske arhitekture dan je u Okviru interoperabilnosti \_BiH, a određivanje organa nadležnog za uspostavu naveden arhitekture (uključujući servisnu sabirnicu) određuje se Odlukom o usvajanju Okvira interoperabilnosti BiH za svaki nivo vlasti pojedinačno.

Integracijska arhitektura mora definisati korisnike usluga, kanale kojima se usluga pruža konvencionalnim i elektronskim putem te način povezivanja usluge sa podacima (izvornim i izvedenim registrima i back-office sistemima).



	PROJEKT IZRADA I USPOSTAVLJANJE OKVIRA INTEROPERABILNOSTI I STANDARDA ZA RAZMJENU PODATAKA	Oznaka: PARCO-IFBIH
	<b>VODILJE I STANDARDI ZA          ARHITEKTURU SUSTAVA I RAZVOJ APLIKACIJA</b>	Page: 41 of 88

Iako pojedini podaci i aplikacije mogu biti implementirani u heterogenim tehnologijama, važno je da na nivou organa kao i na nivou centralne države postoji zajednička integracijska platforma koja omogućava integraciju svih poslovnih procesa i podataka državne uprave u jedinstvenu cjelinu.

Da bi se omogućilo upravljanje procesima i podacima, arhitektura mora definisati i repozitorij poslovnih procesa koji sadrži modele postupaka, repozitorij XML shema elektronskih dokumenata i poruka za razmjenu podataka te katalog web servisa.

Prilikom dizajniranja aplikacija potrebno je obratiti pozornost na slijedeće:

- poželjno je razdvojiti elemente odgovorne za poslovnu logiku aplikacije od elemenata koji služe za komunikacione protokole, transformaciju podataka i ispunjenje poslovnih ugovora. Na ovaj način se pospješuje provedba koncepta raščlambe problema.
- korisnici aplikacije mogu tražiti optimizaciju odgovora aplikacije za određene scenarije npr. neki korisnici mogu tražiti optimizaciju prikaza podataka, dok drugi korisnici traže optimizaciju procesiranja podataka.
- mogućnost različitih operativnih zahtjeva od strane različitih korisnika; npr. sigurnosni zahtjevi za autorizaciju za određene vrste operacija ovisno o vrsti korisnika, različiti oblici transakcijske podrške za različite korisnike i slično.
- mogućnost aplikacija da promptno odgovore na promjene u poslovnoj okolini drastično se povećava u slučaju da su promjene na poslovnoj logici razdvojene od tehnologije preko koje korisnici aplikacije pristupaju aplikaciji. Na primjer određena poslovna logika prvo implementirana kao custom komponenta koja se kasnije primjenjuje kao wrapper u SOA rješenju u pravilu ne bi trebala imati utjecaj na krajnje korisnike aplikacije.

Najbolju perspektivu kao tehnologija za integraciju IT sistema nudi Servisno orijentirana arhitektura (SOA), koja je detaljnije objašnjena u prethodnom poglavlju


SOA se najbolje uklapa u scenarije

- Centralizirane poslovne funkcije korištene od više različitih korisnika (u istoj ili drugim organizacijama)
- Integracija sa podacima u sistemima različitih organizacija
- Prisutne stare tehnologija koje se još uvijek koriste, a koje otežavaju direktnu integraciju podataka

No, neki osjetljivi sektori (bankarstvo, obrana...) ponekad su nepovjerljivi prema novim tehnologijama i njihovim mogućim nepoznatim ranjivostima

SOA arhitektura se ne uklapa dobro u sljedeća okruženja

- Real-time sistemi
- Integracija u homogenom okruženju (npr integracija programskih sistema od istog dobavljača, gdje se ponovna upotrebljivost lako ostvaruje bez SOA)

	PROJEKT IZRADA I USPOSTAVLJANJE OKVIRA INTEROPERABILNOSTI I STANDARDA ZA RAZMJENU PODATAKA	Oznaka: PARCO-IFBIH
	<b>VODILJE I STANDARDI ZA          ARHITEKTURU SUSTAVA I RAZVOJ APLIKACIJA</b>	Page: 42 of 88

## 6.2 Servisne sabirnice javne uprave (GSB – Government Service Bus)

S razvojem servisno orijentisane arhitekture postoji opasnost stvaranja velikog broja poprečnih veza i međuovisnosti između različitih komponenti IT sistema i aplikacija, posebno ako se uvođenje SOA arhitekture ne provodi planski i koordinirano s jednog mjesta, već je razvoj policentričan (kao što je to često slučaj u javnoj upravi).

Takove poprečne veze otežavaju kasnije održavanje i proširivanje, jer nastaje takozvana „špageti arhitektura“: s vremenom svi sistemi ovise o velikom broju drugih sistema, pa planiranje nadogradnje ili održavanja bilo kojeg od njih postaje izuzetno zahtjevno (ili čak nemoguće) zbog potrebe za koordinacijom velikog broja sudionika koji su često i pod različitom administrativnom kontrolom.

Rješenje koje se nameće je razdvajanje direktnih veza i enkapsulacija sistema na nivou organizacije ili kompletne javne uprave uvođenjem integracijske platforme ili servisne sabirnice javne uprave (Government Service Bus - GSB).

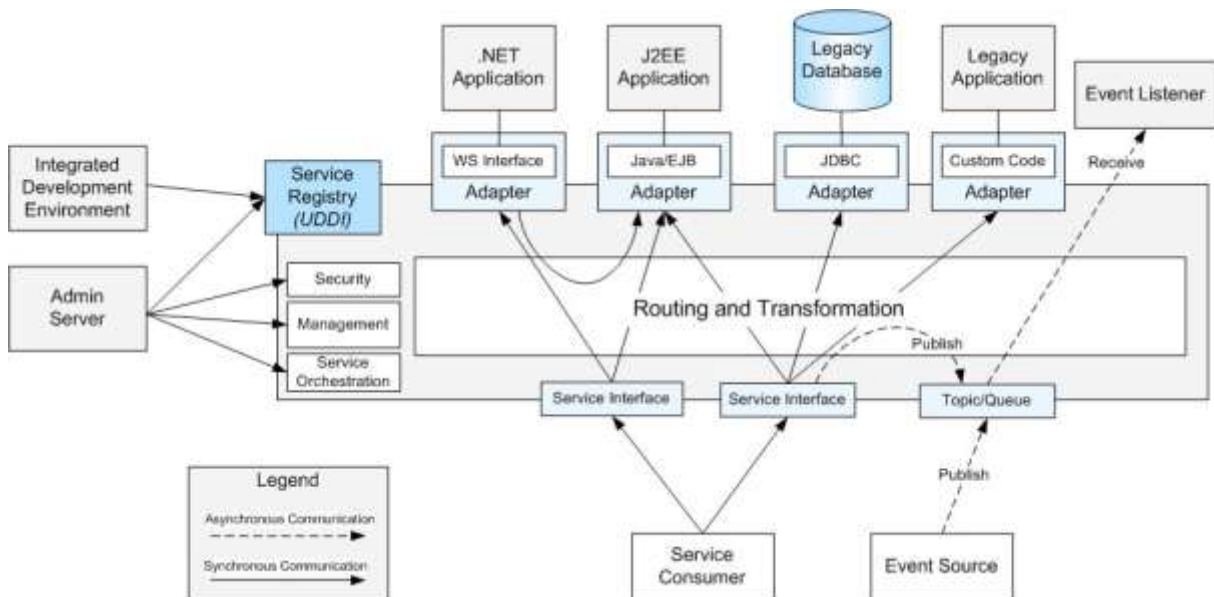
Sabirnica treba služiti kao mjesto za razmjenu podataka između aplikacije koja koristi podatke i organa koji objavljuje uslugu te povezivanje aplikacija i podataka u jedinstvenu cjelinu na nivou institucije javne uprave (ili upravnog nivoa). Servisne sabirnice javne uprave ujedno moraju omogućavati integraciju Servisno orijentisane arhitekture (SOA) te osigurati da se ažurni podaci uvijek uzimaju sa izvora (organa koji je odgovoran za njihovo prikupljanje – umjesto da se stvaraju lokalne kopije podataka koje je teško držati ažurnima), tako da se stvori jedna stabilna tačka za pristup podacima koja će garantirati njihovu dostupnost.

### 6.2.1 Funkcije servisnih sabirnica javne uprave

Uvođenje servisnih sabirnica javne uprave omogućava da sve aplikacije komuniciraju samo sa jednom sabirnicom, a sabirnice komuniciraju između sebe u skladu s predefinisanim orkestracijom poslovnih procesa na pojedinoj sabirnici.

Svaka servisna sabirnica javne uprave obavlja sljedeće funkcije:

- Komunikacione sabirnice - sigurne dostave poruka
- Integracija SOA arhitekture
- Katalog web servisa, shema, validacija i konverzija podataka
- Orkestracije poslovnih procesa
- Implementacije poslovnih pravila i sigurnosnih politika



Slika 9 Funkcije servisnih sabirnica javne uprave <sup>7</sup>

Neki od osnovnih zahtjeva koji GSB mora zadovoljiti su: velik broj pristupnih točaka za spajanje trenutnih i budućih aplikacionih sistema, različiti formati podataka i komunikacioni protokoli, konzistentna implementacija poslovnih pravila, aplikacije koje se izvršavaju paralelno na različitim platformama, aplikacije napisane različitim programskim jezicima i korištenjem različitih programskih modela.


### 6.2.1.1 Komunikaciona sabirnica

Komunikaciona sabirnica treba omogućiti razmjenu poruka (dokumenata) između pojedinih aplikacija, bilo preko servisno orijentisane arhitekture (SOA), bilo preko nekih drugih protokola kao što je elektronska pošta ili čak datoteke na disku. Svaka aplikacija komunicira isključivo sa sabirnicom, dok se sabirnica brine za usmjeravanje poruka, eventualnu promjenu komunikacionih protokola te dostavu potvrde o prijemu izvornoj aplikaciji.

Sabirnica mora omogućavati sigurnu i pouzdanu dostavu i neporecivost poruka, tako da je uvijek moguće utvrditi status svake pojedine poruke i osigurati da je poruka ili isporučena primatelju, ili je obavijest o grešci isporučena pošiljatelju poruke.

Sabirnica mora nuditi mogućnost konverzije formata poruke kada to zahtjeva poslovni proces i bogate mogućnosti transformacije podataka.

<sup>7</sup> USA, National Institutes of Health, Enterprise Architecture

	PROJEKT IZRADA I USPOSTAVLJANJE OKVIRA INTEROPERABILNOSTI I STANDARDA ZA RAZMJENU PODATAKA  <b>VODILJE I STANDARDI ZA          ARHITEKTURU SUSTAVA I RAZVOJ APLIKACIJA</b>	Oznaka: PARCO-IFBIH
		Page: 44 of 88

### **6.2.1.2 Integracija SOA arhitekture**

Objavljivanje servisa na sabirnici omogućava da se same aplikacije „sakriju“, a da se web servisima svih pojedinih aplikacija pristupa preko sabirnice. Na taj se način osigurava konzistentna pristupna tačka za pojedine aplikacije, olakšava verzioniranje i upravljanje web servisima, a time i integracija samih aplikacija. To je posebno važno kod aplikacija koje pružaju podatke vanjskim korisnicima, gdje sabirnica pruža centralno mjesto za pristup podacima prema vanjskim korisnicima, a skriva internu arhitekturu informacionog sistema.

### **6.2.1.3 Katalog web servisa, shema, validacija i konverzija podataka**

Katalog (repozitorij) web servisa i shema podataka treba da omogući objavljivanje (i pronalaženje) web servisa korisnicima radi njihove integracije i korištenja u poslovnim procesima. Katalog mora biti tijesno integrisan sa portalom, tako da se pomoću wizarda za generiranje servisa na portalu omogući i objavljivanje, korištenje i upravljanje sigurnosnim postavkama pojedinog servisa.

Katalog mora omogućavati pohranu WSDL opisa servisa te XML shema svih dokumenata i drugih poruka koje služe za komunikaciju u sistemu.

Sabirnica nudi mogućnost konverzije formata poruke kada to zahtjeva poslovni proces i nudi bogate mogućnosti transformacija i konverzije formata.

Validacija podataka osigurava sintaktičku i semantičku ispravnost ulaznih podataka u platformu te njihovu usklađenost s poslovnim pravilima. Validacija podataka može se provoditi na nekoliko nivoa:


- Validacija strukture podataka definisanih XML shemom
- Validacija poštivanja poslovnih pravila ulaznih podataka korištenjem Schematron tehnologije
- Validacija semantike podataka provjerom da vrijednosti pojedinih podataka odgovaraju kontroliranim rječnicima i drugim šifarnicima

Da bi se platforma što bolje prilagodila specifičnim potrebama pojedinih izdavatelja računa, u platformu se mogu ugraditi i konverteri za pojedine specifične formate, koji omogućavaju (korištenjem bogatih biblioteka pravila, funkcija i podržanih formata) svođenje specifičnih formata korisnika na jedinstveni format platforme.

### **6.2.1.4 Orkestracija poslovnih procesa**

Radi osiguranja fleksibilnosti sistema u slučaju promjena regulatornog okvira ili poslovnih procesa, GSB mora podržavati orkestraciju poslovnih procesa i jednostavnu integraciju pojedinačnih usluga / vanjskih IT sistema od kojih se proces sastoji.

GSB mora omogućiti upravljanje poslovnim pravilima (pregled, unos, izmjenu) bez pisanja programskog koda. Takva poslovna pravila zatim se koriste u samim orkestracijama, na primjer, za donošenje odluka u IF-THEN uvjetima grananja.

	PROJEKT IZRADA I USPOSTAVLJANJE OKVIRA INTEROPERABILNOSTI I STANDARDA ZA RAZMJENU PODATAKA	Oznaka: PARCO-IFBIH
	<b>VODILJE I STANDARDI ZA          ARHITEKTURU SUSTAVA I RAZVOJ APLIKACIJA</b>	Page: 45 of 88

GSB mora omogućavati i pristup legacy aplikacijama koje nisu dizajnirane za SOA arhitekturu te integraciju aplikacija na različitim platformama (npr. Java i .NET) na način da se kreiraju adapteri na sabirnici koji omogućavaju pristup podacima. To je posebno važno kod podataka u legacy sistemima, jer omogućava njihovo korištenje i integraciju bez potrebe njihove zamjene (što je često vrlo skupo i dugotrajno jer je potrebno osigurati očuvanje funkcionalnosti koja nije uvijek niti potpuno poznata i dokumentirana).


## 6.2.2 Načela uvođenja servisne sabirnice

### 6.2.2.1 Osnovna načela

Implementacija servisne sabirnice i odgovarajućih servisa mora ispuniti načela vezana za funkciju integracijskog sistema unutar cjelokupnog informacionog sistema pojedinog organa javne uprave.

Uspostavom sabirnice, svi sistemi i aplikacije u organizaciji moraju poštivati sljedeća pravila:

- Integracijska sabirnica mora funkcionirati kao jedinstvena pristupna tačka za svu programsku elektronsku komunikaciju između organizacije i eksternih korisnika i sistema (drugih organa te fizičkih i pravnih osoba); Pojedine aplikacije ne smiju direktno komunicirati sa vanjskim sistemima
- Prema vanjskim korisnicima odnosno povezanim aplikacijama, sabirnica mora biti dostupna putem niza elektronskih usluga izvedenih na otvorenim internetskim standardima, najčešće XML web servisima;
- Vanjski korisnici za pristupanje moraju dobiti samo ograničeni broj potrebnih tehničkih informacija. Konfiguraciju interne mreže i fizička lokacija pojedinih servisa interno uređuje institucija u skladu sa svojim potrebama. Sve promjene u internoj mreži, odnosno migriranja pojedinih komponenti (aplikacije, podaci) servisa, za krajnje su korisnike potpuno transparentni. Takav pristup bitno povećava fleksibilnost IT operacija i nivo sigurnosti.
- Sva unutarnja podatkovna komunikacija između informacionih sistema institucije (tzv. sistemski podatkovni tokovi – engl. *system workflows*) mora se odvijati putem integracijske sabirnice gdje god je to tehnički moguće; direktna integracija aplikacija stvara međuovisnosti između pojedinih komponenti sistema koje otežavaju kasnije održavanje i proširivanje sistema jer nastaje takozvana „špageti arhitektura“ (s vremenom svi sistemi ovise o velikom broju drugih sistema)
- Zajedničke funkcionalnosti svih elektronskih usluga odnosno podatkovnih tokova, vezane primarno na sigurnost elektronske komunikacije, moraju biti realizovane unutar sabirnice. Na taj način se postiže visoka iskoristivost postojećeg i provjerenog koda, a povezani se sistemi rasterećuju od potrebe za ponovnim razvojem istovrsnih funkcionalnosti.;


	PROJEKT IZRADA I USPOSTAVLJANJE OKVIRA INTEROPERABILNOSTI I STANDARDA ZA RAZMJENU PODATAKA  <b>VODILJE I STANDARDI ZA          ARHITEKTURU SUSTAVA I RAZVOJ APLIKACIJA</b>	Oznaka: PARCO-IFBIH
		Page: 46 of 88

- Translacija između formata definisanih za vanjsku komunikaciju (vanjski korisnici <-> integracioni sistem) i formata interne komunikacije (integracioni sistem <-> interni servisi, aplikacije, podaci, datoteke) provoditi će se na samoj sabirnici;
- Sva poslovna pravila koja imaju utjecaj na izvršavanje poslovnih procesa s više povezanih sistema ili se odnose na elektronske usluge za vanjske korisnike moraju biti definisana u repozitoriju poslovnih pravila (engl. *Business Rules Repository*) unutar ove standardizacije;
- XML sheme svih dokumenata i struktura podataka e-Servisa, kao i njihovo eventualno međusobno preslikavanje, moraju biti definisani u repozitoriju podatkovnih struktura unutar sabirnice kako bi se osigurala uspostava semantičke interoperabilnosti što omogućava provjera sintaktičke ispravnosti (*validacija*) dokumenata i poruka.

#### 6.2.2.2 Nefunkcionalni zahtjevi

Pri implementaciji integracijsko servisne sabirnice moraju se poštivati sljedeći nefunkcionalni zahtjevi:

- Svi aplikativni sistemi koji pristupaju orkestracijama i uslugama na integracionoj sabirnici moraju biti autentifikovani.
- Za autorizaciju izvršavanja e-usluga i orkestracije potrebno je koristiti centralni autorizacioni mehanizam unutar sabirnice. Taj autorizacioni mehanizam omogućava granularno dodjeljivanje dozvola uslugama ili metodama e-servisa, neovisno o samom načinu realizacije tih servisa.
- Sabirnica mora zabilježiti svaki zahtjev za realizaciju e-usluge ili pokretanje orkestracije kao i uspješnost te operacije. Za važnije usluge, mora se bilježiti puni ulazni zahtjev i cjelokupan odgovor.
- Sabirnica omogućava upravljanje poslovnim pravilima bez pisanja programskog koda. Aplikacije koje se povezuju preko sabirnice moraju koristiti odgovarajuće mehanizme sabirnice za provjeru poslovnih pravila prilikom integracije, a ne razvijati vlastite mehanizme
- Integracioni sistem mora raditi provjeru (validaciju) i mapiranje podatkovnih struktura poruka i dokumenata koji prolaze sabirnicom
- Sabirnica mora podržavati izvještavanje i nadzor izvršavanja poslovnih procesa u trenutku dok se odvijaju
- Za potrebe lakše i brže uspostave elektronske komunikacije s vanjskim subjektima katalog elektronskih usluga s njihovim opisima (WSDL datoteke) mora biti dostupan na korisničkom portalu sabirnice
- Sabirnica mora primjenjivati sigurnosne politike koje će definisati odgovarajuće načine autentikacije i druge sigurnosne mjere za pojedine kategorije korisnika.

	PROJEKT IZRADA I USPOSTAVLJANJE OKVIRA INTEROPERABILNOSTI I STANDARDA ZA RAZMJENU PODATAKA	Oznaka: PARCO-IFBIH
	<b>VODILJE I STANDARDI ZA          ARHITEKTURU SUSTAVA I RAZVOJ APLIKACIJA</b>	Page: 47 of 88

Sabirnica mora omogućavati sigurno i neometano izvršavanje poslovnih procesa bez mogućnosti neovlaštenih korisnika da utječu na njen rad ili steknu uvid u podatke.


- Sabirnica mora biti implementirana na način da se osigura visoka dostupnost koristeći odgovarajuća clustering rješenja (network load balancing za aplikativni sloj i multi-instance clustering za podatkovni sloj).
- Sabirnica mora omogućavati oporavak od greške bez gubitka poruka i/ili podataka i osigurati mehanizme ponovnog pokušaja isporuke poruke (durable messaging – korištenje message queue za kompenziranje dugotrajnog gubitka mrežne veze između aplikacionih sistema)
- BAM portal sabirnice mora prikazivati sve bitne parametre funkcioniranja sistema kao i propadanje (drill-down) do nivoa dijelova neke poruke ili cijele poruke.
- Sabirnica mora za komunikaciju koristiti standardne transportne protokole (http, https, mail, datotečni sistem, ftp, message queue) te web standarde koji podržavaju servisno orijentisanu arhitekturu (XML, XMLS Shema, SOAP, WSDL, WS-I, WS-Policy, WS-Addressin, WS-Security, WS-Reliable Messaging, WS-Atomic Transaction)

### 6.2.3 Nadzor rada integracijske platforme

Da bi se omogućio efektovit nadzor rada integracijske platforme i brojnih sistema s kojima je ona povezana, platforma mora sadržavati BAM portal.

BAM portal mora omogućavati personalizirane izvještaje za pojedine korisnike i/ili kategorije korisnika.

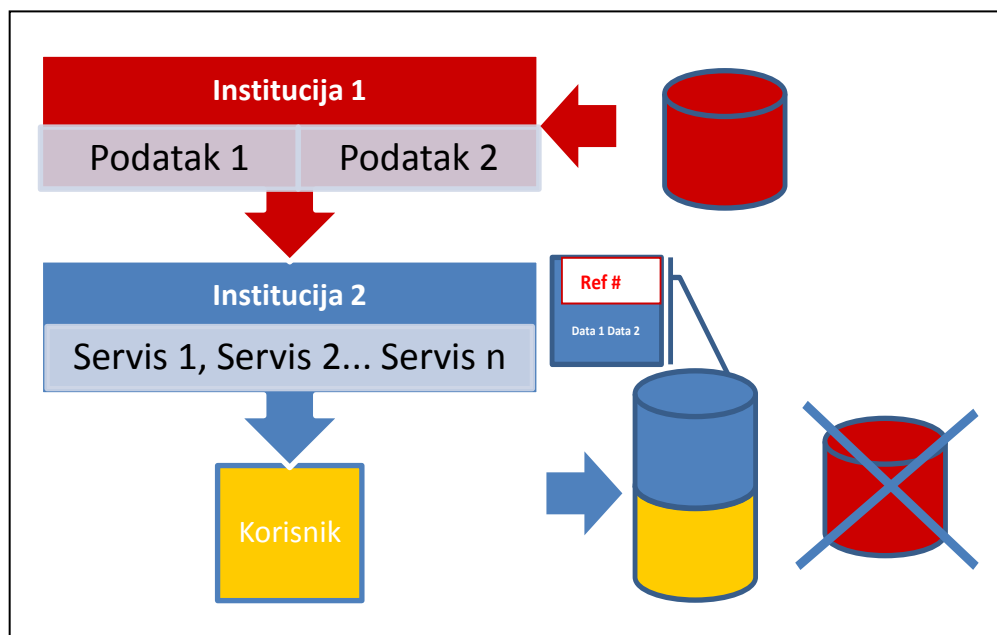
BAM portal mora prikupljati podatke kroz vrijeme i omogućavati mjerenje ostvarenja Ključnih pokazatelja uspješnosti (KPI) u određenom vremenskom periodu.

	PROJEKT IZRADA I USPOSTAVLJANJE OKVIRA INTEROPERABILNOSTI I STANDARDA ZA RAZMJENU PODATAKA  <b>VODILJE I STANDARDI ZA          ARHITEKTURU SUSTAVA I RAZVOJ APLIKACIJA</b>	Oznaka: PARCO-IFBIH
		Page: 48 of 88

## 6.3 Sigurnost i zaštita podataka

### 6.3.1 Nadležnosti i osiguranje zadržavanja vlasništva nad podacima pri realizaciji interoperabilnosti

Prilikom uspostavljanja mehanizama interoperabilnosti neophodno je jasno definisati da institucija, organo ili organ može i smije pohranjivati isključivo podatke u svom vlasništvu u skladu s odgovarajućim pravnim okvirom.




Slika 10 Nadležnosti i osiguranje zadržavanja vlasništva nad podacima

Tako, kada institucija 2 ima uvid u podatke institucije 1 (crveno), a koji su u skladu sa relevantnim zakonima neophodnim za realizaciju funkcije koju institucija 2 obavlja (plavo) prema korisniku informacije (žuto), Institucija 2 ima pravo pohranjivanja isključivo informacija koje predstavljaju vlasništvo Institucije 2, kao i reference na informacije institucije 1 koje omogućavaju identifikaciju i referenciranje na ulazne informacije. Informacije institucije 1 se ne smiju pohranjivati u okviru institucije 2 osim privremenog pohranjivanja tokom tog za procesiranje specifičnog, zahtjeva. Po završetku obrade iste moraju biti poništene uz zadržavanje reference na originalni izvor.

### 6.3.2 Sigurnost informatičkog sistema

Mehanizmi sigurnosti i zaštite podataka nisu u cjelosti tehničko pitanje, već obuhvaćaju politiku organizacije, standardne procedure poslovanja i pitanja procjene rizika. Planiranje i definisanje tih smjernica je preduslov za uspostavljanje sistema za upravljanje sigurnosti (ISMS - Information security management system).



	PROJEKT IZRADA I USPOSTAVLJANJE OKVIRA INTEROPERABILNOSTI I STANDARDA ZA RAZMJENU PODATAKA  <b>VODILJE I STANDARDI ZA          ARHITEKTURU SUSTAVA I RAZVOJ APLIKACIJA</b>	Oznaka: PARCO-IFBIH
		Page: 49 of 88

Kod odabira tehnologije potrebno je utvrditi sljedeće kategorije:

- Stepen povjerljivosti informacija u različitim fazama korištenja
- Integritet podataka kod upravljanja podacima i kod backupa
- Dostupnost podataka prema pravima pristupa
- Analiza rizika određene tehnologije

Prvi korak implementacije sigurnosnog sistema je planiranje o konkretnim parametrima poput algoritama za enkripciju podataka i komunikacije, dužine ključeva, određivanje certifikata i izdavatelja certifikata, način potpisivanja poruke i pravila autentikacije.

Informacioni sistemi moraju biti smješteni unutar različitih zona koje su jasno definisane i implementirane prema zahtjevima za sigurnost usluga i podataka koji se na njima pohranjuju ili obavljaju.

Sigurnosne zone moraju biti razdvojene kako bi se osigurali komunikacioni kanali. Za razdvajanje zona mora biti implementirana odgovarajuća infrastruktura gateway-a.

### 6.3.3 Sigurnost komunikacionih kanala

Komunikacija može postojati između dijelova sistema ili između sistema i vanjskog sistema. U oba slučaja je potrebno odabrati tehnologiju koja će zadovoljiti tražene uvjete u vezi integriteta podataka, sigurnosnih mehanizama i performansi.

Komunikacioni kanali trebaju biti klasificirani u ovisnosti o sigurnosnoj klasifikaciji podataka kojii se preko njih razmjenjuju. Nije dopušteno miješati komunikacione kanale koji su na različitim sigurnosnim nivoima.


Radi sprječavanja neovlaštenog korištenja sistema i podataka, na perimetrima sistema kao i na mjestima koja povezuju sisteme na različitom sigurnosnom nivou moraju se nalaziti usmjerivači (routeri) koji se brinu o tome da se promet usmjerava ispravnim putem i sprečavaju promet podataka iz mreža niže u mreže višeg sigurnosnog nivoa, te „firewall“ sistemi koji detektuju i onemogućavaju neovlašteni promet.

U klasifikovanim mrežama potrebna je također implementacija sistema za analizu mrežnog prometa i automatsku detekciju sigurnosno relevantnih događaja (*Intrusion detection/prevention system*).

### 6.3.4 Sigurnost i zaštita podataka na upravno-servisnoj sabirnici

Budući da se preko sabirnice razmjenjuju razni povjerljivi podaci, što uključuje i osobne podatke građana, sabirnica mora omogućavati visoki nivo sigurnosti i zaštite sadržaja prenesenih poruka.


Svi sistemi koji pristupaju sabirnici moraju biti propisno autentificirani i autorizirani, pri čemu se preferira autentifikacija digitalnim certifikatom, a mogu se (ovisno o sigurnosnim zahtjevima pojedinog procesa) dopustiti i neke druge manje sigurne autentifikacijske metode.

	PROJEKT IZRADA I USPOSTAVLJANJE OKVIRA INTEROPERABILNOSTI I STANDARDA ZA RAZMJENU PODATAKA	Oznaka: PARCO-IFBIH
	<b>VODILJE I STANDARDI ZA ARHITEKTURU SUSTAVA I RAZVOJ APLIKACIJA</b>	Page: 50 of 88

Sistem mora sadržavati centralizirani sistem autorizacije pristupa web-uslugama i web-servisima. Sistem mora omogućavati da, za svaki web servis, organo javne uprave - vlasnik servisa odredi osobe i organa koja su ovlaštena pristupiti pojedinom servisu i koja prava imaju na tom servisu (role-based autorizacija)

Sve poruke moraju biti potpisane digitalnim certifikatima da bi se osigurala nepromjenjivost sadržaja i autentičnost pošiljatelja te ostvarila neporecivost. Poruke, koje sadrže povjerljive (ili osobne) podatke, moraju podržavati i kriptiranje sadržaja.

Sistem mora bilježiti sadržaj, vrijeme slanja i prijema te status svake pojedine poruke / transakcije. Svaka poruka i log zapis mora nositi pouzdanu identifikaciju vremena slanja (timestamp).

	PROJEKT IZRADA I USPOSTAVLJANJE OKVIRA INTEROPERABILNOSTI I STANDARDA ZA RAZMJENU PODATAKA  <b>VODILJE I STANDARDI ZA          ARHITEKTURU SUSTAVA I RAZVOJ APLIKACIJA</b>	Oznaka: PARCO-IFBIH
		Page: 51 of 88

## 7 Tehnološki pogled - razvoj aplikacionih rješenja

### 7.1 Modeliranje i razvoj aplikacionog rješenja

Slično poslovnim procesima, modeliranje softverkog rješenja mora omogućiti različite vidove modela. Prednost se daje standardima u okrilju organizacije OMG (Object Management Group) koji poštuju principe MDA (Model Driven Architecture). Na primjer, UML omogućava sve potrebne vidove modeliranja i predstavlja preporučenu tehnologiju za isporuku razvojne dokumentacije.

Razvoj softverskog rješenja može obuhvatiti veliki broj tehnologija i pristupa programiranju. Ovisno o situaciji može se koristiti bilo koji pristup programiranju, ali se prednost daje:

- Objektno-orijentiranom za apstrakciju rješenja
- Proceduralnom za jednostavne komponente
- Deklarativnom za definisanje sučelja, podataka ili poslovnih pravila

Platforma za razvoj ovisi o postojećoj tehnologiji i o sloju na kojem se implementira:


- Za sučelje se preporuča korištenje web tehnologija, HTML i JavaScript. Posebnu pažnju treba obratiti na kriptiranje osjetljivih podataka koje putuju mrežom odn. koji ostaju na stroju.
- Za srednji sloj, konkretno, autentikacijske i autorizacijske funkcije, obradu podataka, razmjenu poruka, mapiranje objekata prema podacima, distribuirani objektni model i web-servise se preporuča korištenje Java EE tehnologije. U slučaju da se ne koristi distribuirani objektni model, može se koristiti Java SE tehnologija.
- Za integracije s Microsoftovim proizvodima se preporuča korištenje .NET Framework tehnologije.
- Za spremanje podataka se preporuča korištenje adekvatne baze za koju postoji odgovarajuće sučelje prema odabranoj tehnologiji u srednjem sloju (npr. JDBC).

### 7.2 Životni ciklus aplikacija

Životni ciklus aplikacija jest kontinuisani proces upravljanja različitim fazama u cjelokupnom životu softwera kroz automatizaciju procesa i integraciju informacija svake faze procesa. Životni ciklus aplikacije pokriva cjelokupnu historiju, od početnih ideja i zahtjeva, preko implementacije, isporuka, održavanja pa sve do trenutka kad sistem gubi svoju poslovnu vrijednost te se prestaje koristiti. Zbog svoje kompleksnosti životni ciklus aplikacije jest predmet kontinuisanog upravljanja, kontrole, evaluacije, razvoja i održavanja.

Tipične faze u životnom ciklusu aplikacije jesu:

- Upravljanje zahtjevima
- Dizajn i arhitektura
- Razvoj i testiranje

	PROJEKT IZRADA I USPOSTAVLJANJE OKVIRA INTEROPERABILNOSTI I STANDARDA ZA RAZMJENU PODATAKA  <b>VODILJE I STANDARDI ZA          ARHITEKTURU SUSTAVA I RAZVOJ APLIKACIJA</b>	Oznaka: PARCO-IFBIH
		Page: 52 of 88

- Deployment
- Održavanje
- Optimizacija

### 7.2.1 Upravljanje zahtjevima

Tokom faze upravljanja zahtjevima identificiraju se, dokumentiraju, prioritiziraju i prate korisničke potrebe. Sve korisničke zahtjeve je neophodno iskomunicirati i usaglasiti sa relevantnim sudionicima u životnom ciklusu aplikacija.

Upravljanje korisničkim zahtjevima je sistematski početak prikupljanja, dokumentiranja, organizovanja i slijeđenja zahtjeva nekog sistema koji se opisuju u obliku slučajeva korištenja. U ovoj fazi uočavaju se svi ključni slučajevi korištenja sistema, te se detaljno opisuju, a na osnovu njih će se graditi arhitektura budućeg sistema koja ih podržava. Dodatno, potrebno je opisati sve moguće poslovne procese i omogućiti njihovo izvođenje, te definisati slučajeve korištenja koji će predstavljati osnov za planiranje i provođenje testiranja sistema, gdje se provjerava da li sistem radi u skladu sa početno definisanim zahtjevima.

### 7.2.2 Dizajn i arhitektura

Dizajn i arhitektura su faze u kojima se zahtjevi mapiraju u arhitekturu u smislu organizovanja sistema u strukturne elemente, njihova sučelja i ponašanja na višem nivou (arhitektura) i na nižem detaljnom nivou (dizajn). Prilikom dizajna sistema potrebno je u najvećem mogućoj mjeri (uz poštovanje ograničenja koja postavlja javna nabavka) ostvariti kolaboraciju organa uprave i implementatora sistema na dizajnu sistema, kako bi se postigao jedan od ciljeva razvoja sistema e-Uprave a to je usmjerenost na korisnika (customer-centricity).


Definisani zahtjevi su polazište za dizajn i arhitekturu sistema. Svrha ove faze je da se analiziraju zahtjevi za sistemom i oblikuje arhitektura sistema koja će moći zadovoljiti prethodno definisane zahtjeve. U ovoj fazi najintenzivnije se koristi vizualno modeliranje sa UML-om. Arhitektura sistema se oblikuje kao serisno orjentirana arhitektura u skladu s poglavljem 7.5. ovoga dokumenta.

Centralni dio analize i oblikovanja je definisanje stabilne arhitekture sistema. Jaki naglasak na važnost arhitekture sistema i objektno orijentisani pristup razvoju sistema omogućuje da se sistem gradi od komponenata. Ovakav pristup skraćuje vrijeme razvoja sistema i omogućuje višestruko korištenje programskog koda (engl. *reusability*), te istovremeno smanjuje troškove razvoja.

### 7.2.3 Razvoj i testiranje

Tokom faze razvoja i testiranja, moduli i komponente se kodiraju, dokumentiraju, testiraju a identificirana odstupanja se ispravljaju. Krajnja isporuka ove faze jest sam softwareski proizvod, odnosno aplikacija.

Za vrijeme životnog ciklusa projekta voditelj projekta se služi nekom od standardnih metodologija za upravljanje projektima (na primjer PMI, PRINCE), a u samoj fazi razvoja

	PROJEKT IZRADA I USPOSTAVLJANJE OKVIRA INTEROPERABILNOSTI I STANDARDA ZA RAZMJENU PODATAKA  <b>VODILJE I STANDARDI ZA          ARHITEKTURU SUSTAVA I RAZVOJ APLIKACIJA</b>	Oznaka: PARCO-IFBIH
		Page: 53 of 88

aplikacija primjenjuje se: odgovarajuća metodologija (na primjer RUP, AIM); radni okvir (na primjer MSF) ili pristup razvoju aplikacija vođen principima (na primjer agilne metode).

Sa aspekta testiranja softverskih proizvoda potrebno je izvršiti testiranje:

1. Testiranje individualnih komponenti, programa i modula, kako bi utvrdili da zadovoljavaju pojedinačne specifikacije,
2. Testiranje rada programskih komponenti integrisanih u veće programske module,
3. Testiranje cijelog programskog sistema da bi utvrdili da sve komponente funkcioniraju zajedno upravo onako kako je to korisnik specificirao,

U konačnom testiranju korisnik vodi testiranje prihvatanja konkretnog produkta kako bi se uvjerio da on u cjelini zadovoljava njegove potrebe i zahtjeve, te potvrđuje se da su sve projektne isporuke završene prema specifikacijama te isporučene.

Proces testiranja provodi se kroz slijedeće faze:


1. Planiranje – izrada i prihvatanje testnog plana
2. Testni slučajevi – izrada i prihvatanje testnih slučajeva za funkcijsko testiranje pojedinih korisničkih slučajeva (user cases – UC)
3. Unit testovi – izrada i korištenje unit testova pri razvoju softvera
4. Funkcijsko testiranje – testiranje pojedinih UC-a prema odgovarajućim testnim slučajevima
5. Testiranje performansi – testiranje performansi kritičnih i po potrebi nekih važnih UC-a
6. Integralno testiranje – testiranje OSS sistema integrisanog s ostalim sistemima

Sistem se smatra prihvatljivim kada specificirani testni slučajevi (test cases - TC) (u skladu s UC-ovima i njihovim prioritetima navedenim u UC modelu) valjano i pravilno ispunjavaju postavljene zahtjeve korisnika te zajedno s ispunjenjem dogovorenih nefunkcionih zahtjeva (koji se odnose odnosno ne odnose na ponašanje sistema tokom rada kao i postojeća ograničenja) definišu kriterije prihvatljivosti sistema.

#### **7.2.4 Isporuka i puštanje u rad (deployment)**

U fazi isporuke i puštanja u rad planiraju se, izvode i kontroliraju aktivnosti koje imaju za cilj integraciju softvera u postojeću IT okolinu tako da postane dostupan za korištenje krajnjim korisnicima. Tipične aktivnosti u deploymentu su isporuka, instalacija, migracija, verzioniranje, adaptacija, update.

Poslovi isporučivanja sistema osiguravaju da je sistem pušten u rad kod korisnika. Ovisno o vrsti sistema ovaj postupak može trajati od nekoliko sati za jednostavne sisteme do nekoliko tjedana ili čak mjeseci za iznimno složene i velike sisteme. Prije isporuke sistema potrebno je detaljno ispitati korisnikovu okolinu, izraditi plan isporuke i ostale potrebne pripreme. Moguće su tri vrste isporuke: instalacija specifična za nekog korisnika, isporuka kao standardni programski proizvod i pristup proizvodu preko interneta.

	PROJEKT IZRADA I USPOSTAVLJANJE OKVIRA INTEROPERABILNOSTI I STANDARDA ZA RAZMJENU PODATAKA	Oznaka: PARCO-IFBIH
	<b>VODILJE I STANDARDI ZA          ARHITEKTURU SUSTAVA I RAZVOJ APLIKACIJA</b>	Page: 54 of 88

### 7.2.5 Održavanje

Održavanje aplikacije je životna faza u kojoj je software u toku praktične svakodnevne primjene. Održavanje obuhvaća nadgledanje aplikacije i rješavanje incidenata kad se pojave (usluge podrške produkcijskom radu sistema i usluge proaktivnog praćenja sistema). Ovo uključuje održavanje sistemskog krajolika, baze podataka, portala i aplikacione verzije i međuverzije.

Redoviti rad softvera nadgleda se i prati s ciljem preventivnog obavljanja svih potrebnih radnji kako bi softver uvijek optimalno i ispravno radio, a obavlja se prema prethodno dogovorenom planu. Rješavanje incidenata obavlja se prema preporukama iz poglavlja 8.1.

### 7.2.6 Optimizacija

Optimizacija je životna faza aplikacije koja uključuje poboljšanja, dodavanje komponenti i elemenata systemske infrastrukture, potrebne korekcije i podešavanje performansi. Potreba za optimizacijom može doći na inicijativu korisnika (poboljšanja, dogradnje...), identifikacijom problema na osnovu prijavljenih incidenata ili kao posljedica uočenih problema u radu softvera tokom redovitog praćenja rada sistema.


## 7.3 Nadogradnje i poboljšanja

Poslovni procesi unutar svake organizacije, pa tako i javne uprave, mijenjaju se s vremenom: organizacija mijenja ustroj i zadaće, dolazi do promjena modela upravljanja, razvijaju se modeli suradnje s drugim organima, a također i s vanjskim partnerima. Informacioni sistem treba pratiti sve te promjene, a koje skupno ulaze u kategoriju nadogradnji i poboljšanja.

Ovi procesi su u osnovi slični procesima kod razvoja novog informacionog sistema. Naime, nadogradnje i poboljšanja obično uključuju izradu novih elemenata informacionog sistema, različitog obima, ovisno o potrebama konkretne nadogradnje ili poboljšanja.

Međutim, nadogradnje i poboljšanja informacionog sistema imaju nekoliko suštinskih razlika u odnosu na poslove razvoja, a koja treba uzeti u obzir u praksi:

1. nadogradnje i poboljšanja se obavljaju unutar postojeće organizacije, postojeće infrastrukture i postojećeg informacionog sistema;
2. uobičajeno se promjene uvode „odozdo prema gore“, a u idealnom slučaju mijenja se manji dio informacionog sistema, kod većih promjena iznimno je bitno odrediti obim i mogući utjecaj promjena;
3. održavanje se uvijek odvija u „živoj“ organizaciji, dakle tokom obavljanja standardnih aktivnosti, zato je potrebno obratiti pažnju na moguće ometanje poslovnog sistema te potrebe i troškove dodatne edukacije, koja se mora na vrijeme najaviti i pripremiti;
4. rokovi za implementaciju nadogradnji i poboljšanja trebaju biti kratki i precizno definisani – za razliku od razvoja, u pravilu nije moguće održati informacioni sistem u radu tokom implementacije nadogradnji i poboljšanja; u praksi treba voditi računa da se iste obavljaju u vrijeme kada zahtjevi za informacionim sistemom ne postoje ili su minimalni (noću, vikendom i slično); a posebnu pažnju treba obratiti sistemima koji moraju biti aktivni 24 sata.

	PROJEKT IZRADA I USPOSTAVLJANJE OKVIRA INTEROPERABILNOSTI I STANDARDA ZA RAZMJENU PODATAKA  <b>VODILJE I STANDARDI ZA          ARHITEKTURU SUSTAVA I RAZVOJ APLIKACIJA</b>	Oznaka: PARCO-IFBIH
		Page: 55 of 88

Procesi nadogradnji i poboljšanja dijele se u pet faza:

1. analiza utjecaja (impact analysis)
2. dizajn (design)
3. realizacija (realisation)
4. testiranje (testing)
5. uvođenje u rad (implementation)

### 7.3.1 Analiza utjecaja

Analiza utjecaja je iznimno bitan proces kod nadogradnji i poboljšanja informacionog sistema. Potrebno je pažljivo pripremiti novu verziju, i ispuniti sve predulove kako bi se nadogradnje i poboljšanja sistema realizirali bez dodatnih poteškoća, u predviđenom vremenu i unutar predviđenih troškova.

Prije rada na modifikacijama postojećeg informacionog sistema potrebno je precizno definisati koje će modifikacije biti uključene u novu verziju, koji dijelovi informacionog sistema će time biti zahvaćeni, da li postoje alternative nadogradnji, koja rješenja će se za nadogradnju koristiti, i kada će vremenski modifikacija biti spremna. Ovi podaci će se koristiti za planiranje potrebnih kapaciteta, vremena i načina implementacije.

Analizu utjecaja ne treba ograničiti samo na „inženjerske“ aspekte. Potrebno je odrediti utjecaj na samu organizaciju, konkretan poslovni i informacioni okoliš same organizacije, te moguće dugoročne efekte. U to treba uključiti i procjene vremena i troškova. Na primjer, da li je novo rješenje finansijski opravdano sa stanovišta troškova. Da li je potrebna dodatna edukacija kadrova i slično. Tehnološki najnaprednija rješenja ponekad ne moraju biti optimalna kada se uračunaju svi troškovi.


### 7.3.2 Dizajn

Svrha faze dizajna je specificirati potrebne promjene aplikacija, kako bi informacioni sistem mogao funkcionirati u skladu s definisanim zahtjevima koji moraju uključivati zahtjeve povezane s pružanjem elektronskih usluga, kada je primjenjivo, odnosno povezivanja s drugim informacionim sistemima u svrhu preuzimanja, odnosno davanja podataka..

Dobar dizajn aplikacija ispunjava nekoliko zahtjeva:

1. pruža nedvosmislenu definiciju o informacionom sistemu, odnosno dijelu informacionog sistema koji će se mijenjati, kako bi se točno znao obim projekta;
2. uređuje komunikaciju s naručiteljem (nadležnima unutar organa javne uprave), kako bi bili upoznati koje modifikacije ili funkcionalnosti će biti obuhvaćene;
3. priprema dokumente potrebne za prihvatanje i puštanje u rad unaprijeđenog sistema, kako bi se izbjegli kasniji nesporazumi, ovo uključuje i preciziranje pravnog okvira dovršetka posla.

Organizacija OMG (Object Management Group) se bavi definisanjem standarda u procesu dizajna i modeliranja softvera. Smjernice korištenja tih standarda definisane su kao MDA (Model Driven Architecture). MDA razdvaja poslovnu i aplikacionu logiku od implementacijske platforme

	PROJEKT IZRADA I USPOSTAVLJANJE OKVIRA INTEROPERABILNOSTI I STANDARDA ZA RAZMJENU PODATAKA  <b>VODILJE I STANDARDI ZA          ARHITEKTURU SUSTAVA I RAZVOJ APLIKACIJA</b>	Oznaka: PARCO-IFBIH
		Page: 56 of 88

i tehnologije pa se postiže nivo apstrakcije sistema koja se može samostalno razvijati, nezavisno od implementacije. Ovaj način osigurava interoperabilnost i unutar i izvan granica sistema.

### 7.3.3 Realizacija

Realizacija može uključivati modifikaciju postojećih aplikacija, odnosno razvoj novih. Uz modifikaciju, odnosno izradu novih aplikacija potrebno je po potrebi prilagoditi i okoliš u kojem će se aplikacija koristiti (od tehničkih uvjeta do obuke službenika). Također, potrebno je prilagoditi i dokumentaciju, backup procedure te pravila informacione sigurnosti.

Potrebno je nadzirati kvalitetu rada pri realizaciji. Također, potrebno je pripremiti aplikaciju za testiranje, u skladu s nekom od priznatih metodologija testiranja softvera.

### 7.3.4 Testiranje

Upravljanje i razvoj aplikacija predstavlja radno intenzivan i kompleksan rad. Zato nije za očekivati da će informacioni sistem, pa i njegova nadogradnja bilo kojeg tipa, biti razvijeni potpuno bez pogrešaka („bugova“). Jedan jedini bug može značiti da aplikacioni softver, pa time i cijeli informacioni sistem neće pravilno ili uopće funkcionirati, što može imati ozbiljne posljedice. Zato isporučeni aplikacioni sistem ili bilo koja njegova nadogradnja moraju biti testirani. Ova faza je obavezna kod bilo kojih nadogradnji i poboljšanja informacionog sistema.

Faza testiranja osigurava da isporučeni softver i definicije podataka odgovaraju onome što je projektnom dokumentacijom trebalo biti isporučeno, u skladu sa specifikacijama koje su zadali korisnik i dizajneri informacionih sistema.


Proces testiranja mora biti strukturisan i precizno definisan. Potrebno je koristiti prihvaćene metodologije testiranja, alate i pristupe koji će jamčiti da će programski bugovi biti identifikovani i riješeni.

U praksi se zna dogoditi da isporučio softvera zbog kašnjenja prethodnih faza, postavljanja niskog prioriteta fazi skeniranja, nedostatka kvalitetnih stručnjaka za testiranje ili nekih drugih razloga skraćuju ili čak preskaču fazu testiranja, isporučujući nedovoljno testirane sisteme. Zato je potrebno fazu testiranja odgovarajuće ugovoriti i nadzirati. Testiranje mora biti dokumentirano i ta dokumentacija provjerena i prihvaćena od strane naručitelja.

Faza testiranja se dijeli u pet pod-faza:

1. programsko testiranje – testiranje novog ili promijenjenog softvera;
2. testiranje tehničkog sistema – testiranje cjelokupnog informacionog sistema uključujući hardver, mreže, napajanje električnom energijom i slično;
3. funkcionalne testove – testiranje da li nove verzije podržavaju tražene nove funkcionalnosti;
4. operacionalne testove – određivanje da li tehnička infrastruktura može podržati nove verzije i funkcionalnosti;
5. test prihvaćanja – u ovoj fazi osoba naručitelja odgovorna za procese koji su predmet rada aplikacija ili nadogradnji provjerava da li je proizvod u potpunosti ispušten u skladu sa svim specifikacijama.



	PROJEKT IZRADA I USPOSTAVLJANJE OKVIRA INTEROPERABILNOSTI I STANDARDA ZA RAZMJENU PODATAKA	Oznaka: PARCO-IFBIH
	<b>VODILJE I STANDARDI ZA          ARHITEKTURU SUSTAVA I RAZVOJ APLIKACIJA</b>	Page: 57 of 88

### 7.3.5 Uvođenje u rad

Završni stepen svake nadogradnje i poboljšanja informacionog sistema je uvođenje u rad. Ovaj proces uključuje sve aktivnosti potrebne za efektovito ostvarenje zahtjeva za promjenom. Cilj uvođenja u rad je pružiti uvjete za optimalno i operativno korištenje novih aplikacija, odnosno dijelova aplikacija, te dovršetak procesa unaprjeđenja.

Ovaj dio procesa primarno se fokusira na realizaciju zahtijevanih promjena u okruženju aplikacionog sistema. Aktivnosti su uglavnom usmjerene na podršku uvođenju (te usklađivanju tehničkog i poslovnog upravljanja) te kompletiranju svih zadataka vezanih za novu verziju aplikacionog sistema).

Uvođenje nove verzije ili modula aplikacije često traži promjene u okolišu informacionog sistema (tehničke – npr. prilagodbe prostorija za nove servere; poslovne – promjena podzakonskih akata i radnih uputa; kadrovske – edukacija korisnika).

Postoji mogućnost da je potrebno izvršiti i konverziju postojećih podataka. Proces konverzije treba biti precizno i unaprijed razrađen, sa svim nužnim koracima vezanim uz backup, očuvanje integriteta podataka, osiguranje prijenosa svih potrebnih podataka u novi sistem. Potrebno je izraditi i analizu rizika vezano za migraciju. Korisnici trebaju biti upoznati sa svim poteškoćama i zastojima koje postupak migracije može dovesti. Izvorni podaci moraju ostati sačuvani dogovoreno vrijeme nakon migracije. Ukoliko se radi migracija osobnih podataka, potrebno je ispuniti i preduslove vezane uz sistem zaštite osobnih podataka.

Ukoliko nove verzije mijenjaju strukturu podataka, ona mora biti detaljno obrazložena i sigurno pohranjena.

Ukoliko nove verzije mijenjaju radne procese, uključuju nove funkcionalnosti za krajnje korisnike ili mijenjaju izgled aplikacija potrebno je napisati nove verzije uputa za rad.

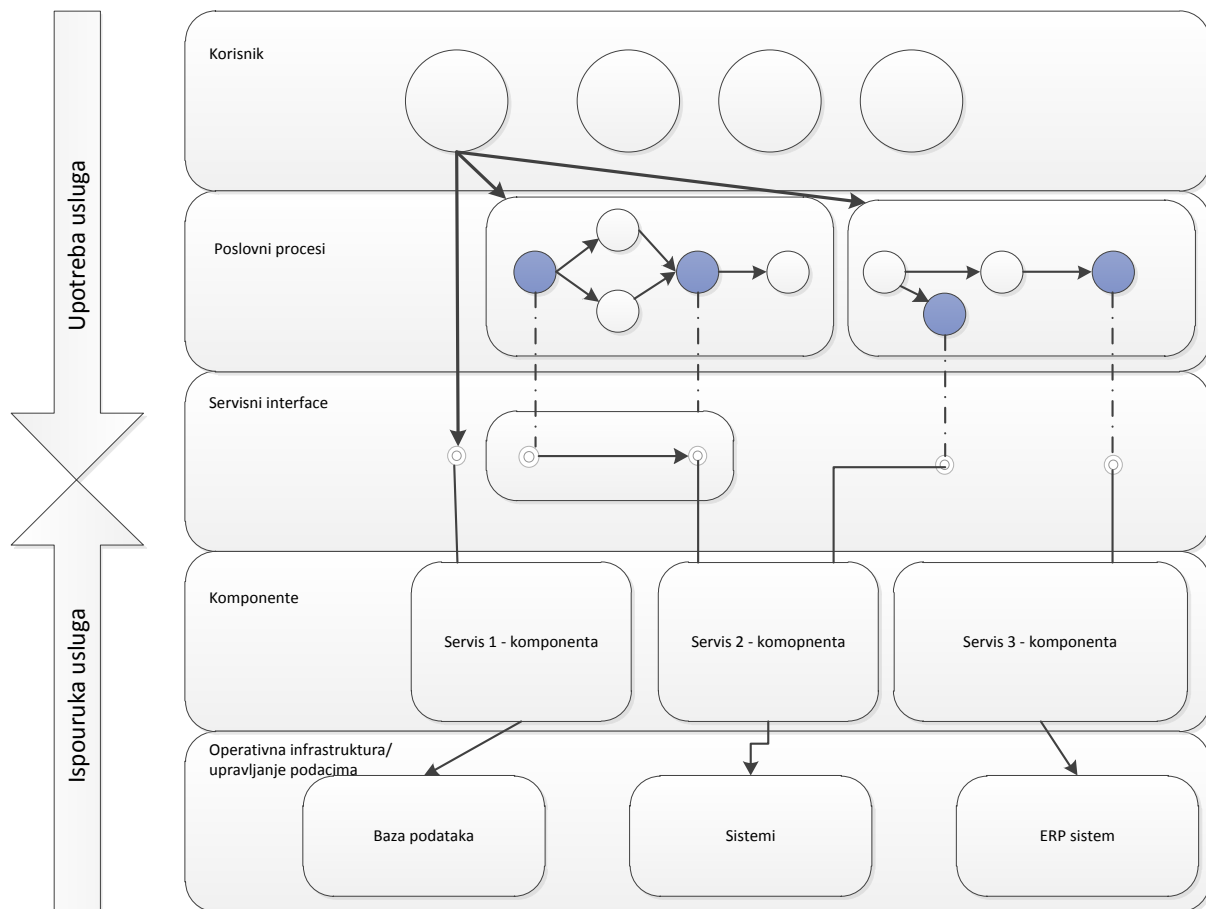
Osoblje zaduženo za održavanje te help-desk funkcije također mora biti na vrijeme obaviješteno o novim promjenama.

Potrebno je procijeniti i moguće dodatne troškove (prekovremeni rad, rad vikendom i noćni rad), s obzirom da se nadogradnje u pravilu uvode izvan redovnog uredovnog vremena. Kod većih nadogradnji potrebno je uzeti u obzir da će krajnji korisnici nekoliko dana možda imati smanjenu radnu produktivnost, dok se ne naviknu na nov način rada.

## 7.4 Načela implementacije servisno orijentisane arhitekture u aplikativnim sistemima

U SOA arhitekturi aplikacije se razvijaju kao skup softverskih servisa tako da svaki ima sučelje za uslugu preko kojeg korisnici usluga koriste njihove funkcionalnosti.

Na slijedećoj slici dan je prikaz referentnog SOA modela.

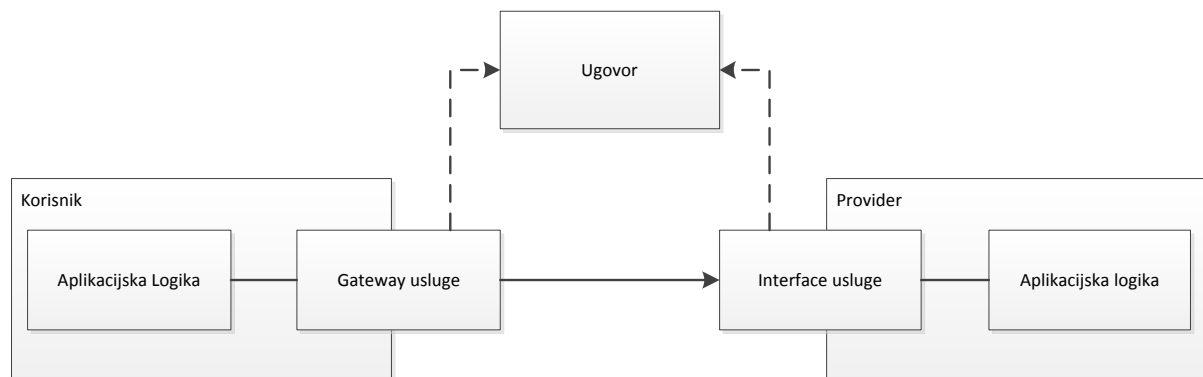


**Slika 11 SOA referentni model<sup>8</sup>**

Servis je diskretna jedinica aplikacione logike koja nudi sučelje bazirano na konceptu razmjene poruka i kojem pristupaju druge aplikacije. Svaki servis ima povezano sučelje koje predstavlja korisnicima usluge. Sučelje definiše i implementira ugovor između korisnika i pružatelja usluge. Ovaj ugovor i njegovu implementaciju nazivamo sučeljem servisa.

Slijedeća slika prikazuje korisnika usluge kako koristi uslugu providera ponuđenu preko sučelja usluge. Kolaboracija između ova dva elementa regulirana je ugovorom.

<sup>8</sup> Prema SAGA 4.0, str. 69, Slika 6.2, <http://www.cio.bund.de/saga>



Slika 12 Elementi usluge<sup>9</sup>

Radi integracije aplikacija na logičko-poslovnom nivou potrebno je omogućiti da povezani sistemi mogu primiti i pružati XML bazirane Web servise. Pri tome se koriste Web Services Description Language ugovori (WDSL contracts) za opis sučelja prema tim sistemima. Radi osiguravanja interoperabilnosti važno je osigurati implementaciju prema specifikacijama za Web usluge (kao npr. Web Services Security specifikacija).

Interakcije između sistema realizuje se tako da oni komuniciraju na način da koriste Web servise, prilikom čega aplikacija koja postavlja zahtjev kreira XML dokument u obliku poruke i šalje ga prema mreži pružatelja Web usluge. Opcionalno pružatelj šalje odgovor podnosiocu zahtjeva u obliku XML dokumenta. Standardi za Web servise specificiraju detalje za sučelje na koje se šalje poruka, format poruke, mapiranje sadržaja poruke na implementaciju usluge, opcionalna zaglavlja i način na koji se usluge publiciraju i na koji mogu biti pronađene od strane drugih Web servisa.


#### 7.4.1 Sučelja servisa

*Sučelje usluge* predstavlja krajnju tačku koju korisnici usluge koriste radi pristupa funkcionalnosti ponuđene aplikacije. Sučelje servisa se obično može referencirati preko mrežne adrese tako da joj korisnici mogu pristupiti preko komunikacione mreže. Ove adrese mogu biti javno poznate lokacije ili se mogu dosegnuti kroz imeničke servise poput UDDI.

Ključni aspekt dizajna servisnog sučelja je *razdvajanje* implementacionih elemenata potrebnih za komunikaciju s drugim sistemima od poslovne logike implementacije. Sučelje usluge pruža puno robusnije sučelje zadržavajući pri tome semantiku i finu granularnost aplikacione logike. Također predstavlja barijeru koja omogućava izmjenu aplikacione logike bez utjecaja na korisnike sučelja usluge.

<sup>9</sup> Prema: The Code Project, .NET/Java Interoperability using Service Interface and DTO Architecture Patterns

<http://www.codeproject.com/KB/architecture/SerInterfaceXXsd2Code.aspx>

	PROJEKT IZRADA I USPOSTAVLJANJE OKVIRA INTEROPERABILNOSTI I STANDARDA ZA RAZMJENU PODATAKA	Oznaka: PARCO-IFBIH
	<b>VODILJE I STANDARDI ZA          ARHITEKTURU SUSTAVA I RAZVOJ APLIKACIJA</b>	Page: 60 of 88

Sučelje servisa implementira ugovor između korisnika i pružatelja usluge. Ugovor omogućava razmjenu informacija bez obzira na sisteme na kojima se nalaze. Sučelje usluge mora osigurati implementaciju svih detalja potrebnih za komunikaciju. Implementacioni detalji koje je između ostalog potrebno osigurati jesu:

- **Mrežni protokol.** Sučelje servisa mora obuhvaćati sve aspekte mrežnih protokola koji se koriste za komunikaciju između korisnika i pružatelja usluge. Implementacija usluge ima vlastiti ugovor sa sučeljem usluge i nebi trebala imati veze s detaljima tehnologije koju korisnici usluge koriste za komunikaciju sa sučeljem.
- **Formati podataka.** Sučelje servisa obavlja pretvorbu između formata koje koriste korisnički podaci i formata podataka koje usluga očekuje. Implementacija usluge ne treba imati veze s specifičnostima podatkovnih formata koje usluga koristi pri komunikaciji s korisnicima servisa.
- **Sigurnost.** Servisno sučelje predstavlja vlastitu granicu što se tiče sigurnosti. Različiti korisnici imaju različite zahtjeve glede sigurnosti tako da se isti implementiraju kroz specifične sigurnosne zahtjeve na sučelje usluge.
- **Ugovori o nivou usluge (SLA).** Sučelje usluge ima važnu ulogu u osiguravanju tog da usluga bude sukladna obavezama o nivou usluge za korisnike. Servisna sučelja mogu imati cache kako bi se poboljšalo vrijeme odziva i potrošnja bandwidtha. Također je moguće implementirati više instanci servisnih sučelja te na taj način napraviti load-balancing procesnih node-ova kako bi se udovoljilo zahtjevima za skalabilnost, raspoloživost i fault tolerance.


#### 7.4.2 Eksplicitne granice servisa

Usluzi prilikom njenog pozivanja treba proslijediti sve što je potrebno za osiguravanje njene funkcionalnosti. Pristup servisu mora biti isključivo preko njenog javnog sučelja, bez ikakvih pretpostavki ili drugih preduslova za njeno funkcionisanje. Poziv servisa u pravilu ne smije ovisiti o dijeljenom kontekstu; umjesto toga svi se pozivi usluge moraju modelirati kao state-less. Sučelje koje prezentira servis opisano je ugovorom koji opisuje njegove funkcionalne i nefunkcionalne mogućnosti i karakteristike. Poziv servisa je akcija koja ima poslovni efekt, može potencijalno biti skupa u smislu trošenja resursa i uvodi kategoriju grešaka koje su različite od slučaja kad pozivamo lokalne metode ili RPC pozive. Poziv servisa nije poziv udaljenih procedura.

Ukratko, usluga predstavlja svoju funkcionalnost kroz eksplicitno sučelje koje obavlja i izolira interne detalje funkcionalnosti; interakcija s uslugom je eksplicitna radnja koja se realizuje kroz razmjenu poruka između korisnika i pružatelja usluge.

#### 7.4.3 Dijeljenje servisnih opisa i shema a ne klasa

Polazeći od opisa usluga (ugovora) i korisnik i pružatelj usluge moraju imati na raspolaganju sve što im je potrebno za njenu upotrebu. U skladu s principom labavog povezivanja pružatelj usluge nemože računati na to da će korisnik usluge ponovno iskoristiti kod koji on pruža, u svojoj vlastitoj okolini. Ovaj princip jasno ograničava tipove podataka koji se mogu razmjenjivati kroz

	PROJEKT IZRADA I USPOSTAVLJANJE OKVIRA INTEROPERABILNOSTI I STANDARDA ZA RAZMJENU PODATAKA	Oznaka: PARCO-IFBIH
	<b>VODILJE I STANDARDI ZA          ARHITEKTURU SUSTAVA I RAZVOJ APLIKACIJA</b>	Page: 61 of 88

SOA. U najboljem slučaju podaci se razmjenjuju kao XML dokumenti koje se može validirati u odnosu jedne ili više shema jer je to podržano u bilo kojoj od programerskih okolina. Slijedno tome, ovo isključuje upotrebu DCOM i RMI baziranih okolina kao opcija za SOA.

#### 7.4.4 Upravljanje pomoću politika

Kako bi se servis mogao koristiti potrebno je ispuniti dva preduslova:

1. Funkcionalnost, sintaksa i semantika pružatelja usluge moraju biti usklađene s zahtjevima korisnika usluge
2. Usklađenost tehničkih mogućnosti i potreba

Zbog nastojanja da se osigura pristup usluzi od strane što većeg broja različito opremljenih korisnika usluge potrebno je da se kao dio SOA seta alata uvedu odgovarajući mehanizmi implementacije politika. Dok su funkcionalni aspekti usluge opisani u njegovom sučelju, ortogonalni, ne-funkcionalni zahtjevi i potrebe se specificiraju u okviru politika.

#### 7.4.5 Autonomija servisa


U skladu s principom eksplicitnih granica servisa, servis je autonoman u tom smislu što je njegova jedina veza s vanjskim svijetom – barem što se tiče SOA perspektive – kroz njegovo sučelje. Konkretno mora biti moguće promijeniti runtime okolinu servisa, npr. iz prototipske okoline koja je aktivna na razvojnoj stanici u punu produkcijsku okolinu koja se oslanja na serverski baziranoj skupini kolaborirajućih komponenti, bez ikakvog utjecaja na korisnike. Mora biti moguće mijenjati i implementirati usluge, te mijenjati njihove verzije neovisno jednu od druge. Pružatelj usluga ne smije podrazumijevati to da su korisnici njegovih usluga prilagodljivi na nove verzije usluge; neki od njih ne moraju bili sposobni ili ne moraju uopće planirati prihvaćanje nove verzije sučelja usluge.

#### 7.4.6 Korištenje formata za prijenos a ne programskih API-eva

Usluge su ponuđene na upotrebu preko komunikacije u formatima koji moraju biti podržani od svih sudionika komunikacije. Ovaj princip uvodi dodatnu perspektivu na aspekt komunikacije: Kako bi se osigurala dostupnost i trajna iskoristivost usluge usluga mora biti dohvatljiva s bilo koje platforme koja podržava razmjenu poruka usklađenih sa servisnim sučeljem, sve dok je ta komunikacija usklađena s politikama usluge.

Kriteriji za verifikaciju ovog principa mogu biti:

- Formatu poruka opisani su korištenjem otvorenih standarda ili u čitljivom obliku
- Moguće je kreirati poruke prema definisanim shemama uz razuman napor bez potrebe za određenom programerskom bibliotekom
- Semantika i sintaksa dodatnih informacija potrebnih za uspješnu komunikaciju, poput headera koja se koristi zbog sigurnosti ili pouzdanosti usklađena je s nekom od javnih specifikacija ili standarda
- Barem jedan od transportnih protokola koji se koriste za povezivanje sa servisom je neki od standardnih mrežnih protokola

	PROJEKT IZRADA I USPOSTAVLJANJE OKVIRA INTEROPERABILNOSTI I STANDARDA ZA RAZMJENU PODATAKA	Oznaka: PARCO-IFBIH
	<b>VODILJE I STANDARDI ZA          ARHITEKTURU SUSTAVA I RAZVOJ APLIKACIJA</b>	Page: 62 of 88

#### 7.4.7 Orjentacija na elektronske dokumente

Prilikom razmjene podataka pomoću servisa podaci se prosljeđuju u formi elektronskih dokumenata. Ti dokumenti su eksplicitno modelirani korištenjem postojećih zajedničkih podatkovnih komponenti iz rječnika podataka, kao hijerarhioni kontejneri za podatke. Jedan od važnih aspekata orjentacije na elektronske dokumente je njihova samo-dokumentovanost (korištenjem odgovarajućih alata, a u svrhu objave sučelja za preuzimanje/razmjenu dokumenata). Idealno elektronski dokumenti se formiraju kao dokumenti koje koristimo u stvarnom svijetu, poput narudžbenica, računa ili izvještaja o stanju računa. Dokumente je potrebno dizajnirati na taj način da budu korisni u okviru predmetne domene, te da ih koristi jedan ili više servisa.

Kao i kod dokumenata u papirnatom obliku razmjenjeni elektronski dokumenti uključuju i redundantne informacije kao npr. ID korisnika uz koji se razmjenjuju podaci o adresi korisnika (iako bi bilo dovoljno razmjeniti samo ID korisnika). Ovakva redundancija se prihvaća iz razloga što ona omogućava izolaciju sučelja usluge od podatkovnog modela pružatelja i korisnika usluge. Kod korištenja na dokument orijentisanih stilova, pozivi servisa postaju smislene razmjene poslovnih poruka, za razliku od kontekstno neovisnih RPC poziva. Iako nije obavezujuće, obično se pretpostavlja da se kao format/sintaksa dokumenta koristi XML.


Poruke koje se razmjenjuju između sudionika u SOA služe za povezivanje različitih sistema koji se mogu razvijati jedan neovisno od drugog. Princip labavog povezivanja nalaže da se zavisnost o zajedničkim informacijama i znanju minimizira koliko je to god moguće. Kad se poruke šalje kroz Distribuirane objekte ili RPC infrastrukturu, klijent i poslužioc se oslanjaju na skup proxy klasa (stub-ovi i skeletoni) koji se generiraju iz istog dokumenta s opisom sučelja. U suprotnom slučaju obustavlja se komunikacija jer se pretpostavlja da ugovor ne podržava komunikaciju između ove dvije strane. Iz istog razloga RPC infrastruktura zahtjeva sinkroniziran razvoj klijentskog i serverskog programskog koda.

#### 7.4.8 Labavo vezanje sistema

Većina pobornika SOA arhitekture složiti će se kako je labavo vezanje vrlo važan koncept. No problem je što postoje različita mišljenja o karakteristikama koje sistem određuju kao „labavo vezan“. Postoje mnogi aspekti i načini na koje sistem može biti čvrsto ili labavo vezan i u ovisnosti od konteksta i zahtjeva na njega se može gledati kao čvrsto vezanog s jednog aspekta a labavo s drugog.

Aspekti mogu biti:

- **Vrijeme:** Labavo vezani sudionici nemoraju stalno biti aktivni kako bi mogli komunicirati. To se realizuje formiranjem neke vrsta reda između njih. Slanje poruke od jednog sudionika do drugog ne uvjetuje trenutčan odgovor kako bi se nastavilo s procesiranjem (niti logički, niti fizički).
- **Lokacija:** U slučaju da sudionici postavje upit za adresom drugog sudionika, lokacije se mogu mijenjati bez potrebe za ponovnim programiranjem, rekonfiguracijom ili restartom partnera u komunikaciji. Ovo podrazumijava neku vrstu lookup procesa korištenjem imenika ili adresara koji sadrži adrese krajnjih točaka usluga.

	PROJEKT IZRADA I USPOSTAVLJANJE OKVIRA INTEROPERABILNOSTI I STANDARDA ZA RAZMJENU PODATAKA	Oznaka: PARCO-IFBIH
	<b>VODILJE I STANDARDI ZA          ARHITEKTURU SUSTAVA I RAZVOJ APLIKACIJA</b>	Page: 63 of 88

- Tip: Analogno konceptu statičnog vs. dinamičnog i slabog vs. jakog tipiziranja u programskim jezicima sudionik se može osloniti na cjelokupnu ili na dio strukture dokumenta kako bi obavio željenu radnju.
- Verzija: Sudionici mogu ovisiti o određenoj verziji sučelja usluge ili mogu biti potpuno neosjetljivi na promjene (do određene mjere). Što više sudionici ovise o konkretnijoj verziji to su manje labavo vezani (u odnosu na promatrani aspekt). Kod toga se dobro drži principa Postelovog Zakona: Provider usluga treba implementirati tako da prihvate što je više moguće raznih verzija poruka, te da budu liberalni u tome što prihvaćaju (i po mogućnosti tolerantni na greške), dok korisnici servisa moraju nastojati da budu usklađeni s točnom gramatikom i tipovima dokumenta. Na ovaj način se povećava stabilnost i fleksibilnost sistema.
- Kardinalnost: Između korisnika i providera usluge može se uspostaviti 1:1 odnos naročito u slučajevima zahtjev/odgovor interakcije ili gdje se koristi eksplicitni red za poruke. U drugim slučajevima korisnik usluge (koji je u ovom slučaju „pošiljalac poruke“ ili „izvor događaja“) ne mora znati niti brinuti o broju primatelja poruke.
- Lookup: Sudionik koji namjerava koristiti uslugu može ili koristiti ime (fizičko ili logičko) pružatelja usluge s kojim komunicira ili može napraviti operaciju upita korištenjem opisa karakteristika usluga umjesto toga. To podrazumijeva da postoji registar i/ili repozitorij koji zna mapirati usluge između zahtjeva korisnika i karakteristika pružatelja usluga (bilo direktno ili indirektno).
- Sučelje: Sudionici mogu zahtijevati usklađenost sa specifičnim sučeljima za usluge ili mogu podržavati korištenje generičkih sučelja. U slučaju da se koristi generičko sučelje, svi korisnici koji koriste isto sučelje mogu međusobno komunicirati. Na primjer, princip jednog generičkog sučelja je osnov WWW arhitekture.


Najčešće nije moguće a niti poželjno kreirati labavo vezane sisteme kroz sve navedene aspekte.

#### 7.4.9 Usklađenost s standardima

Ključni princip koji moramo poštovati prilikom primjene SOA koncepta je oslanjanje na standarde namjesto specifičnih API-a i formata. Standardizacija se primjenjuje kako na tehničke aspekte poput podatkovnih formata, metapodataka, transportnih i protokola za prijenos, tako i na artefakte poslovnog nivoa poput tipizacije dokumenata (npr. kod UBL-a).

#### 7.4.10 Nezavisnost od proizvođača

Nijedan od koncepata vezanih na arhitekturu sistema ne smije biti povezan s nekim konkretnim produktom nekog proizvođača. Prilikom uspostavljanja sistema i odabira konkretnih rješenja proizvođača, izabrana rješenja ni u kojem slučaju ne smiju imati utjecaj na samu arhitekturu. To podrazumijeva da se koriste standardi za interoperabilnost kao i za portabilnost u najboljoj mogućoj mjeri. Na taj se način može implementirati pružatelj ili korisnik usluge koristeći bilo koju od tehnologija koje podržavaju odgovarajuće standarde neovisno o proizvođaču.

	PROJEKT IZRADA I USPOSTAVLJANJE OKVIRA INTEROPERABILNOSTI I STANDARDA ZA RAZMJENU PODATAKA	Oznaka: PARCO-IFBIH
	<b>VODILJE I STANDARDI ZA          ARHITEKTURU SUSTAVA I RAZVOJ APLIKACIJA</b>	Page: 64 of 88

#### 7.4.11 Upravljanje meta-podacima

Sve metadata artefakte u okviru SOA-e potrebno je pohranjivati na takav način da ih je lako pronalaziti, uzimati i interpretirati i prilikom dizajna kao i prilikom rada sistema. Artefakti su opisi sučelja usluga, sudionici, krajnje tačke i informacije za povezivanje, organizacijske jedinice, role i odgovornosti, tipovi/scheme dokumenata, korisnik/provider odnosi i dr. Koliko je moguće, potrebno je automatizirati korištenje tih artefakata, bilo korištenjem generatora ili interpretatora koda kako bi postali dio životnog ciklusa usluge i sudionika procesa.

### 7.5 SOA Governance

SOA Governance je proširenje IT upravljanja, koja je produžetak korporativnog upravljanja. SOA Governance implementira kontrolu životnog ciklusa usluga i složenih aplikacija u organizaciji koja koristi uslužno orijentisane arhitekture (SOA).

Funkcija SOA Governance je prvenstveno:

- Donijeti odluku o razvoju, implementaciji, poslovanju i upravljanju novim uslugama
- Nadzor i izvješćivanje na osnovu odluke i postignutih rezultata

SOA Governance pruža Business/IT usklađivanje kako bi se postigle SOA postavljene vrijednosti: ponovna upotreba usluga i unapređenje poslovne agilnosti. Dvije ključne komponente SOA Governance uključuje usluge upravljanja i organizacijske promjene.

Efektivita SOA Governance obuhvaća ljude, procese i tehnologije uključene u cijeli životni ciklus iz SOA poslovne perspektive, dakle usklađivanja poslovanja s njom.

SOA Governance Referentni Model (SGRM) je osnovni generički model koji se koristi kao podloga za prilagodbu SOA governance modela u skladu s potrebama organizacije kroz upravljani proces njegove prilagodbe.


Osnovni dijelovi modela su:

- SOA Governance principi
- SOA Governance procesi
- SOA Governance procesi kojima se upravlja
- SOA Governance artefakti
- SOA Governance role i odgovornosti
- SOA Governance tehnologija

#### 7.5.1 Definicija SOA Governancea

SOA governance je primjena upravljanja poslovnom, IT i EA governancea na Servisno orijentiranu arhitekturu (SOA). Kao takva predstavlja proširenje IT i EA governancea u nastojanju



	PROJEKT IZRADA I USPOSTAVLJANJE OKVIRA INTEROPERABILNOSTI I STANDARDA ZA RAZMJENU PODATAKA  <b>VODILJE I STANDARDI ZA          ARHITEKTURU SUSTAVA I RAZVOJ APLIKACIJA</b>	Oznaka: PARCO-IFBIH
		Page: 65 of 88

da se iskoriste prednosti SOA arhitekture, te pokriva sve aspekte iskorištavanja SOA arhitekture, od strateškog pa sve do operativnog nivoa.

### 7.5.2 Obim SOA Governancea

Obim primjene SOA governancea treba postaviti tako da ne obuhvaća samo tehnološki nego sve aspekte primjene SOA tehnologije. Tako da mora obuhvaćati u istoj mjeri ljudski faktor, procese i tehnološke aspekte primjene i governance-a SOA-e.

### 7.5.3 SOA Governance Framework

SOA Governance Framework omogućava definiciju vlastitog SOA modela governancea prilagođenog organizaciji i njenim potrebama. Pristup primjeni modela je u fazama kako bi se minimizirali svi rizici pri njegovoj primjeni i maksimalno iskoristile kvalitete koje on pruža.

Framework se sastoji od referentnog modela (SOA Governance Referentni model – SGRM) i metode za njegovu primjenu – SOA Governance Vitality Metoda (SGVM).


#### 7.5.3.1 SOA Governance Reference Model (SGRM)

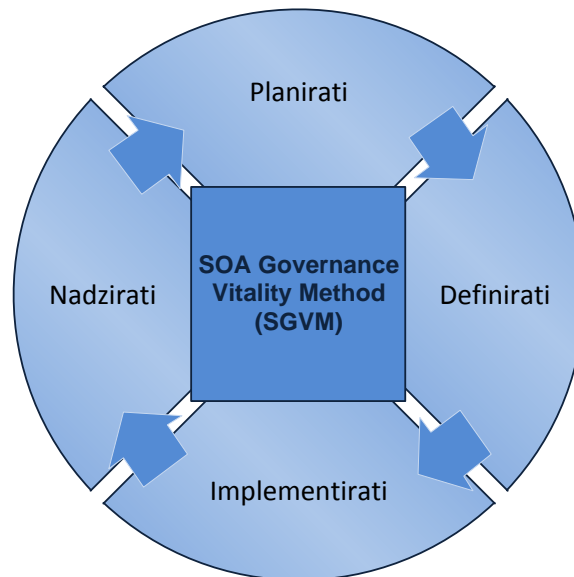
SOA Governance Model (SGRM) je generički model koji služi kao podloga za analitiku i daljnju prilagodbu modela upravljanja SOA usklađenosti prema potrebama organizacije. Prilagodbu modela je potrebno napraviti za sve aspekte SGRM modela.

#### 7.5.3.2 SOA Governance Vitality Method (SGVM)

SOA Governance Vitality Method (SGVM) je zapravo proces koji se koristi radi prilagodbe SGRM modela prema potrebama organizacije. Na SGVM je potrebno gledati kao proces a ne kao projekt jer je potrebno osigurati kontinuisano provođenje aktivnosti SGVM u smislu implementacije kontinuisanih aktivnosti usavršavanja pri čemu se kontinuisano prati rad sistema, uspoređuju se stvarne s ciljanim metrikama i poduzimaju odgovarajuće korektivne aktivnosti kad je to potrebno.

SOA Governance Vitality Metoda (SGVM) je proces koji koristi SOA Governance Referentni Model (SGRM) kao osnovni model SOA Governance-a, te kroz niz faza i aktivnosti prilagođava osnovni model specifičnostima organizacije u koju se uvodi. Implementaciju SOA Governance-a treba promatrati kao proces a ne projekt. Faze SGVM treba promatrati kao kontinuisani iterativne aktivnosti u kojima se između ostalog mjeri napredak i rade korekcije smjera kojim se provodi SOA Governance-a. Metoda se koristi također prema potrebi za unaprjeđenje trenutnog implementiranog SOA Governance režima i planova daljnjeg razvoja SOA Governance-a (SOA Governance Roadmap).

	PROJEKT IZRADA I USPOSTAVLJANJE OKVIRA INTEROPERABILNOSTI I STANDARDA ZA RAZMJENU PODATAKA <b>VODILJE I STANDARDI ZA          ARHITEKTURU SUSTAVA I RAZVOJ APLIKACIJA</b>	Oznaka: PARCO-IFBIH
		Page: 66 of 88




Slika 13. Faze metode prilagodbe SOA referentnog modela<sup>10</sup>

Faze SGVM-a prikazane su:

- **Faza Planiranja.** U ovoj fazi identificiraju se i analiziraju osnovne oblasti upravljanja na kojima su moguća poboljšanja. Uspostavljaju se ciljevi, tj. planovi i specifične mjere za predloženi inkrementalni skup poboljšanja modela. Prijašnja implementirana poboljšanja se također evaluiraju u cilju korekcija istih.
- **Faza Definisanja.** U fazi definisanja izrađuje se SOA Governance Plan Prijelaza na osnovu kojeg će se ostvariti ciljevi definisani u fazi planiranja.
- **Faza Implementacije.** Na osnovu Plana Prijelaza u ovoj fazi se implementiraju potrebni procesi, organizacijske strukture i tehnologije SOA Governance modela.
- **Faza Nadzora.** Provodi se nadzor i mjerenje efektivnosti implementiranog SOA Governance modela. Mjerenjem efektivnosti provjerava se da li implementirani model dostiže predhodno definisane ciljeve. Ova faza može pokrenuti još jedan iteracioni ciklus SGVM-a.

<sup>10</sup> Prema: The Open Group: <http://www.opengroup.org/soa/source-book/gov/sgvm.htm>


	<p>PROJEKT IZRADA I USPOSTAVLJANJE OKVIRA INTEROPERABILNOSTI I STANDARDA ZA RAZMJENU PODATAKA</p>	<p>Oznaka: PARCO-IFBIH</p>
	<p><b>VODILJE I STANDARDI ZA ARHITEKTURU SUSTAVA I RAZVOJ APLIKACIJA</b></p>	<p>Page: 67 of 88</p>

#### 7.5.4 SOA Governance Tehnologija

SOA Governance tehnologija koristi se kako bi se omogućio governance te djelomična ili cjelokupna automatizacija governance procesa. Tehnologije koje se koriste mogu biti od najjednostavnije do najsloženijih.

SOA Governance procesi između ostalog koriste slijedeće tehnološke mogućnosti:

- sistemi za pohranu i upravljanje pristupom podacima – radi pohrane i omogućavanje dostupa artefaktima procesa
- sistemi za osiguravanje provođenja politika – kao podrška i mogućnost automatiziranja nadzora povreda nastalih odstupanjem od politika, kako prilikom dizajna tako i prilikom operative
- sistemi za nadzor – prikupljanje i obrada metrika koje su ključni pokazaoi rada sistema governancea; mogu pokrenuti procese provjere usklađenosti
- sistemi za upravljanje – omogućavanje upravljanja promjenama, upravljanja konfiguracijama, upravljanje sigurnošću; mogu se koristiti radi implementacije i održavanja governancea
- sistemi za upravljanje radnim zadacima – radi automatiziranja procesa poput procesa za osiguravanje usklađenosti ili upravljanja izuzecima od usklađenosti

	PROJEKT IZRADA I USPOSTAVLJANJE OKVIRA INTEROPERABILNOSTI I STANDARDA ZA RAZMJENU PODATAKA	Oznaka: PARCO-IFBIH
	<b>VODILJE I STANDARDI ZA          ARHITEKTURU SUSTAVA I RAZVOJ APLIKACIJA</b>	Page: 68 of 88

## 8 Upravljanje uslugama (eng. service management), sigurnost i drugi nefunkcionalni zahtjevi

Održavanje sistema na dnevnom nivou iznimno je važan segment upravljanja informacionim sistemom. Krajnji korisnici često upravo na osnovu kvaliteta održavanja i podrške sistema stiču dojam o kvalitetu cijele organizacije. Potrebno je razviti takve sisteme koji će ispuniti očekivanja krajnjih korisnika, a što uključuje i brzo i efektovito otklanjanje zastoja u radu aplikacionog sistema.

Održavanje informacionog sistema zasniva se na konceptu upravljanja informacionom uslugom, a sadrži pet procesa:

- upravljanje incidentima (*incident management*)
- upravljanje konfiguracijama (*configuration management*)
- upravljanje dostupnošću (*availability management*)
- upravljanje kapacitetom (*capacity management*)
- upravljanje kontinuitetom poslovanja (*continuity management*)

### 8.1 Upravljanje incidentima


Bitno pitanje u procesu upravljanja incidentima je komunikacija s korisničkom organizacijom. Važno je napomenuti da proces upravljanja incidentima ponekad izlazi iz okvira upravljanja aplikacijama: on uključuje i upravljanje tehničkom infrastrukturom te poslovnim informacijama.

Kao primjer specifičnosti upravljanja incidentima možemo navesti niz problema koji se mogu doći izvan aplikacionog sistema. Na primjer tu spadaju fizički kvarovi u računalnoj mreži, prekidi u napajanju električnom energijom i slično. Krajnji korisnik će dobar dio takvih zastoja prijavljivati kao zastoj aplikacionog sistema.

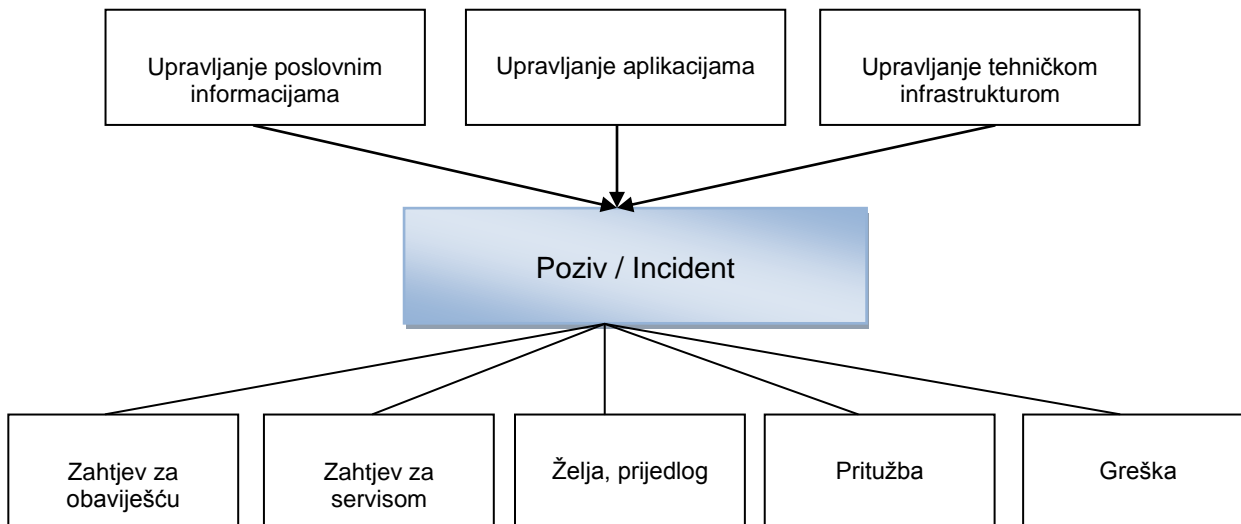
Komunikacija kod upravljanja rizicima dijeli se na reaktivnu (unatrag) i proaktivnu (unaprijed).

Reaktivna komunikacija primarno predstavlja odgovore na „pozive“ i „incidente“. Incident predstavlja svaki pitanje, zahtjev, žalba, prijava pogreške, koja dolazi od strane korisnika i krajnjeg korisnika. Primjedbe također mogu dolaziti od strane osoba zaduženih za upravljanje aplikacijama (na primjer prijava da sistem radi bitno sporije od očekivanog) te od osoba zaduženih za upravljanje tehničkom infrastrukturom (prijava da je sistem prestao funkcionirati).

Upravljanje incidentima mora obuhvatiti efektovitu koordinaciju svih oblika održavanja, i treba biti razrađeno ko prima pozive, kome se prosljeđuju i kako se izvještava o provedenom.

	PROJEKT IZRADA I USPOSTAVLJANJE OKVIRA INTEROPERABILNOSTI I STANDARDA ZA RAZMJENU PODATAKA  <b>VODILJE I STANDARDI ZA          ARHITEKTURU SUSTAVA I RAZVOJ APLIKACIJA</b>	Oznaka: PARCO-IFBIH
		Page: 69 of 88

Upravljanje incidentima:



Drugi oblik komunikacije jest proaktivna komunikacija. Primjer takve komunikacije, a koju treba razraditi i primijeniti u praksi je upoznavanje s aplikacionim sistemom novih službenika. Kvalitetno upoznavanje i obuka povećati će efektivnost rada službenika i smanjiti broj poziva službi za korisnike (otklanjanje uzroka zastoja u radu aplikacione programske opreme koje se obavlja po prijavi zastoja ili neispravnosti u radu).

## 8.2 Upravljanje konfiguracijama


Procesi vezani za upravljanje konfiguracijama daju odgovore na pitanja koje aplikacije se pokreću na kojoj infrastrukturi. U upravljanje konfiguracijama također ulaze i vođenje zapisa vezanih za rad, brzinu i usklađenost infrastrukture i aplikacija.

Upravljanje konfiguracijama je posebno važno ukoliko se infrastruktura fizički ne nalazi na jednom mjesto. Također, potrebno je raditi zapise koja verzija aplikacija je instalirana na kojoj lokaciji, koje su njene komponente i točne tehničke karakteristike infrastrukture svake lokacije. Ove su informacije neophodne kako bi se moglo adekvatno odgovoriti na probleme i upite.

Kod upravljanja konfiguracijama potrebno je dužnu pažnju posvetiti i ugovornoj strani održavanja. Pri tome posebno treba uzeti u obzir da različite lokacije i različite aplikacije na istoj tehničkoj infrastrukturi mogu zahtijevati različite ugovore o održavanju, s više privrednih subjekata. Potrebno je obratiti pažnju na usklađenost svih subjekata i ugovornih obaveza.

## 8.3 Upravljanje dostupnošću

Upravljanje dostupnošću se zasniva na dva kvalitativna parametra: dostupnošću informacionog sistema i njegovom pouzdanosti.

	PROJEKT IZRADA I USPOSTAVLJANJE OKVIRA INTEROPERABILNOSTI I STANDARDA ZA RAZMJENU PODATAKA	Oznaka: PARCO-IFBIH
	<b>VODILJE I STANDARDI ZA          ARHITEKTURU SUSTAVA I RAZVOJ APLIKACIJA</b>	Page: 70 of 88

Dostupnost znači da aplikacije moraju pružati tražene funkcionalnosti tokom zadanog perioda. Ovo se odnosi na sve faze rada aplikacije: pokretanje informacionog sistema, odvijanje operacija u traženom vremenu i funkcionalnosti, uključivanje ad-hoc dodatnih zadataka i procesa, te pohranjivanje rezultata. Korisnici trebaju biti u mogućnosti koristiti informacionih sistem, obrađivati informacije te obavljati druge poslove za koje je informacionih sistem predviđen unutar zadanih parametara brzine rada sistema.

Pouzdanost znači da aplikacije ili komponente aplikacija trebaju pružati odgovarajuću funkcionalnost tokom točno određenog vremenskog razdoblja. Nije dovoljno da je informacioni sistem dostupan, on mora i pravilno raditi. Broj pogrešaka, prekida (npr. blokiranja, resetiranja, potreba za ponovnim pokretanjem programa) može imati veliki utjecaj na kvalitetu rada i prihvaćenost aplikacije kod krajnjeg korisnika. Potrebno je pratiti ovakve pojave i uvesti odgovarajuće kontrole kako bi se ovakve neželjene pojave mogle mjeriti i na vrijeme eskalirati zaduženima za održavanje sistema.

Kod uspostave help-desk funkcionalnosti potrebno je osigurati odgovarajući kapacitet, ali i radno vrijeme (a posebno uzeti u obzir da neke službe javne uprave rade 24/7/365).

## 8.4 Upravljanje kapacitetima

Unatoč velikom povećanju kapaciteta informacionih sistema, pitanje upravljanja kapacitetima i dalje je iznimno važno. Nove potrebe korisnika i nove aplikacije kontinuisano traže veće kapacitete informacionih sistema.


Često se pitanje performansi i kapaciteta informacionog sistema veže uz upravljanje tehničkom infrastrukturom, ali takav pogled na pitanje upravljanja kapacitetima je preuzak. Pitanje upravljanja kapacitetima neminovno ulazi u oblast upravljanja aplikacijama.

Performanse informacionog sistema uvelike su određene dizajnom softvera i programskim zahtjevima. Dakle, puštanje u rad novih verzija softvera, ili novih programskim modula može tražiti dodatne kapacitete. Zato ove aktivnosti moraju biti usklađene.

U sistemima koji obrađuju velike količine podataka, a gdje u pravilu ulaze sistemi javne uprave, može se dogoditi da količina podataka u informacionom sistemu prijeđe onu za koju je informacioni sistem prvotno projektiran, što dovodi do neželjenih usporavanja i drugih zastoja u radu. Ove promjene potrebno je mjeriti i nadzirati, kako bi se na vrijeme izvršile potrebne nadogradnje.

Prilikom projektiranja potrebno je uzeti u obzir da u mnogim djelatnostima javnog sektora postoje razdoblja iznimno intenzivne upotrebe informacionog sistema (npr. u školstvu je to doba upisa učenika i zaključivanja ocjena), te informacioni sistem treba biti predviđen za takva vršna opterećenja.

Nadalje, zbog potreba planiranja na godišnjem nivou (državni, entitetski, kantonalni budžeti) te nabavki opreme u skladu s procedurama postupka javne nabavke pitanje upravljanja kapacitetima je u javnom sektoru čak i važnije nego u privredi. Novi kapaciteti se obično ne mogu nabaviti „na brzinu“, kada se za njih pojavi neposredna potreba. Zato je preaktivno djelovanje i planiranje od iznimne važnosti.

	PROJEKT IZRADA I USPOSTAVLJANJE OKVIRA INTEROPERABILNOSTI I STANDARDA ZA RAZMJENU PODATAKA	Oznaka: PARCO-IFBIH
	<b>VODILJE I STANDARDI ZA          ARHITEKTURU SUSTAVA I RAZVOJ APLIKACIJA</b>	Page: 71 of 88

## 8.5 Upravljanje kontinuitetom

Uredan rad javne uprave danas uvelike ovisi o informacionim sistemima. Zato je zaštita kako od bilo kojih oblika izvanrednih događanja tako i neautoriziranog pristupa od iznimne važnosti.

Česta pogreška je ocijeniti stanje sa: „sve će biti u redu“ ili „do sada nismo imali problema, pa nećemo ni u budućnosti“. Takvo shvaćanje dolazi iz vremena kada je informatizacija uglavnom bila predstavljena u obliku PC-računala koja su ponajviše služila kao sofisticirane pisaće mašine potpuno su neprimjerena sadašnjem vremenu, kada se bez informacionog sistema ne mogu obavljati poslovi za koje je javni sektor zadužen.

Potrebno je osigurati da informacioni sistem radi kontinuisano, bez ometanja, ili s prihvatljivim rizikom vanjskih utjecaja.


Informacioni sistem može biti izložen vanjskim prijetnjama, na primjer pokušajima neautoriziranog pristupa. Potrebno je uspostaviti sve potrebne mjere fizičke sigurnosti i sigurnosti informacionog sistema. Preporuča se primjena normi i smjernica standarda informacione sigurnosti ISO 27001.

Prilikom uspostave mjera zaštite, treba uzeti u obzir da su zloporabe, na primjer neautorizirani pristup mogući i unutar sistema, od strane službenika same organizacije. Javni služba upravlja velikim brojem važnih informacija, koje uključuju osobne podatke, strateške planove, tokove novca i slično. Sistemi se moraju zaštititi od mogućih neprimjerenih, zlonamjernih ili zabranjenih korištenja (na primjer zapisivanjem pristupa, zapisivanjem pretraživanja i slično).

Aplikacioni sistem može biti ugrožen i eksternim prijetnjama tehničkoj infrastrukturi: dugotrajni prekidi napajanja električnom energijom, poplave, potresi i drugi razlozi mogu otežati, pa i onemogućiti korištenje aplikacionog sistema. Aplikacionih sistem je ovisan o tehničkoj infrastrukturi i ne može se prihvatiti stav da je tehnička infrastruktura „data“, na način da ne postoje planovi rada aplikacija u slučaju ozbiljnih vanjskih utjecaja na tehničku infrastrukturu. Zato je potrebna suradnja u planiranju upravljanja kontinuitetom aplikacija i tehničke infrastrukture, na primjer usklađivanje uključivanja pričuvnih lokacija.

Potrebno je uvesti mjere čuvanja izvornog koda. Ukoliko javna uprava nije vlasnik koda, preporuča se ugovoriti njegovu pohranu kod treće osobe (npr. odvjetnički ured), te mogućnost daljnjeg korištenja u slučaju da vlasnik koda više nije u mogućnosti pružati usluge vezane za održavanje. Politika čuvanja izvornog koda i pitanja vlasništva nad njime moraju biti jasni i dokumentirani.

Potrebno je pratiti razvoj aplikacija te u skladu s planovima migracije isključivati zastarjele sisteme koji se više ne koriste. Umanjen nadzor na tim sistemima i omogućena komunikaciona infrastruktura (npr. „otvoreni“ portovi za komunikaciju takvih aplikacija) ozbiljan su sigurnosni rizik.

	PROJEKT IZRADA I USPOSTAVLJANJE OKVIRA INTEROPERABILNOSTI I STANDARDA ZA RAZMJENU PODATAKA  <b>VODILJE I STANDARDI ZA          ARHITEKTURU SUSTAVA I RAZVOJ APLIKACIJA</b>	Oznaka: PARCO-IFBIH
		Page: 72 of 88

## 9 Zaključak

U ovom dokumentu predloženi su koncepti, pravila i standardi za arhitekturu informacionih sistema i razvoj aplikacija koji će omogućiti ostvarivanje strateških ciljeva definisanih projektom.

Standardi i preporuke za dizajn arhitekture informacionih sistema i razvoj aplikacija bazirani su na nekoliko različitih pogleda i modela arhitekture informacionih sistema.

Strateški pogled definiše strateške ciljeve koje se želi postići definisanjem vodilja i standarda za arhitekturu sistema i razvoj aplikacija.

Procesni pogled definiše način modeliranja poslovnih procesa, kao preduslov izgradnji bilo kojeg informatičkog sistema.

Funkcionalni pogled na arhitekturu informacionog sistema definiše izradu specifikacije potrebnih funkcionalnosti sistema te odabir njegove arhitekture.

Podatkovni pogled sadrži preporuke za modeliranje, pohranu i razmjenu podataka u informacionom sistemu.

Integracioni pogled bavi se problemima povezivanja različitih aplikacija i drugih komponenti informatičkog sistema u jedinstvenu cjelinu te načinima sigurnog pristupa i razmjene podataka između informacionih sistema.

Preporuka za korištenje servisnih sabirnica u integracionoj arhitekturi osigurava da su servisne tačke za razmjenu podataka dostupne svima koji podatke trebaju razmjenjivati i neovisne o aplikacijama u kojima su pohranjeni ti podaci i formatima razmjene podataka (pa time osigurava i da je moguća prilagodba drugoj strani čak i ako ona raspolaže skromnim informatičkim resursima).


Servisna sabirnica kao arhitekturni uzorak (pattern) rješava centralno osiguranje poštovanja poslovnih pravila (ko se smije razmjenjivati podatke) na nivou organa državne uprave bez potrebe da se za to mijenjaju pojedine aplikacije, integraciju pojedinih aplikacija za implementaciju kompleksnih procesa, transformaciju podataka ako je potrebna, autentikaciju i autorizaciju i logiranje na jednom mjestu, što sve pojednostavljuje i pojeftinjuje same aplikacije.

Tehnološki pogled definiše modeliranje i razvoj aplikacionih rješenja u različitim scenarijima: od vlastitog razvoja aplikacija, kupovine i implementacije gotovih proizvoda do kombinovanih pristupa.


Poglavlje „Upravljanje uslugama (eng. service management), sigurnost i drugi nefunkcionalni zahtjevi“ daje preporuke za upravljanje životnim ciklusom informacionih sistema i njihovog održavanje na dnevnom nivou.

Prilog 1. ovog dokumenta je Katalog tehničkih standarda, koji definiše tehničke standarde i preporučenu oblast njihove primjene.



	<p>PROJEKT IZRADA I USPOSTAVLJANJE OKVIRA INTEROPERABILNOSTI I STANDARDA ZA RAZMJENU PODATAKA</p> <p><b>VODILJE I STANDARDI ZA ARHITEKTURU SUSTAVA I RAZVOJ APLIKACIJA</b></p>	Oznaka: PARCO-IFBIH
		Page: 73 of 88


Kako se standardi mijenjaju s vremenom, prilog je potrebno periodični osvježavati. Preporuka je da se osvježavanje kataloga tehničkih standarda radi na godišnjoj osnovi.

	PROJEKT IZRADA I USPOSTAVLJANJE OKVIRA INTEROPERABILNOSTI I STANDARDA ZA RAZMJENU PODATAKA  <b>VODILJE I STANDARDI ZA          ARHITEKTURU SUSTAVA I RAZVOJ APLIKACIJA</b>	Oznaka: PARCO-IFBIH
		Page: 74 of 88


## Prilog 1. Katalog tehnoloških standarda

Sljedeća tablica određuje tehnološke standarde i na koji način će se koristiti.


Sigurnosni mehanizmi		
<b>Sistemi upravljanja za informacionu sigurnost</b>	<b>Preporuka:</b>	BSI standard 100-1: Management systems for Information Security  <a href="https://www.bsi.bund.de/cae/servlet/contentblob/471428/publicationFile/28221/standard_100-1_e_pdf.pdf">https://www.bsi.bund.de/cae/servlet/contentblob/471428/publicationFile/28221/standard_100-1_e_pdf.pdf</a>
<b>IT osnovni pristup zaštiti</b>	<b>Preporuka:</b>	BSI standard 100-2: IT baseline protection approach]  <a href="https://www.bsi.bund.de/cae/servlet/contentblob/471430/publicationFile/28223/standard_100-2_e_pdf.pdf">https://www.bsi.bund.de/cae/servlet/contentblob/471430/publicationFile/28223/standard_100-2_e_pdf.pdf</a>
<b>Analiza rizika</b>	<b>Preporuka:</b>	BSI-Standard 100-3: Risk analysis on the basis of IT baseline protection  <a href="https://www.bsi.bund.de/cae/servlet/contentblob/471432/publicationFile/28219/standard_100-3_e_pdf.pdf">https://www.bsi.bund.de/cae/servlet/contentblob/471432/publicationFile/28219/standard_100-3_e_pdf.pdf</a>
<b>Implementacija sigurnosnog koncepta</b>	<b>Preporuka:</b>	1. Industrial Signature Interoperability Specification – MailTrust (ISIS-MTT) v1.1.  <a href="http://www.t7-isis.de/uploads/media/ISIS-MTT_Core_Specification_v1.1.pdf">http://www.t7-isis.de/uploads/media/ISIS-MTT_Core_Specification_v1.1.pdf</a> 2. IT Baseline Protection Catalogues  <a href="https://www.bsi.bund.de/EN/Topics/ITGrundschutz/ITGrundschutzCatalogues/itgrundschutzcatalogues_node.html">https://www.bsi.bund.de/EN/Topics/ITGrundschutz/ITGrundschutzCatalogues/itgrundschutzcatalogues_node.html</a> 3. KoopA ADV, Guideline for the Introduction of the Electronic Signature and Encryption in the Administration v1.1 4. BSI, eGovernment manual
Modeli poslovnih procesa		
<b>Modeliranje poslovnih procesa</b>	<b>Obavezno:</b>	<ul style="list-style-type: none"> <li>• UML (<a href="http://www.omg.org/spec/UML">http://www.omg.org/spec/UML</a>)</li> <li>• BPMN (<a href="http://www.omg.org/spec/BPMN/2.0">http://www.omg.org/spec/BPMN/2.0</a>)</li> </ul>

	PROJEKT IZRADA I USPOSTAVLJANJE OKVIRA INTEROPERABILNOSTI I STANDARDA ZA RAZMJENU PODATAKA  <b>VODILJE I STANDARDI ZA          ARHITEKTURU SUSTAVA I RAZVOJ APLIKACIJA</b>	Oznaka: PARCO-IFBIH
		Page: 75 of 88


	<b>Preporuka:</b> <ul style="list-style-type: none"> <li>EPC (<a href="http://www.softwareag.com/corporate/products/aris_platform/modeling/default.asp">http://www.softwareag.com/corporate/products/aris_platform/modeling/default.asp</a>, <a href="http://en.wikipedia.org/wiki/Event-driven_process_chain">http://en.wikipedia.org/wiki/Event-driven_process_chain</a>)</li> <li>XPDL (<a href="http://www.wfmc.org/index.php?option=com_docman&amp;task=doc_download&amp;Itemid=72&amp;gid=132">http://www.wfmc.org/index.php?option=com_docman&amp;task=doc_download&amp;Itemid=72&amp;gid=132</a>)</li> </ul>
<b>Jezici za izvršavanje poslovnih procesa</b>	<b>Preporuka:</b> <ul style="list-style-type: none"> <li>BPMN (<a href="http://www.omg.org/spec/BPMN/2.0">http://www.omg.org/spec/BPMN/2.0</a>)</li> <li>WS-BPEL (<a href="http://docs.oasis-open.org/wsbpel/2.0/OS/wsbpel-v2.0-OS.html">http://docs.oasis-open.org/wsbpel/2.0/OS/wsbpel-v2.0-OS.html</a>)</li> <li>XPDL (<a href="http://www.wfmc.org/index.php?option=com_docman&amp;task=doc_download&amp;Itemid=72&amp;gid=132">http://www.wfmc.org/index.php?option=com_docman&amp;task=doc_download&amp;Itemid=72&amp;gid=132</a>)</li> </ul>
<b>Procesni modeli</b>	
<b>Tehnologije za procesno modeliranje</b>	<b>Obavezno:</b> <ol style="list-style-type: none"> <li>Role models and flow charts</li> <li>Unified Modeling Language (UML) v.2.0 <a href="http://www.omg.org/spec/UML/">http://www.omg.org/spec/UML/</a></li> </ol>
<b>Formati razmjene za procesne modele</b>	<b>Preporučeno:</b>  XML Metadata Interchange (XMI) v2.x  <a href="http://www.omg.org/spec/XMI/">http://www.omg.org/spec/XMI/</a>
<b>Podatkovni modeli</b>	
<b>Tehnologije za podatkovno modeliranje</b>	<b>Obavezno:</b> <ul style="list-style-type: none"> <li>Entity Relationship Diagram (ERD)</li> <li>Unified Modeling Language (UML) v. 2.0 <a href="http://www.omg.org/spec/UML">http://www.omg.org/spec/UML</a></li> </ul>
<b>Formati razmjene za podatkovne modele</b>	<b>Obavezno:</b> <ol style="list-style-type: none"> <li>XML Schema Definition (XSD) v1.0 <a href="http://www.w3.org/XML/Schema">http://www.w3.org/XML/Schema</a></li> </ol> <b>Preporučeno:</b> <ol style="list-style-type: none"> <li>Regular Language Description for XML New Generation (Relax NG)  <a href="http://www.relaxng.org/">http://www.relaxng.org/</a></li> <li>XML Metadata Interchange (XMI) v2.x <a href="http://www.omg.org/spec/XMI">http://www.omg.org/spec/XMI</a></li> </ol>

	PROJEKT IZRADA I USPOSTAVLJANJE OKVIRA INTEROPERABILNOSTI I STANDARDA ZA RAZMJENU PODATAKA  <b>VODILJE I STANDARDI ZA          ARHITEKTURU SUSTAVA I RAZVOJ APLIKACIJA</b>	Oznaka: PARCO-IFBIH
		Page: 76 of 88


<b>Opisni jezik za metapodatke datoteka</b>	<b>Preporučeno:</b> 1. Resource Description Framework (RDF)  <a href="http://www.w3.org/TR/rdf-primer/">http://www.w3.org/TR/rdf-primer/</a> 2. Dublin Core (DC)  <a href="http://dublincore.org/documents/dcmi-terms/">http://dublincore.org/documents/dcmi-terms/</a>
<b>Aplikativna arhitektura</b>	
<b>Aplikativna arhitektura sa middleware-om</b>	<b>Preporuka:</b> 1. Java Platform, Enterprise Edition (Java EE) v5 2. Java Platform, Standard Edition (Java SE) v5  <b>Potencijalno:</b> 3. .NET Framework (C#)/Common Language Infrastructure (CLI) 4. Business Process Execution Language for Web Services (BPEL4WS) v1.1 5. Web Services Business Process Execution Language (WS-BPEL) v2.0 <a href="http://docs.oasis-open.org/wsbpel/2.0/wsbpel-v2.0.html">http://docs.oasis-open.org/wsbpel/2.0/wsbpel-v2.0.html</a>
<b>Aplikativna arhitektura bez middleware-a</b>	<b>Preporučeno:</b>  PHP Hypertext Preprocessor (PHP) v5.x
<b>Klijent</b>	
<b>Pristup informacijama pomoću računara</b>	<b>Preporuka:</b>  Java Network Launching Protocol (JNLP) v1.5  <a href="http://www.jcp.org/en/jsr/detail?id=56">http://www.jcp.org/en/jsr/detail?id=56</a>
<b>Pristup informacijama pomoću mobilnih uređaja</b>	HTML, CSS ( <a href="http://www.w3.org/TR/NOTE-html40-mobile">http://www.w3.org/TR/NOTE-html40-mobile</a> )
<b>Pristup informacijama pomoću eksternih sistema</b>	XML via SOAP
<b>Tehnologije za autentifikaciju</b>	<b>Preporuka:</b> 1. Security Assertion Markup Language (SAML) v2.0  <a href="http://www.oasis-open.org/standards#samlv2.0">http://www.oasis-open.org/standards#samlv2.0</a>  <b>Potencijalno:</b> 2. Kerberos v5
<b>Prezentacija</b>	

	PROJEKT IZRADA I USPOSTAVLJANJE OKVIRA INTEROPERABILNOSTI I STANDARDA ZA RAZMJENU PODATAKA  <b>VODILJE I STANDARDI ZA          ARHITEKTURU SUSTAVA I RAZVOJ APLIKACIJA</b>	Oznaka: PARCO-IFBIH
		Page: 77 of 88


<b>Prezentacija bez ograničenja</b>	<b>Preporuka:</b> 1. Web Content Accessibility Guidelines  <a href="http://www.w3.org/TR/#tr_Web_Content_Accessibility_Guidelines_WCAG">http://www.w3.org/TR/#tr_Web_Content_Accessibility_Guidelines_WCAG</a>
<b>Setovi karaktera</b>	<b>Obavezno:</b> 1. Unicode v4.x UTF-8  <b>Preporučeno:</b> 2. Unicode v4.x UTF-16  <a href="http://www.unicode.org/">http://www.unicode.org/</a>
<b>Tehnologije za informaciono procesiranje</b>	<b>Obavezno:</b> 1. Hypertext Markup Language (HTML) v4.01 <a href="http://www.w3.org/TR/html401/">http://www.w3.org/TR/html401/</a> 2. Multipurpose Internet Mail Extensions (MIME) v1.0 <a href="http://www.iana.org/assignments/media-types/index.html">http://www.iana.org/assignments/media-types/index.html</a> 3. Java Servlet v2.5 <a href="http://www.jcp.org/en/jsr/detail?id=154">http://www.jcp.org/en/jsr/detail?id=154</a> 4. Java Server Pages (JSP) v2.1 <a href="http://www.jcp.org/en/jsr/detail?id=245">http://www.jcp.org/en/jsr/detail?id=245</a>  <b>Preporuka:</b> 5. Extensible Hypertext Markup Language (XHTML) v1.0 <a href="http://www.w3.org/TR/xhtml1/">http://www.w3.org/TR/xhtml1/</a> 6. Cascading Style Sheets Language Level 2 (CSS2) <a href="http://www.w3.org/TR/CSS2/">http://www.w3.org/TR/CSS2/</a> 7. Extensible Stylesheet Language (XSL) v1.0 <a href="http://www.w3.org/TR/xsl/">http://www.w3.org/TR/xsl/</a> 8. Extensible Stylesheet Language Transformations (XSLT) v1.0 <a href="http://www.w3.org/TR/xslt">http://www.w3.org/TR/xslt</a>  <b>Potencijalno:</b> 9. Extensible Stylesheet Language (XSL) v1.1 <a href="http://www.w3.org/TR/xsl11/">http://www.w3.org/TR/xsl11/</a> 10. Extensible Stylesheet Language Transformations (XSLT) v2.0 <a href="http://www.w3.org/TR/xslt20/">http://www.w3.org/TR/xslt20/</a>
<b>Aktivni sadržaji</b>	<b>Obavezno:</b>  ECMAScript Language Specification  <a href="http://www.ecma-international.org/publications/standards/Ecma-262.htm">http://www.ecma-international.org/publications/standards/Ecma-262.htm</a>
<b>Definicija formi za unos</b>	<b>Potencijalno:</b>

	PROJEKT IZRADA I USPOSTAVLJANJE OKVIRA INTEROPERABILNOSTI I STANDARDA ZA RAZMJENU PODATAKA  <b>VODILJE I STANDARDI ZA          ARHITEKTURU SUSTAVA I RAZVOJ APLIKACIJA</b>	Oznaka: PARCO-IFBIH
		Page: 78 of 88

	XForms v1.0  <a href="http://www.w3.org/TR/xforms/">http://www.w3.org/TR/xforms/</a>
<b>Formati za razmjenu podataka</b>	<b>Obavezno:</b> 1. Extensible Markup Language (XML) v1.0 <a href="http://www.w3.org/XML/">http://www.w3.org/XML/</a>  <b>Potencijalno:</b> 2. Election Markup Language (EML) v4.0 <a href="http://www.oasis-open.org/standards#emlv4.0">http://www.oasis-open.org/standards#emlv4.0</a> 3. Extensible Markup Language (XML) v1.1 <a href="http://www.w3.org/TR/xml11/">http://www.w3.org/TR/xml11/</a>
<b>Formati za razmjenu dokumenata</b>	
<b>Formati za razmjenu informacija dokumentima za razmjenu tekstuslanim dokumentima</b>	<b>Obavezno:</b> 1. Portable Document Format (PDF) v1.4  <a href="http://www.images.adobe.com/www.adobe.com/content/dam/Adobe/en/devnet/pdf/pdfs/pdf_reference_archives/PDFReference.pdf">http://www.images.adobe.com/www.adobe.com/content/dam/Adobe/en/devnet/pdf/pdfs/pdf_reference_archives/PDFReference.pdf</a> 2. Hypertext Markup Language (HTML)  <b>Preporučeno:</b> 3. Portable Document Format (PDF) v1.5  <a href="http://www.images.adobe.com/www.adobe.com/content/dam/Adobe/en/devnet/pdf/pdfs/pdf_reference_archives/PDFReference15_v6.pdf">http://www.images.adobe.com/www.adobe.com/content/dam/Adobe/en/devnet/pdf/pdfs/pdf_reference_archives/PDFReference15_v6.pdf</a>  <b>Potencijalno:</b> 4. Portable Document Format (PDF) v1.6  <a href="http://www.images.adobe.com/www.adobe.com/content/dam/Adobe/en/devnet/pdf/pdfs/pdf_reference_archives/PDFReference16.pdf">http://www.images.adobe.com/www.adobe.com/content/dam/Adobe/en/devnet/pdf/pdfs/pdf_reference_archives/PDFReference16.pdf</a>
<b>Formati dokumenata za daljnju obradu tekstualnih dokumenata za daljnju obradu</b>	<b>Obavezno:</b> 1. Text  <b>Preporučeno:</b> 2. Open Document Format for Office Applications (OpenDocument) v1.0 <a href="http://www.oasis-open.org/committees/download.php/12572/OpenDocument-v1.0-os.pdf">http://www.oasis-open.org/committees/download.php/12572/OpenDocument-v1.0-os.pdf</a>  <b>Potencijalno:</b> 3. Office Open XML (OOXML)


	PROJEKT IZRADA I USPOSTAVLJANJE OKVIRA INTEROPERABILNOSTI I STANDARDA ZA RAZMJENU PODATAKA  <b>VODILJE I STANDARDI ZA          ARHITEKTURU SUSTAVA I RAZVOJ APLIKACIJA</b>	Oznaka: PARCO-IFBIH
		Page: 79 of 88

	<a href="http://www.ecma-international.org/publications/standards/Ecma-376.htm">http://www.ecma-international.org/publications/standards/Ecma-376.htm</a>
<b>Formati za razmjenu informacija tabelama</b>	<p><b>Obavezno:</b></p> <ol style="list-style-type: none"> <li>Portable Document Format (PDF) v1.4   <a href="http://www.images.adobe.com/www.adobe.com/content/dam/Adobe/en/devnet/pdf/pdfs/pdf_reference_archives/PDFReference.pdf">http://www.images.adobe.com/www.adobe.com/content/dam/Adobe/en/devnet/pdf/pdfs/pdf_reference_archives/PDFReference.pdf</a> </li> </ol> <p><b>Preporučeno:</b></p> <ol style="list-style-type: none"> <li>Portable Document Format (PDF) v1.5   <a href="http://www.images.adobe.com/www.adobe.com/content/dam/Adobe/en/devnet/pdf/pdfs/pdf_reference_archives/PDFReference15_v6.pdf">http://www.images.adobe.com/www.adobe.com/content/dam/Adobe/en/devnet/pdf/pdfs/pdf_reference_archives/PDFReference15_v6.pdf</a> </li> </ol> <p><b>Potencijalno:</b></p> <ol style="list-style-type: none"> <li>Portable Document Format (PDF) v1.6  <a href="http://www.images.adobe.com/www.adobe.com/content/dam/Adobe/en/devnet/pdf/pdfs/pdf_reference_archives/PDFReference16.pdf">http://www.images.adobe.com/www.adobe.com/content/dam/Adobe/en/devnet/pdf/pdfs/pdf_reference_archives/PDFReference16.pdf</a> </li> </ol>
<b>Formati tabela za daljnju obradu</b>	<p><b>Obavezno:</b></p> <ol style="list-style-type: none"> <li>Comma Separated Value (CSV)  <a href="http://www.ietf.org/rfc/rfc4180.txt">http://www.ietf.org/rfc/rfc4180.txt</a> </li> </ol> <p><b>Potencijalno:</b></p> <ol style="list-style-type: none"> <li>Open Document Format for Office Applications (OpenDocument) v1.0  <a href="http://www.oasis-open.org/committees/download.php/12572/OpenDocument-v1.0-os.pdf">http://www.oasis-open.org/committees/download.php/12572/OpenDocument-v1.0-os.pdf</a> </li> <li>Office Open XML (OOXML)  <a href="http://www.ecma-international.org/publications/standards/Ecma-376.htm">http://www.ecma-international.org/publications/standards/Ecma-376.htm</a> </li> </ol>
<b>Formati za razmjenu informacija prezentacijama</b>	<p><b>Obavezno:</b></p> <ol style="list-style-type: none"> <li>Portable Document Format (PDF) v1.4   <a href="http://www.images.adobe.com/www.adobe.com/content/dam/Adobe/en/devnet/pdf/pdfs/pdf_reference_archives/PDFReference.pdf">http://www.images.adobe.com/www.adobe.com/content/dam/Adobe/en/devnet/pdf/pdfs/pdf_reference_archives/PDFReference.pdf</a> </li> <li>Hypertext Markup Language (HTML)</li> </ol> <p><b>Preporučeno:</b></p> <ol style="list-style-type: none"> <li>Portable Document Format (PDF) v1.5   <a href="http://www.images.adobe.com/www.adobe.com/content/dam/Adobe/en/devnet/pdf/pdfs/pdf_reference_archives/PDFReference15_v6.pdf">http://www.images.adobe.com/www.adobe.com/content/dam/Adobe/en/devnet/pdf/pdfs/pdf_reference_archives/PDFReference15_v6.pdf</a> </li> </ol>


	PROJEKT IZRADA I USPOSTAVLJANJE OKVIRA INTEROPERABILNOSTI I STANDARDA ZA RAZMJENU PODATAKA  <b>VODILJE I STANDARDI ZA          ARHITEKTURU SUSTAVA I RAZVOJ APLIKACIJA</b>	Oznaka: PARCO-IFBIH
		Page: 80 of 88

	<b>Potencijalno:</b> 4. Portable Document Format (PDF) v1.6  <a href="http://www.images.adobe.com/www.adobe.com/content/dam/Adobe/en/devnet/pdf/pdfs/pdf_reference_archives/PDFReference16.pdf">http://www.images.adobe.com/www.adobe.com/content/dam/Adobe/en/devnet/pdf/pdfs/pdf_reference_archives/PDFReference16.pdf</a> 5. Synchronized Multimedia Integration Language (SMIL) v2.0 <a href="http://www.w3.org/TR/2005/REC-SMIL2-20050107/">http://www.w3.org/TR/2005/REC-SMIL2-20050107/</a>
<b>Formati prezentacija za daljnju obradu</b>	<b>Potencijalno:</b> 1. Open Document Format for Office Applications (OpenDocument) v1.0 <a href="http://www.oasis-open.org/committees/download.php/12572/OpenDocument-v1.0-os.pdf">http://www.oasis-open.org/committees/download.php/12572/OpenDocument-v1.0-os.pdf</a> 2. Office Open XML (OOXML)  <a href="http://www.ecma-international.org/publications/standards/Ecma-376.htm">http://www.ecma-international.org/publications/standards/Ecma-376.htm</a>
<b>Sigurna razmjena dokumenata</b>	<b>Obavezno:</b> 1. Industrial Signature Interoperability Specification – MailTrust (ISIS-MTT) v1.1., Part 3 <a href="http://www.t7-isis.de/uploads/media/ISIS-MTT_Core_Specification_v1.1.pdf">http://www.t7-isis.de/uploads/media/ISIS-MTT_Core_Specification_v1.1.pdf</a> 2. XML Signature <a href="http://www.w3.org/TR/xmlsig-core/">http://www.w3.org/TR/xmlsig-core/</a> 3. XML Encryption <a href="http://www.w3.org/TR/xmlenc-core/">http://www.w3.org/TR/xmlenc-core/</a>  <b>Potencijalno:</b> 4. XML Advanced Electronic Signatures (XAdES) v1.2 <a href="http://www.w3.org/TR/XAdES/">http://www.w3.org/TR/XAdES/</a>
<b>Formati za razmjenu grafike</b>	<b>Obavezno:</b> 1. Graphics Interchange Format (GIF) v89a <a href="http://www.w3.org/Graphics/GIF/spec-gif89a.txt">http://www.w3.org/Graphics/GIF/spec-gif89a.txt</a> 2. Joint Photographic Experts Group (JPEG) <a href="http://www.jpeg.org/index.html">http://www.jpeg.org/index.html</a>  <b>Preporučeno:</b> 3. Portable Network Graphics (PNG) v1.2 <a href="http://www.w3.org/TR/PNG/">http://www.w3.org/TR/PNG/</a> 4. Tagged Image File Format (TIFF) v6.0 <a href="http://partners.adobe.com/public/developer/en/tiff/TIFF6.pdf">http://partners.adobe.com/public/developer/en/tiff/TIFF6.pdf</a> 5. Geo Tagged Image File Format (GeoTIFF) <a href="http://trac.osgeo.org/geotiff/">http://trac.osgeo.org/geotiff/</a>  <b>Potencijalno:</b> 6. Joint Photographic Experts Group 2000 (JPEG2000) / Part 1




	PROJEKT IZRADA I USPOSTAVLJANJE OKVIRA INTEROPERABILNOSTI I STANDARDA ZA RAZMJENU PODATAKA  <b>VODILJE I STANDARDI ZA          ARHITEKTURU SUSTAVA I RAZVOJ APLIKACIJA</b>	Oznaka: PARCO-IFBIH
		Page: 81 of 88


	<a href="http://www.jpeg.org/jpeg2000/">http://www.jpeg.org/jpeg2000/</a>
<b>Animacija</b>	<b>Obavezno:</b> Animated Graphics Interchange Format (Animated GIF) v89a <a href="http://www.w3.org/Graphics/GIF/spec-gif89a.txt">http://www.w3.org/Graphics/GIF/spec-gif89a.txt</a>
<b>Audio i video podaci</b>	
<b>Formati za razmjenu audio i video datoteka</b>	<b>Preporučeno:</b> <ol style="list-style-type: none"> <li>Quicktime <a href="http://www.apple.com/quicktime/">http://www.apple.com/quicktime/</a></li> <li>MPEG-4 Part 14 (MP4) <a href="http://www.iso.org/iso/iso_catalogue/catalogue_tc/catalogue_detail.htm?csnumber=38538">http://www.iso.org/iso/iso_catalogue/catalogue_tc/catalogue_detail.htm?csnumber=38538</a></li> <li>Ogg Encapsulation Format (Ogg) <a href="http://www.ietf.org/rfc/rfc3533.txt">http://www.ietf.org/rfc/rfc3533.txt</a></li> </ol> <b>Potencijalno:</b> <ol style="list-style-type: none"> <li>Windows Media Video (WMV) v9</li> <li>RealMedia v10 <a href="http://www.realnetworks.com/">http://www.realnetworks.com/</a></li> </ol>
<b>Formati za razmjenu audio i video streaming-a</b>	<b>Obavezno:</b> <ol style="list-style-type: none"> <li>Hypertext Transfer Protocol (HTTP) v1.1 <a href="http://www.ietf.org/rfc/rfc2616.txt">http://www.ietf.org/rfc/rfc2616.txt</a></li> </ol> <b>Preporučeno:</b> <ol style="list-style-type: none"> <li>Quicktime <a href="http://www.apple.com/quicktime/">http://www.apple.com/quicktime/</a></li> <li>MPEG-4 Part 14 (MP4) <a href="http://www.iso.org/iso/iso_catalogue/catalogue_tc/catalogue_detail.htm?csnumber=38538">http://www.iso.org/iso/iso_catalogue/catalogue_tc/catalogue_detail.htm?csnumber=38538</a></li> <li>Ogg Encapsulation Format (Ogg) <a href="http://www.ietf.org/rfc/rfc3533.txt">http://www.ietf.org/rfc/rfc3533.txt</a></li> </ol> <b>Potencijalno:</b> <ol style="list-style-type: none"> <li>Windows Media Video (WMV) v9</li> <li>RealMedia v10 <a href="http://www.realnetworks.com/">http://www.realnetworks.com/</a></li> </ol>
<b>Formati za razmjenu geo-informacija</b>	<b>Preporučeno:</b> <ol style="list-style-type: none"> <li>Geography Markup Language (GML) v3.1.1</li> <li>Geography Markup Language (GML) v2.1.2 <a href="http://www.opengeospatial.org/standards/gml">http://www.opengeospatial.org/standards/gml</a></li> </ol>
<b>Kompresija podataka</b>	<b>Obavezno:</b> <ol style="list-style-type: none"> <li>ZIP v2.0 <a href="http://www.pkware.com/documents/casestudies/APPNOTE.T">http://www.pkware.com/documents/casestudies/APPNOTE.T</a></li> </ol>

	PROJEKT IZRADA I USPOSTAVLJANJE OKVIRA INTEROPERABILNOSTI I STANDARDA ZA RAZMJENU PODATAKA  <b>VODILJE I STANDARDI ZA          ARHITEKTURU SUSTAVA I RAZVOJ APLIKACIJA</b>	Oznaka: PARCO-IFBIH
		Page: 82 of 88


	<a href="#">XT</a>  <b>Preporučeno:</b> 2. Gnu ZIP (GZIP) v4.3 / Tape ARchive (TAR) <a href="http://www.ietf.org/rfc/rfc1952.txt">http://www.ietf.org/rfc/rfc1952.txt</a>
<b>Tehnologije za prikaz na mobilnim uređajima</b>	<b>Obavezno:</b> 1. Short Message Services (SMS) <a href="http://www.3gpp.org/ftp/Specs/html-info/23040.htm">http://www.3gpp.org/ftp/Specs/html-info/23040.htm</a>  <b>Potencijalno:</b> 2. Wireless Application Protocol (WAP) v2.0 <a href="http://www.openmobilealliance.org/tech/affiliates/wap/wapindex.html">http://www.openmobilealliance.org/tech/affiliates/wap/wapindex.html</a> 3. Extensible Hypertext Markup Language (XHTML) Basic v1.0 <a href="http://www.w3.org/TR/2000/REC-xhtml-basic-20001219/">http://www.w3.org/TR/2000/REC-xhtml-basic-20001219/</a>
<b>Komunikacija</b>	
<b>Middleware komunikacija</b>	
<b>Middleware komunikacija unutar administracije</b>	<b>Obavezno:</b> 1. Remote Method Invocation (RMI) <a href="http://www.oracle.com/technetwork/java/javase/tech/index-jsp-136424.html">http://www.oracle.com/technetwork/java/javase/tech/index-jsp-136424.html</a> 2. Simple Object Access Protocol (SOAP) v1.1 <a href="http://www.w3.org/TR/soap11/">http://www.w3.org/TR/soap11/</a> 3. Web Services Description Language (WSDL) v1.1 <a href="http://www.w3.org/TR/wsdl">http://www.w3.org/TR/wsdl</a> 4. Java Message Service (JMS) v1.1 <a href="http://www.jcp.org/en/jsr/detail?id=914">http://www.jcp.org/en/jsr/detail?id=914</a> 5. J2EE Connector Architecture (JCA) v1.5 <a href="http://www.jcp.org/en/jsr/detail?id=112">http://www.jcp.org/en/jsr/detail?id=112</a>  <b>Preporučeno:</b> 6. Java Language Mapping to OMG IDL <a href="http://www.omg.org/technology/documents/idl2x_spec_catalog.htm">http://www.omg.org/technology/documents/idl2x_spec_catalog.htm</a> 7. Web Services (WS) Security v1.1 <a href="http://www.oasis-open.org/standards#wssv1.1">http://www.oasis-open.org/standards#wssv1.1</a>  <b>Potencijalno:</b> 8. Universal Description, Discovery and Integration (UDDI) v2.0 <a href="http://uddi.xml.org/">http://uddi.xml.org/</a>
<b>Middleware komunikacija sa aplikacijama izvan administracije</b>	<b>Obavezno:</b> 1. Simple Object Access Protocol (SOAP) v1.1 <a href="http://www.w3.org/TR/soap11/">http://www.w3.org/TR/soap11/</a> 2. Web Services Description Language (WSDL) v1.1 <a href="http://www.w3.org/TR/wsdl">http://www.w3.org/TR/wsdl</a>

	PROJEKT IZRADA I USPOSTAVLJANJE OKVIRA INTEROPERABILNOSTI I STANDARDA ZA RAZMJENU PODATAKA  <b>VODILJE I STANDARDI ZA          ARHITEKTURU SUSTAVA I RAZVOJ APLIKACIJA</b>	Oznaka: PARCO-IFBIH
		Page: 83 of 88


	<p><b>Preporučeno:</b></p> <p>3. Web Services (WS) Security v1.1  <a href="http://www.oasis-open.org/standards#wssv1.1">http://www.oasis-open.org/standards#wssv1.1</a></p> <p><b>Potencijalno:</b></p> <p>4. Universal Description, Discovery and Integration (UDDI) v2.0  <a href="http://uddi.xml.org/">http://uddi.xml.org/</a></p>
<b>Mrežni protokoli</b>	<p><b>Obavezno:</b></p> <p>1. Internet Protocol (IP) v4  <a href="http://www.ietf.org/rfc/rfc791.txt">http://www.ietf.org/rfc/rfc791.txt</a></p> <p>2. Domain Name System (DNS)  <a href="http://www.ietf.org/rfc/rfc1034.txt">http://www.ietf.org/rfc/rfc1034.txt</a>  <a href="http://www.ietf.org/rfc/rfc1035.txt">http://www.ietf.org/rfc/rfc1035.txt</a></p> <p><b>Potencijalno:</b></p> <p>3. Internet Protocol (IP) v6  <a href="http://www.ietf.org/rfc/rfc2460.txt">http://www.ietf.org/rfc/rfc2460.txt</a></p>
<b>E-mail komunikacija</b>	<p><b>Obavezno:</b></p> <p>1. Simple Mail Transfer Protocol (SMTP) / Multipurpose Internet Mail Extensions (MIME) v1.0</p> <p>2. Post Office Protocol (POP) v3 / Internet Message Access Protocol (IMAP) v4rev1  <a href="http://www.ietf.org/rfc/rfc1939.txt">http://www.ietf.org/rfc/rfc1939.txt</a></p> <p>3. Industrial Signature Interoperability Specification – MailTrust (ISIS-MTT) v1.1., Part 1 to Part 6  <a href="http://www.t7-isis.de/uploads/media/ISIS-MTT_Core_Specification_v1.1.pdf">http://www.t7-isis.de/uploads/media/ISIS-MTT_Core_Specification_v1.1.pdf</a></p>
<b>IP telefonija</b>	<p><b>Preporučeno:</b></p> <p>1. H.323  <a href="http://www.itu.int/rec/T-REC-H.323/en">http://www.itu.int/rec/T-REC-H.323/en</a></p> <p><b>Potencijalno:</b></p> <p>2. Session Initiation Protocol (SIP) v2.0  <a href="http://www.ietf.org/rfc/rfc3261.txt">http://www.ietf.org/rfc/rfc3261.txt</a></p>
<b>Aplikativni protokoli</b>	<p><b>Obavezno:</b></p> <p>1. File Transfer Protocol (FTP)  <a href="http://www.ietf.org/rfc/rfc959.txt">http://www.ietf.org/rfc/rfc959.txt</a></p> <p>2. Hypertext Transfer Protocol (HTTP) v1.1  <a href="http://www.ietf.org/rfc/rfc2616.txt">http://www.ietf.org/rfc/rfc2616.txt</a></p> <p>3. Online Service Computer Interface (OSCI) Transport v1.2  <a href="http://www.immagic.com/eLibrary/TECH/OSCI_DE/O090601T.pdf">http://www.immagic.com/eLibrary/TECH/OSCI_DE/O090601T.pdf</a></p> <p>4. Transport Layer Security (TLS) v1.0  <a href="http://www.ietf.org/rfc/rfc2246.txt">http://www.ietf.org/rfc/rfc2246.txt</a></p> <p><b>Preporučeno:</b></p>

	PROJEKT IZRADA I USPOSTAVLJANJE OKVIRA INTEROPERABILNOSTI I STANDARDA ZA RAZMJENU PODATAKA  <b>VODILJE I STANDARDI ZA          ARHITEKTURU SUSTAVA I RAZVOJ APLIKACIJA</b>	Oznaka: PARCO-IFBIH
		Page: 84 of 88


	5. Secure Shell v2 (SSH-2) <a href="http://www.ietf.org/rfc/rfc4251.txt">http://www.ietf.org/rfc/rfc4251.txt</a> 6. WWW Distributed Authoring and Versioning (WebDAV) <a href="http://www.ietf.org/rfc/rfc2518.txt">http://www.ietf.org/rfc/rfc2518.txt</a>  <b>Potencijalno:</b> 7. Transport Layer Security (TLS) v1.1 <a href="http://www.ietf.org/rfc/rfc4346.txt">http://www.ietf.org/rfc/rfc4346.txt</a>
<b>Geo-servisi</b>	<b>Preporučeno:</b> 1. Catalogue Services Specification v2.0 – ISO Metadata Application Profile v1.0 <a href="http://www.opengeospatial.org/standards/cat">http://www.opengeospatial.org/standards/cat</a> 2. Web Map Service (WMS) v1.0 <a href="http://www.opengeospatial.org/standards/wms">http://www.opengeospatial.org/standards/wms</a> 3. Web Coverage Service (WCS) v1.0.0 <a href="http://www.opengeospatial.org/standards/wcs">http://www.opengeospatial.org/standards/wcs</a> 4. Web Feature Service (WFS) v1.0 <a href="http://www.opengeospatial.org/standards/wfs">http://www.opengeospatial.org/standards/wfs</a> 5. Web Feature Service (WFS) v1.1 <a href="http://www.opengeospatial.org/standards/wfs">http://www.opengeospatial.org/standards/wfs</a> 6. Simple Feature Access – Part 2: SQL option (SFA-2) v1.1.0 <a href="http://www.opengeospatial.org/standards/sfs">http://www.opengeospatial.org/standards/sfs</a>
<b>Backend</b>	
<b>Servisi direktorija i registara</b>	<b>Obavezno:</b> 1. Lightweight Directory Access Protocol (LDAP) v3 <a href="http://www.ietf.org/rfc/rfc4510.txt">http://www.ietf.org/rfc/rfc4510.txt</a>  <b>Potencijalno:</b> 2. Universal Description, Discovery and Integration (UDDI) v2.0 <a href="http://uddi.xml.org/">http://uddi.xml.org/</a> 3. Directory Services Markup Language (DSML) v2 <a href="http://www.oasis-open.org/standards#dsmlv2">http://www.oasis-open.org/standards#dsmlv2</a> 4. ebXML Registry Services and Protocols (ebXML RS) v3.0/ ebXML Registry Information Model (ebXML RIM) v3.0 <a href="http://www.oasis-open.org/standards#ebxmlrimv3.0">http://www.oasis-open.org/standards#ebxmlrimv3.0</a>
<b>Pristup bazama podataka</b>	<b>Obavezno:</b>  Java Database Connectivity (JDBC) v3.0  <a href="http://www.jcp.org/en/jsr/detail?id=54">http://www.jcp.org/en/jsr/detail?id=54</a>
<b>Pristup postojećim sistemima</b>	<b>Obavezno:</b> 1. Remote Method Invocation (RMI) <a href="http://www.oracle.com/technetwork/java/javase/tech/index-jsp-136424.html">http://www.oracle.com/technetwork/java/javase/tech/index-jsp-136424.html</a> 2. Simple Object Access Protocol (SOAP) v1.1 <a href="http://www.w3.org/TR/soap11/">http://www.w3.org/TR/soap11/</a>

	PROJEKT IZRADA I USPOSTAVLJANJE OKVIRA INTEROPERABILNOSTI I STANDARDA ZA RAZMJENU PODATAKA  <b>VODILJE I STANDARDI ZA          ARHITEKTURU SUSTAVA I RAZVOJ APLIKACIJA</b>	Oznaka: PARCO-IFBIH
		Page: 85 of 88


	3. Web Services Description Language (WSDL) v1.1 <a href="http://www.w3.org/TR/wsdl">http://www.w3.org/TR/wsdl</a> 4. Java Message Service (JMS) v1.1 <a href="http://www.jcp.org/en/jsr/detail?id=914">http://www.jcp.org/en/jsr/detail?id=914</a> 5. J2EE Connector Architecture (JCA) v1.5 <a href="http://www.jcp.org/en/jsr/detail?id=112">http://www.jcp.org/en/jsr/detail?id=112</a>  <b>Preporučeno:</b> 6. Java Language Mapping to OMG IDL <a href="http://www.omg.org/technology/documents/idl2x_spec_catalog.htm">http://www.omg.org/technology/documents/idl2x_spec_catalog.htm</a> 7. Web Services (WS) Security v1.1 <a href="http://www.oasis-open.org/standards#wssv1.1">http://www.oasis-open.org/standards#wssv1.1</a>
<b>Enkripcija</b>	
<b>Metode asimetrične enkripcije</b>	<b>Obavezno:</b>  RSA  <a href="http://www.ietf.org/rfc/rfc2437.txt">http://www.ietf.org/rfc/rfc2437.txt</a>
<b>Metode simetrične enkripcije</b>	<b>Obavezno:</b>  Advanced Encryption Standard (AES)  <a href="http://csrc.nist.gov/publications/fips/fips197/fips-197.pdf">http://csrc.nist.gov/publications/fips/fips197/fips-197.pdf</a>
<b>Elektronski potpis</b>	
<b>Elektronski potpis</b>	<b>Obavezno:</b>  Kriptografski algoritmi za elektronski potpis prema važećem zakonu
<b>Haširanje podataka</b>	<b>Obavezno:</b> 1. Secure Hash Algorithm (SHA)-256 <a href="http://csrc.nist.gov/publications/fips/fips180-2/fips180-2.pdf">http://csrc.nist.gov/publications/fips/fips180-2/fips180-2.pdf</a>  <b>Preporučeno:</b> 2. Secure Hash Algorithm (SHA)-224 / Secure Hash Algorithm (SHA)-384 / Secure Hash Algorithm (SHA)-512 <a href="http://csrc.nist.gov/publications/fips/fips180-2/fips180-2.pdf">http://csrc.nist.gov/publications/fips/fips180-2/fips180-2.pdf</a>
<b>Metode asimetričnog potpisa</b>	<b>Obavezno:</b> 1. RSA <a href="http://www.ietf.org/rfc/rfc2437.txt">http://www.ietf.org/rfc/rfc2437.txt</a>  <b>Preporučeno:</b> 2. Digital Signature Algorithm (DSA) <a href="http://csrc.nist.gov/publications/fips/fips186-3/fips_186-3.pdf">http://csrc.nist.gov/publications/fips/fips186-3/fips_186-3.pdf</a>

	PROJEKT IZRADA I USPOSTAVLJANJE OKVIRA INTEROPERABILNOSTI I STANDARDA ZA RAZMJENU PODATAKA  <b>VODILJE I STANDARDI ZA          ARHITEKTURU SUSTAVA I RAZVOJ APLIKACIJA</b>	Oznaka: PARCO-IFBIH
		Page: 86 of 88

<b>Upravljanje ključevima</b>	<b>Preporučeno:</b>  XML Key Management Specification (XKMS) v2  <a href="http://www.w3.org/TR/xkms2/">http://www.w3.org/TR/xkms2/</a>
<b>Pametne kartice</b>	
<b>Kontaktne pametne kartice</b>	<b>Obavezno:</b>  Identification Cards - Integrated circuit cards  <a href="http://www.iso.org/iso/iso_catalogue/catalogue_tc/catalogue_detail.htm?csnumber=35168">http://www.iso.org/iso/iso_catalogue/catalogue_tc/catalogue_detail.htm?csnumber=35168</a>
<b>Bezkontaktne pametne kartice</b>	<b>Obavezno:</b>  Identification Cards - Contactless integrated circuit cards  <a href="http://www.iso.org/iso/iso_catalogue/catalogue_tc/catalogue_detail.htm?csnumber=39693">http://www.iso.org/iso/iso_catalogue/catalogue_tc/catalogue_detail.htm?csnumber=39693</a>
<b>Čitači i interfejsi za pametne kartice</b>	<b>Obavezno:</b> 1. Technical guideline for federal-government eCard projects (BSI TR-03116) v1.0 2. Industrial Signature Interoperability Specification – MailTrust (ISIS-MTT) v1.1., Part 7  <a href="http://www.t7-isis.de/uploads/media/ISIS-MTT_Core_Specification_v1.1.pdf">http://www.t7-isis.de/uploads/media/ISIS-MTT_Core_Specification_v1.1.pdf</a>  <b>Potencijalno:</b> 1. Interoperability Specification for ICCs and Personal Computer Systems (PC/SC) v2.0 <a href="http://www.pcscworkgroup.com/specifications/specdownload.php">http://www.pcscworkgroup.com/specifications/specdownload.php</a> 2. OpenCard Framework (OCF) v1.2 <a href="http://www.openscdp.org/ocf/">http://www.openscdp.org/ocf/</a> 3. Secure Interoperable ChipCard Terminal (SICCT) v1.10 <a href="http://www.teletrust.de/uploads/media/SICCT-Spezifikation-120.pdf">http://www.teletrust.de/uploads/media/SICCT-Spezifikation-120.pdf</a>
<b>Dugoročno arhiviranje</b>	
	<b>Preporučeno:</b> 1. Tagged Image File Format (TIFF) v6.0 <a href="http://partners.adobe.com/public/developer/en/tiff/TIFF6.pdf">http://partners.adobe.com/public/developer/en/tiff/TIFF6.pdf</a> 2. Joint Photographic Experts Group (JPEG) <a href="http://www.jpeg.org/index.html">http://www.jpeg.org/index.html</a> 3. Extensible Markup Language (XML) v1.0

	PROJEKT IZRADA I USPOSTAVLJANJE OKVIRA INTEROPERABILNOSTI I STANDARDA ZA RAZMJENU PODATAKA	Oznaka: PARCO-IFBIH
	<b>VODILJE I STANDARDI ZA ARHITEKTURU SUSTAVA I RAZVOJ APLIKACIJA</b>	Page: 87 of 88

	<p><a href="http://www.w3.org/XML/">http://www.w3.org/XML/</a></p> <p>4. ArchiSig <a href="http://www.pdfa.org/tag/archisig/">http://www.pdfa.org/tag/archisig/</a></p> <p>5. Portable Document Format Archive - 1 (PDF/A-1) <a href="http://www.pdfa.org/publications/">http://www.pdfa.org/publications/</a></p> <p><b>Potencijalno:</b></p> <p>6. Extensible Markup Language (XML) v1.1 <a href="http://www.w3.org/TR/xml11/">http://www.w3.org/TR/xml11/</a></p>
--	--

	PROJEKT IZRADA I USPOSTAVLJANJE OKVIRA INTEROPERABILNOSTI I STANDARDA ZA RAZMJENU PODATAKA	Oznaka: PARCO-IFBIH
	VODILJE I STANDARDI ZA ARHITEKTURU SUSTAVA I RAZVOJ APLIKACIJA	Page: 88 of 88

## Prilog 2. Bibliografija

1. **Working Group on IT Architecture within the Coordinating Information Committee.** *White Paper on Enterprise Architecture in Denmark.* Copenhagen : Ministry of Science, Technology and Innovation, 2003.
2. **Bosna i Hercegovina, Agencija za identifikacijske/identifikacione isprave/dokumente, evidenciju i razmjenu podataka.** *Standardi za opremu i softver.* Banja Luka : an., 2010.
3. **The Open Group.** *SOA Governance Framework.* 2009.
4. **Republika Hrvatska, Centralni državni ured za e-Hrvatsku.** *Odrednice Hrvatskog okvira za interoperabilnost (HROI) ver. 1.0.* Zagreb : an., 2010.
5. **eGovernment Economics Project (eGEP).** *Expenditure Study, Final Version.* Brussels : eGovernment Unit, DG Information Society, European Commission.
6. **Der Beauftragte der Bundesregierung für Informationstechnik.** *SAGA - Standards and Architectures for eGovernment Applications v.4.0.* 2008.
7. **UN/CEFACT - United Nations Centre for Trade Facilitation and Electronic Business.** *Core Components Technical Specification ver. 2.01.* 2003.