

REPUBLIKA SLOVENIJA
UNIVERZA V MARIBORU
EKONOMSKO-POSLOVNA FAKULTETA

Magistrsko delo

Povezovanje CMMI in COBIT metode v metodo izdelave ali naročanja programske opreme

Junij 2007

Drago Perc

REPUBLIKA SLOVENIJA
UNIVERZA V MARIBORU
EKONOMSKO-POSLOVNA FAKULTETA

Magistrsko delo
Povezovanje CMMI in COBIT metode v metodo
izdelave ali naročanja programske opreme

Kandidat: Drago Perc, univ. dipl. ekon.,
rojen leta 1967 v kraju Celje
zaposlen na Zavodu Republike Slovenije za zaposlovanje
kot koordinator zaposlovanja
Absolvent na smeri Poslovna informatika
Tema odobrena na seji senata EPF dne 18. 5. 2007
z delovnim naslovom Povezovanje CMMI in COBIT metode v metodo izdelave ali
naročanja programske opreme
Mentor: Dr. Samo Bobek, redni profesor

KAZALO

1 UVOD.....	6
1.1 Opredelitev področja in opis problema.....	6
1.2 Namen, cilji in trditve raziskave.....	6
1.3 Predpostavke in omejitve.....	7
1.4 Uporabljene raziskovalne metode.....	7
2 NAROČANJE, IZDELAVA PROGRAMSKE OPREME IN PREVERJANJE KAKOVOSTI PROGRAMSKIH PROIZVODOV.....	9
2.1 Proces nabave programske opreme.....	9
2.1.1 Linearni pristop.....	10
2.1.2 Prototipni pristop.....	11
2.1.3 Objektni pristop.....	12
2.2 Metode nabave programske opreme.....	16
2.2.1 CMMI.....	16
2.2.2 COBIT.....	27
2.3 Specifikacija zahtev za programsko opremo.....	33
2.4 Kakovost in testiranje programske opreme	35
2.4.1 ISO 9000.....	36
2.4.2 Testiranje programske opreme po IEEE standardu	41
2.5 Metodologija vodenja projektov v državni upravi.....	45
2.5.1 Uvod.....	46
2.5.2 Namen.....	47
2.5.3 Struktura gradiva Metodologija vodenja projektov v državni upravi.....	47
2.5.4 Predstavitev metodologije.....	48
2.5.5 Prednosti in pričakovane koristi uporabe metodologije	50
2.6 Sklep.....	52
3 METODA IZDELAVE ALI NAROČANJA PROGRAMSKE OPREME.....	53
3.1 Vzpostavitev projektne organizacije.....	53
3.2 Analiza stanja.....	53
3.3 Postopkovni (procesni) model.....	55
3.4 Podatkovni model.....	57
3.5 Funkcionalnost aplikacije (tipke).....	60
3.6 Ekranski vmesniki.....	60
3.7 Specifikacija zahtev za programsko opremo.....	61
3.8 Javni razpis.....	62
3.9 Prenos podatkov.....	63
3.10 Programiranje.....	63
3.11 Testiranje.....	63
3.12 Kakovost.....	64
3.13 Izobraževanje.....	64
3.14 Uvajanje v produkcijo.....	64
3.15 Upravljanje in vzdrževanje.....	65
4 PREVERJANJE UPORABNOSTI METODE.....	66
4.1 Projekt naročila aplikacije ZP_Net.....	66
4.1.1 Priprava projektne dokumentacije s terminskim planom.....	67

4.1.2 Analiza obstoječega stanja in izdelava postopkovnega modela.....	71
4.1.3 Definicije potrebnih podatkov.....	81
4.1.4 Izdelava podatkovnega modela.....	86
4.1.5 Funkcionalnost aplikacije.....	88
4.1.6 Ekranski vmesniki.....	88
4.1.7 Javni razpis.....	90
4.1.8 Prenos podatkov.....	92
4.1.9 Programiranje, testiranje, izobraževanje in zaključek projekta.....	92
4.2 Projekt izdelave aplikacije APZ_Net.....	97
4.2.1 Priprava projektne dokumentacije s terminskim planom.....	98
4.2.2 Analiza obstoječega stanja in izdelava postopkovnega modela.....	103
4.2.3 Definicija potrebnih podatkov.....	105
4.2.4 Izdelava podatkovnega modela.....	107
4.2.5 Funkcionalnost aplikacije.....	113
4.2.6 Ekranski vmesniki.....	114
4.2.7 Prenos podatkov.....	114
4.2.8 Programiranje, testiranje, izobraževanje in zaključek projekta.....	115
5 SKLEP.....	120
SEZNAM LITERATURE IN VIROV.....	123
SEZNAM UPORABLJENIH KRATIC	
SEZNAM SLIK	
DELOVNI ŽIVLJENJEPIŠ	

Povzetek

V svetu je poznanih veliko metod, ki omogočajo izdelavo ali naročanje programske opreme. Med njimi je zelo težko izbrati tisto, ki bi posameznemu podjetju najbolj ustrezala. Na Zavodu Republike Slovenije za zaposlovanje smo imeli to težavo, da nismo poznali nobene izmed metod. Med proučevanjem CMMI in COBIT metode smo spoznali, da bi našemu delu najbolj ustrezala kombinacija obeh. Pri izdelavi lastne metode smo uporabili še klasične pristope življenjskega cikla kot so linearni, prototipni in objektni pristop. Samo metodo smo razširili še s prilagojeno metodo specifikacije zahtev za programsko opremo po standardu IEEE 830-1998.

Na takšen način smo oblikovali metodo, ki je uporabna tako za naročanje, kot tudi za izdelavo programske opreme. Metodo sestavljajo naslednja poglavja:

1. Vzpostavitev projektne organizacije
2. Analiza stanja
3. Postopkovni (procesni) model
4. Podatkovni model
5. Funkcionalnost aplikacije
6. Ekranski vmesniki
7. Specifikacija zahtev za programsko opremo
8. Javni razpis
9. Prenos podatkov
10. Programiranje
11. Testiranje
12. Kakovost
13. Izobraževanje
14. Uvajanje v produkcijo
15. Upravljanje in vzdrževanje

Metoda je bila preizkušena pri naročanju izdelave programske opreme pri zunanjem izvajalcu in pri izdelavi aplikacije z lastnimi programerji. Pri tem se je izkazala kot uspešna, saj se je v prvem primeru projekt že uspešno zaključil, v drugem primeru pa se zaključuje.

Ključne besede

Linearni pristop, prototipni pristop, objektni pristop, CMMI, COBIT, projektno vodenje, nabava programske opreme, izdelava programske opreme.

Title: Connecting of CMMI and COBIT method to method of making or of ordering of software

Abstract

Many methods for the making or ordering the software are known. Among them it is hard to choose the method which would be the most suitable for the particular company. At Employment Service of Slovenia we had a problem of not knowing these methods. While studying CMMI and COBIT, we found out that the combination of both methods would be the most suitable. At making out our own method, we used also classical approaches of life cycle as are: linear, prototype or object approach. Merely method expanded still with adapted procedure of specification of demands for software at standard IEEE 830-1998.

In this way we designed the method, which will be used for ordering and for the making of software. The method is composed of the following chapters:

- 1.Establishing of project organization
- 2.Analysis of situation
- 3.Procedural (processing) model
- 4.Data model
- 5.Functionality of application
- 6.Screen interfaces
- 7.Specification of requests for software
- 8.Call for tenders
- 9.Transmission of data
- 10.Programming
- 11.Testing
- 12.Quality
- 13.Training
- 14.Implementing into production
- 15.Administration and maintenance

The method was tested at ordering of the software at the organization of outside Employment service of Slovenia and at the production of application with our own IT experts. It was proved as successful method, because the first part of the project was successfully carried out and the second part will be finished soon.

Key words:

Linear approach, prototype approach, objective approach, CMMI, COBIT, project management, purchase of program equipment, production of software.

1 UVOD

1.1 Opredelitev področja in opis problema

Na Zavodu RS za zaposlovanje je bilo potrebno izdelati novo programsko opremo za evidentiranje in spremljanje brezposelnih oseb in za posredovanje teh oseb na prosta delovna mesta. Dosedanji program je bil napisan v Clipperju in deloval v okolju DOS. Sam program je bil že na meji zmožnosti tega okolja in ga je bilo zaradi tega nujno potrebno prenoviti in prilagoditi okolju Windows. Ker sami nismo imeli znanja za razvoj in izdelavo programske opreme v okolju Windows, je bilo potrebno izdelavo naročiti pri zunanjih izvajalcih.

Z naročanjem izdelave programske opreme nismo imeli nobenih izkušenj, saj smo vso programsko opremo do sedaj izdelali sami. Poznali nismo nobene metode¹, zato smo metodo naročanja pripravili tako, kot smo mislili, da bo za izvajalca najbolj razumljivo.

Samo naročilo in kasneje izdelava sta temeljila na klasičnem življenjskem ciklu izdelave programske opreme, delno na CMMI metodi, delno pa po metodi COBIT.

Vse metode, ki smo jih uporabili, bom združil v eno metodo, ki bo veljala za naročanje ali izdelavo ostale programske opreme, ki jo na Zavodu še moramo prenoviti. Večina te programske opreme je še vedno napisana v Clipperju in jo bo zaradi tega potrebno prenoviti in na novo napisati. Na podlagi izkušenj, ki smo jih pridobili pri naročilu opisanega programa, bom v magistrskem delu, razvil in opisal metodo, ki jo bomo lahko uporabili pri nadaljnji prenovi aplikativne opreme.

1.2 Namen, cilji in trditve raziskave

Pri razvoju programske opreme smo na Zavodu precej zaostali. Predolgo smo namreč vztrajali na zastarelih aplikacijah, ko pa smo bili v prenavo prisiljeni, smo bili nanjo nepripravljeni. Poznali nismo nobenih metod in metodologij s katerimi bi si pri prenovi aplikacij lahko pomagali. Zanašali smo se lahko samo na iznajdljivost in znanje posameznikov.

Tehnološki razvoj je nenehen in temu bo potrebno prilagoditi tudi programsko opremo. Zaradi specifične dejavnosti, ki jo Zavod opravlja in nenehnega spreminjanja zakonskih podlag, je potrebno izdelati programe po naročilu ali pa v lastni režiji.

¹ Metoda - oblika načrtnega, premišljenega dejanja, ravnanja ali mišljenja za doseg kakega cilja; način, postopek; (SSK 1995).

Osnovni namen je torej razviti metodo, ki bo omogočila nadaljnjo prenovu in razvoj novih računalniških aplikacij ne glede na to ali bomo izdelavo naročili pri zunanjih izvajalcih ali pa jo bomo izdelali sami.

Cilji magistrskega dela so:

1. Pregled obstoječih metod: teorije življenjskega cikla, CMMI in COBIT metode, ugotavljanje katera metoda je v dani situaciji najprimernejša oz. katere metode je potrebno prilagoditi, da bodo ustrezale Zavodovemu načinu dela.
2. Preveriti, če je oblikovana metoda primerna za naročanje in izdelavo programske opreme.
3. Ugotoviti, kdaj je najugodnejši trenutek za zaključek projekta pri izdelavi programske opreme. Velik problem v večini podjetij je namreč določitev roka, ko projektni management zaključi s svojim delom, nadaljnji razvoj programa pa prevzame funkcijski management.

1.3 Predpostavke in omejitve

Predpostavljam, da bom pri pisanju magistrske naloge ugotovil, da se z upoštevanjem ustreznih metod in standardov nabavi kvalitetnejša programska oprema, kot bi se brez upoštevanja le teh.

Pri pisanju magistrske naloge se bom omejil predvsem na metodi CMMI in COBIT. Menim, da si lahko s pomočjo teh dveh metod dovolj pomagamo pri naročilu in nakupu programske opreme, da bo ta zadovoljila kupca.

Omejitev je tudi primerjava samo dveh aplikacij ZP_Net in APZ_Net, na osnovi katerih bo temeljila opisana metoda.

1.4 Uporabljene raziskovalne metode

- Deskriptivna metoda – opisoval bom posamezne metode in jih primerjal med seboj
- Metoda klasifikacije – definiral bom posamezne pojme, ki jih bom pri raziskavi uporabil
- Metoda analize – analiziral bom posamezne metode in njihovo uporabnost za Zavod
- Metoda konkretizacije – iz posameznih metod bom izločil bistvene in uporabne značilnosti, ki jih bom uporabil pri razvoju metode

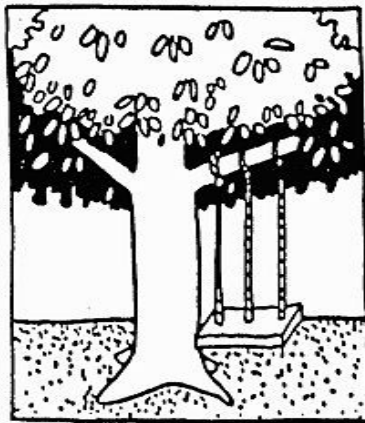
- Metoda komparacije – primerjal bom uporabnost metode pri naročanju programske opreme in pri izdelavi v lastni režiji
- Metoda kompilacije – povzemal bom opazovanja, stališča, sklepe in spoznanja drugih avtorjev

2 NAROČANJE, IZDELAVA PROGRAMSKE OPREME IN PREVERJANJE KAKOVOSTI PROGRAMSKIH PROIZVODOV

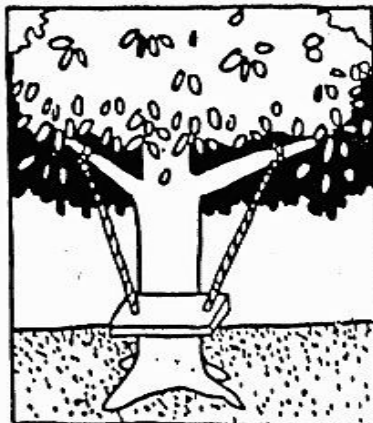
2.1 Proces nabave programske opreme

Programska oprema se lahko nabavi na dva načina. En način jo, da ga kupimo v trgovini, drugi pa je, da njeno izdelavo naročimo. Pri prvem načinu dobimo tisto, kar je proizvajalec predvidel, da bo ustrezalo večini uporabnikov. Ker pa vedno ni tako in ker imajo določeni uporabniki specifične zahteve, je potrebno programsko opremo tem uporabnikom prilagoditi oz. izdelati novo. Pri naročanju je potrebno paziti, da se ne zgodi kaj podobnega, kot je prikazano na spodnji sliki (Solina 1998, 4):

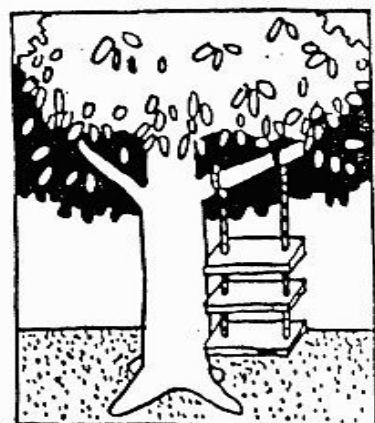
SLIKA 1: KRIZA PROGRAMSKE OPREME



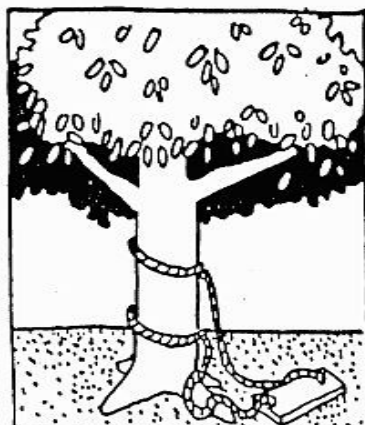
1. predlog zastopnika uporabnika



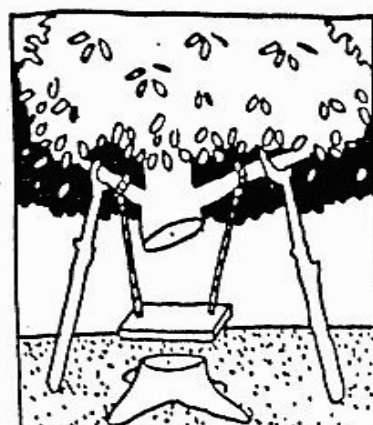
2. načrt podatkovne strukture



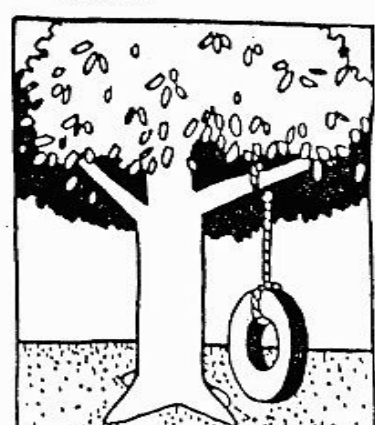
3. rezultat sistemske analize



4. izdelek programerjev



5. končna izvedba po odpravi manjših težav



6. dejanske potrebe uporabnikov

Proces izdelave programske opreme se izvaja v več fazah, ki jih imenujemo življenjski cikel. Osnovni model življenjskega cikla je klasični življenjski cikel, katerega značilnost je zaporednost izvajanja posameznih faz.

Seveda pa klasični življenjski cikel izdelave programske opreme ni ustrezen v vseh primerih, zato so se kasneje razvile nove alternativne metode in orodja za razvijanje informacijskih sistemov kot so (Rainer, Turban in Potter 2007, 318-321):

- prototipna metoda (prototyping)
- skupno aplikativno načrtovanje (joint application design)
- celostno računalniško-podprta programska upravljalna orodja (integrated computer-assisted software engineering tools)
- hiter aplikativen razvoj (rapid application development)
- razvoj za končnega uporabnika (end-user development)
- objektno-orientiran razvoj (object-oriented development)
- objektno-orientirana analiza in načrtovanje (object-oriented analysis and design)

V nadaljevanju bodo na kratko opisani trije osnovni pristopi pri načrtovanju in gradnji informacijskega sistema. To so linearni, prototipni in objektni pristop.

2.1.1 Linearni pristop

Pri linearnem pristopu se nobena faza ne more začeti, dokler ni v popolnosti dokončana predhodna faza. Značilnosti linearnega pristopa so (Kovačič in Vintar 1994, 45):

- razvoj IS poteka po kaskadnem principu skozi natančno določene faze, ki si sukcesivno sledijo,
- vsaka razvojna faza je natančno definirana in podrobno dokumentirana,
- modelni mehanizmi so razmeroma preprosti in temeljijo na analitskih in dokumentacijskih tehnikah ter večinoma verbalnih opisih obravnavanega problema,
- iz začetne analitične faze obravnavanega problema se praviloma izvrši neposreden prehod v izdelavo izvedbenega modela IS, ki je praviloma nenatančna, neformalizirana in zato vir mnogih napak.

Osnovni model linearnega pristopa je klasični življenjski cikel. Naročilo in izdelava programske opreme mu lahko sledita, sestavljen pa je iz naslednjih faz (Potočnik in drugi 1996, 188):

- zahteve uporabnika,
- specifikacije zahtev,

- analize,
- zasnove,
- kodiranja,
- testiranja in
- uvajanja ter vzdrževanja.

Primeri uporabe posameznih faz bodo predstavljeni kasneje pri opisu naročila izdelave nove aplikacije za Zavod RS za zaposlovanje.

Linearni pristop ima tudi nekaj slabosti, ki se kažejo v tem, da posameznih faz ni možno popolnoma opredeliti niti ni možno določiti natančnih prehodov iz ene v drugo fazo. Velikokrat se posamezne faze ne opravijo dovolj natančno, pomanjkljiva pa je tudi dokumentacija. Velikokrat se je potrebno vračati nazaj in posamezne faze ponovno opredeljevati in spreminjati dokumentacijo.

2.1.2 Prototipni pristop

Prototipni pristop je nastal kot odgovor na slabosti linearnega pristopa. Značilnost prototipnega pristopa (metode) je, da se najprej izgradi prototip, ki se s testiranjem, dopolnjevanjem in razširjanjem kasneje spremeni v uporaben izdelek.

Preden se izdela celoten sistem, se s preprostim prototipom uporabnikom oz. naročniku pokažejo osnove funkcije sistema, vhodi, izhodi, pri tem se zanemarijo detajli in različne kontrole. V tesni interakciji z uporabniki se poskuša izdelati prototip sistema, ki se dopolnjuje in spreminja ali celo zavrže in izdela nov. Prototip se izpopolnjuje toliko časa, da se ga privede do končne rešitve, ki zadovoljuje potrebe uporabnikov (Gradišar in drugi 2005, 269).

Za prototipni pristop so značilne naslednje prednosti (Srića, Treven in Pavlić 1995,207):

1. Uporabnik je vključen v projektiranje in gradnjo informacijskih sistemov, zato sprejema nov sistem z več zaupanja.
2. Projekt se hitro in enostavno izdela na računalniku s pomočjo prototipnih orodij oziroma programskih jezikov četrte generacije, programi so učinkovitejši.
3. Programi, ki so izdelani s pomočjo prototipnih orodij, se lahko izločijo iz uporabe ter se razvijejo novi; to ni mogoče v primeru klasičnega življenjskega ciklusa. Projekti so hitreje izdelani.
4. Za informacijske sisteme se uporablja standardna dokumentacija.
5. Uporabnik predvideva, da je mogoče preprosto in hitro razviti informacijski sistem, kar povzroča težave, če je potreben klasični razvoj sistema.

Prototipni pristop se uporablja kadar zahteve niso povsem jasno določene (Solina 1998, 14). Prototip mora biti hitro in poceni narejen s pomočjo raznih razvojnih

orodij, pri tem se implementirajo le nekatere funkcije, uporabi pa se lahko že obstoječ produkt.

Prototipni pristop ima tudi nekatere slabosti:

- zanašamo se na intuicijo,
- ne sili nas v sistemiziranost in vodenje dokumentacije o opravljenem delu,
- ni jasno določenih faz,
- na koncu nimamo dobre dokumentacije, ki bi omogočala vzdrževanje rešitev.

2.1.3 Objektni pristop

Objektni pristop se bistveno razlikuje od doslej uveljavljenih pristopov k razvoju informacijskega sistema. Modelov ne ločujemo več na podatkovne in procesne, pač pa obravnavamo oba vidika skupaj (Gradišar in drugi 2005, 269). S tem se rešujejo nekateri klasični problemi razvoja, kot so težave s konsistentnostjo, težave s tipizacijo osnovnih gradbenih blokov sistema, ki so za enkratno uporabo, daljšim potrebnim časom in večjimi stroški za razvoj sistema.

Ta pristop temelji na treh temeljnih konceptih (Kovačič in Vintar 1994, 49):

Objekti, ki vsebujejo podatkovne strukture in pripadajoče postopke na teh strukturah. Objekti so največkrat objekti, ki nastopajo v realnem svetu, pri kovencionalnem pristopu jih obravnavamo kot entitete.

Sporočilih, ki so sredstva, s pomočjo katerih objekti komunicirajo med seboj pri izvajanju poslovnih postopkov.

Tipih objektov, ki omogočajo realizacijo konceptov abstrakcije, ki jih poznamo iz podatkovnih modelov, kot so klasifikacija in generalizacija.

Prednosti objektnega pristopa so:

- večkratna uporaba istih objektov skrajša razvojni čas in zmanjša stroške,
- ker uporabljamo standardizirane objekte je tudi zanesljivost večja in kakovost boljša,
- objekti so zaključene celote, zato se lahko individualni objekti spreminjajo, ne da bi s tem spremenili druge objekte,
- nove rešitve bodo sestavljene iz že obstoječih in preizkušenih gradbenih blokov (objektov).

Objektni pristop je najbolj uveljavljen na področju programskih jezikov.

Unified Modeling Language (UML) je jezik za (Rupnik 2005, 147):

- specifikacijo,

- vizualizacijo,
- konstrukcijo in
- dokumentacijo.

Jezik UML je bil kmalu po svoji predstavitvi in vpeljavi privzet za standardni jezik modeliranja programskih sistemov, hkrati pa je bil enako dobro sprejet tudi v poslovnem svetu. Namen objektno orientiranega modeliranja poslovnih procesov je ustvariti opise in abstrakcije kompleksne in zapletene stvarnosti ključnih funkcij poslovanja ali združbe. Jezik UML je dokazal svojo uporabnost tudi na drugih tehnoloških in poslovnih področjih, pri čemer najbolj izstopa ravno modeliranje poslovnih modelov in procesov.

Jezik UML 2.0 se deli na tri dele, ki tvorijo celoto. Ti trije deli so (Object Management Group 2007, 9):

- UML 2.0 infrastruktura,
- UML 2.0 jezik za specifikacijo objektov (Object Constraint Language) in
- UML 2.0 diagramске tehnike.

UML 2.0 specifikacije so prikazane v naslednjih dokumentih:

- UML Infrastructure Specification² – v tem dokumentu je opisana terminologija UML jezika, ki določa arhitekturo in temeljne konstrukte jezika,
- UML Superstructure Specification³ – v tem dokumentu je opisan uporabniški vidik UML jezika, v njem so opisane diagramске tehnike in pravila uporabe, podaja pa tudi napotke za predstavitev,
- UML Diagram Interchange Specification⁴ – ta dokument določa izmenljiv format UML modelov,
- UML OCL Specification⁵ – jezik za specifikacijo omejitev in definiranje pravil.

Jezik UML standardizira zapis oziroma beleženje poslovnega procesa, vendar ne standardizira procesa opisovanja (poslovnih) procesov. UML uporablja simbole in nabor pravil, ki usmerjajo uporabo tega jezika. Pravila so sintaktična ali skladna (določajo videz in način uporabe geometričnih simbolov), semantična ali pomenoslovna (opisujejo pomen uporabljenih simbolov, bodisi kot samostojne simbole ali kot skupine in različne kombinacije simbolov) in pragmatična ali stvarna (določajo način uporabe jezika).

² Dokument se nahaja na spletni strani Object Management Group in je dosegljiv na spletnem naslovu: <http://www.omg.org/cgi-bin/doc?formal/05-07-04>. [15. 02. 2007].

³ Dokument se nahaja na spletni strani Object Management Group in je dosegljiv na spletnem naslovu: <http://www.omg.org/cgi-bin/doc?formal/05-07-05>. [15. 02. 2007].

⁴ Dokument se nahaja na spletni strani Object Management Group in je dosegljiv na spletnem naslovu: <http://www.omg.org/cgi-bin/doc?formal/06-04-04>. [15. 02. 2007].

⁵ Dokument se nahaja na spletni strani Object Management Group in je dosegljiv na spletnem naslovu: <http://www.omg.org/cgi-bin/doc?formal/06-05-01>. [15. 02. 2007].

Jezik modeliranja UML sestavlja 13 diagramskih tipov, pri čemer vsak diagram prikazuje specifične statične in dinamične vidike sistema. Diagrami so razdeljeni v tri skupine: strukturne diagrame, vedenjske diagrame in diagrame interakcije (Object Management Group 2007).

Strukturni diagrami so:

1. Razredni diagram (Class Diagram) – Razredni diagram prikazuje statično strukturo modela sistema. Struktura je sestavljena iz razredov in odnosov oziroma povezav. Razredi lahko predstavljajo in strukturirajo informacije, proizvode, listine ali organizacijo.
2. Objektni diagram (Object Diagram) – Objektni diagram izraža možne objektne kombinacije specifičnega razrednega diagrama. Uporablja se za ponazarjanje primerov razrednih diagramov.
3. Diagram sestavnih delov (Component Diagram) – Diagram sestavnih delov je poseben primer razrednega diagrama, ki se uporablja za opis sestavnih delov znotraj sistema programske opreme.
4. Diagram sestavljenih struktur (Composite Structure Diagram) – Diagram sestavljenih struktur je diagram, ki prikazuje strukturo notranjih razvrstitev, vključno z njihovo interakcijo z ostalimi deli sistema.
5. Paketni diagram (Package Diagram) – Paketni diagram prikazuje kako se sistem razcepi na logične skupine s prikazom odvisnosti med njimi. Prikazuje logično hierarhično strukturo sistema.
6. Diagram vpeljave (Deployment Diagram) – Diagram vpeljave je posebna zvrst razrednega diagrama, uporabljena za opis strojne opreme znotraj programske opreme.

Vedenjski diagrami:

7. Diagram primerov uporabe (Use Case Diagrams) – Diagram uporabe prikazuje odnose med uporabljenimi primeri. Vsak tak primer, tipično določen z besedilom, opisuje del funkcionalnosti celotnega sistema.
8. Diagram dejavnosti (Activity Diagram) – Diagram dejavnosti opisuje dejavnosti in aktivnosti, ki se dogajajo znotraj sistema.
9. Diagram stanja (State Machine Diagram) – Diagram stanja prikazuje zaporedje stanj, ki jih lahko nek objekt ima med svojim življenjem, vključno z dogodki, ki povzročajo spremembo stanj.

Diagrami interakcije:

10. Diagram zaporedja (Sequence Diagram) – Sekvenčni diagram prikazuje eno ali več zaporedij sporočil, ki se pošiljajo med različnimi nabori objektov. Je diagram, ki obsega časovno sosledje obvestil med objekti.
11. Diagram komunikacije (Communication Diagram) – Diagram komunikacije prikazuje množico interakcij med izbranimi objekti v podani situaciji. Pozornost je pri tem usmerjena na razmerja med objekti in na njihovo topologijo.
12. Časovni diagram (Timing Diagram) – Časovni diagram prikazuje vedenje objektov v časovnem obdobju.

13. Pregledni diagram interakcije (Interaction Overview Diagram) – Pregledni diagram interakcije opisuje le en, glavni, tok dogodkov. Za prikaz več tokov je potrebno za vsakega uporabiti svoj pregledni diagram interakcije.

Osnovni principi objektne usmerjenosti so abstrakcija, ograjevanje, modularnost in hierarhija. S pomočjo abstrakcije se zanemarijo podrobnosti. Osredotočiti se je potrebno predvsem na bistvene in skupne lastnosti objektov. Osnovne abstrakcije pri objektno orientiranem podatkovnem modeliranju so generalizacija, specializacija, agregacija in klasifikacija (Prabhu 2005, 3). Pri generalizaciji se posameznim razredom priredi bolj splošen razred na višjem nivoju. Zanj je značilno, da se lastnosti posplošenega razreda dedujejo po hierarhiji navzdol. Specializacija deluje ravno obratno. Splošen razred se bolj podrobno okarakterizira oziroma se ga bolj podrobno opredeli v določeno manjšo, lažje obvladljivo, skupino. Pri agregaciji se objektom, ki predstavljajo dele sestavljenega objekta, priredi sestavljeni objekt na višjem nivoju. Klasifikacija je koncept abstrakcije, pri katerem se posameznim objektom, ki imajo neko skupno lastnost, priredi tip objekta ali razred, ki predstavlja vse primerke s to lastnostjo (Kovačič in Vintar 1994, 82).

Osnovni koncepti objektno orientiranega pristopa so na Wikipedii (2007) povzeti po Deborah J. Armstrong (2006, 123–128) naslednji:

1. Razred – je opis skupine objektov z enakimi lastnostmi (atributi), enakim obnašanjem, povezavami in pomenom. Več podobnih objektov specifikiramo z istim splošnim opisom.
2. Objekt – je konkretna ali abstraktna posamezna oblika v razredu. Je združba podatkov in metod, ki se lahko izvajajo.
3. Metoda – je sposobnost objekta za opravljanje določene naloge. Metoda je dovoljena operacija, ki se na objektu lahko uporablja. Uporaba metode vpliva samo na tisti objekt, ki mu je metoda namenjena. Metoda je implementacija neke operacije razreda.
4. Sporočilo – je način komuniciranja objektov med seboj. S pomočjo sporočil se na objektih sproži izvajanje metod. Sporočila vsebujejo metode in oznako objekta na katerem se metode izvajajo.
5. Dedovanje – je pravilo, da se lastnosti objektov dedujejo po hierarhiji navzdol. Dedujejo se atributi in vedenje staršev. Z dodajanjem novih objektov v podrazrede tudi ti objekti dedujejo attribute in lastnosti višje ležečega razreda.
6. Ograjevanje – je način ločevanja (skrivanja) delovanja objektov pri uporabi programske kode oziroma pošiljanja sporočil. Program ima zunanji vmesnik, na katerem je vidno, kaj se je spremenilo, ne vidi pa se, kako je bilo to narejeno.
7. Abstrakcija – poenostavlja kompleksne realne sisteme z modeliranjem razredov. Pri abstrakciji zavestno zanemarimo podrobnosti. Abstrakcija pomeni, da skupino podobnih objektov obravnavamo na enak način.
8. Polimorfizem – pomeni, da se lahko različni razredi objektov odzivajo na isto sporočilo na različne načine (mnogoličnost). Pomeni tudi, da je ista operacija implementirana na več različnih načinov oziroma zanjo obstaja več metod.

Objektno orientiran pristop se uporablja za modeliranje zapletenih sistemov. Z njegovo pomočjo je predstava kompleksnih sistemov precej bolj enostavna in lažje razumljiva.

2.2 Metode nabave programske opreme

2.2.1 CMMI⁶

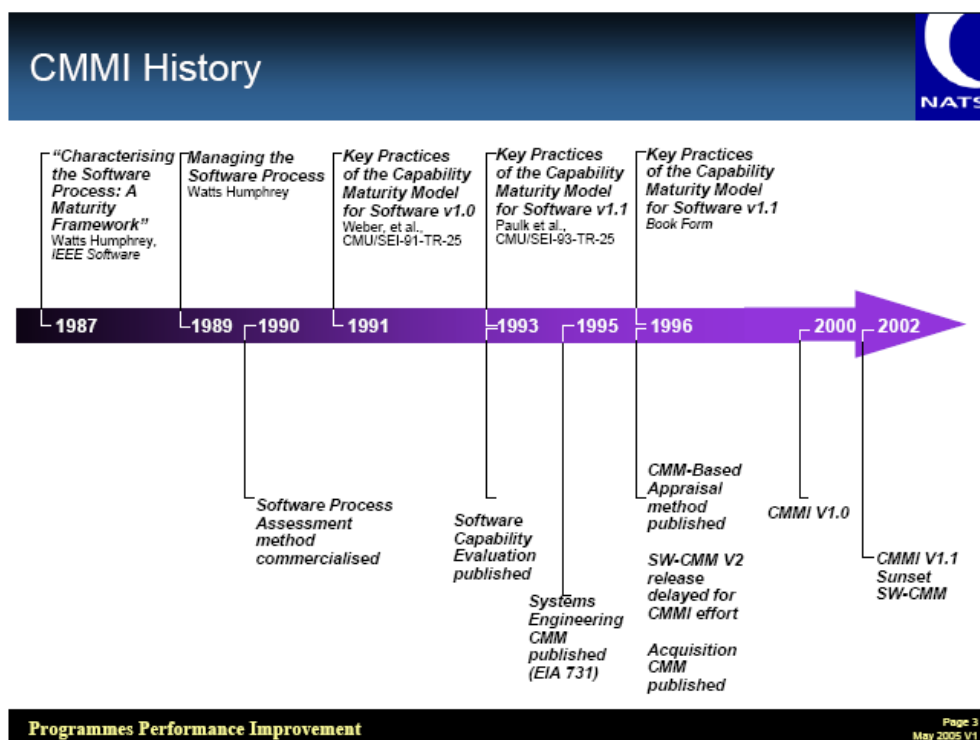
CMMI (The Capability Maturity Model Integration) metoda je metoda za ovrednotenje kakovosti procesa razvoja programske opreme. Sprva je bila ta metoda namenjena za ocenjevanje razvijalcev programske opreme za vladne organizacije, kasneje pa se je razvila v splošno uporabno metodo. Namen združenega zmožnostno zrelostnega modela (CMMI) je priskrbeti navodila za razvoj programske opreme in storitev. Vsebuje najboljše postopke pri razvoju in vzdrževanju aktivnosti v življenjskem ciklu programske opreme od ustvarjanja preko dobave do vzdrževanja.

V zadnji verziji modela je združeno znanje, ki je bilo v preteklosti ločeno in je bistveno za razvoj in vzdrževanje programske opreme, kot na primer programski inženiring, sistemski inženiring, strojna oprema in oblikovalski inženiring, ostali inženiringi ter nabavljanje programske opreme (SEI 2006, i).

CMMI model postopkovno in procesno opisuje razvoj programske opreme in storitev. Razvil se je iz CMM modela in je njegova nadgradnja.

⁶ Povzeto po: CMMI for Development 2006. in po CMMISM 2002.

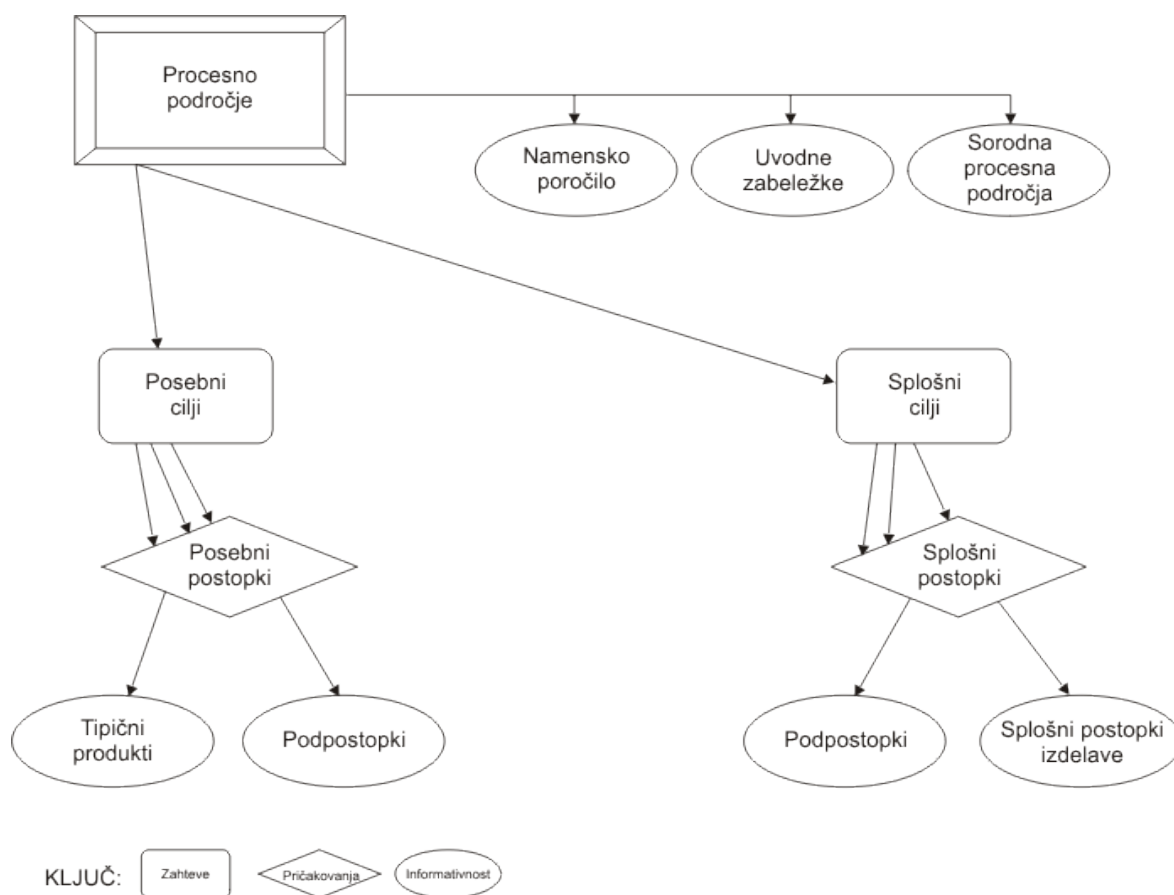
SLIKA 2: RAZVOJ CMMI MODELA



Vir: Civil Aviation Authority 2007

V letu 2006 je izšla trenutno [22. 02. 2007] zadnja verzija CMMI modela – CMMI for Development, Version 1.2.

Združene komponente CMMI modela, ki so opisane v drugem poglavju CMMI for Development, so predstavljene na naslednji sliki (SEI 2006, 17):

SLIKA 3: KOMPONENTE CMMI MODELA

Vir: CMMI for Development, Version 1.2. (SEI 2006, 17)

Procesno področje

Procesno področje je zbir sorodnih postopkov, ki so skupno implementirane, za zadovoljitev zahtev po pomembnosti in izboljšanju. V njem je združenih 22 procesnih področij. Ta področja so (SEI 2006, 18): vzorčna analiza in zaključki, konfiguracijski management, odločitvena analiza in zaključki, integrirano vodenje projektov, merjenje in analiza, organizacijska prenova in razvoj, organizacijska definicija procesov, izboljšanje procesov v organizaciji, izvedba organizacijskega procesa, izobraževanje v organizaciji, produktna integracija, projektno spremljanje in nadzor, planiranje projektov, zagotavljanje kakovosti procesov in proizvodov, kvantitativno projektno vodenje, razvojne zahteve, vodstvene zahteve, rizično vodenje, upravljanje s pogodbami dobaviteljev, tehnične rešitve, validacija⁷ in verifikacija⁸.

⁷ Validacija je proces vrednotenja programske opreme na koncu njenega razvoja z namenom, da se ugotovi skladnost izdelane programske opreme z zahtevami (Dogša 1993, 17) povzeto po IEEE pojmovniku, 1990b.

⁸ Verifikacija je proces, pri katerem se preverja, ali produkt tekoče faze ustreza zahtevam, postavljenim v predhodni fazi. Verifikacija je formalno dokazovanje pravilnosti programov (Dogša 1993, 13) povzeto po IEEE pojmovniku, 1990b.

Namensko poročilo

Namensko poročilo je poročilo procesnega področja in je informativna komponenta CMMI modela.

Uvodna zabeležka

Uvodna zabeležka opisuje osnovne koncepte procesnega območja in je informativna komponenta CMMI modela. Na primer uvodna zabeležka procesa planiranja projekta je: „Planiranje se prične z zahtevami, ki definirajo produkt in projekt.“

Sorodna procesna področja

Sestavni del sorodnih procesnih področij so sezname napotkov do sorodnih procesnih področij in odsevajo visok nivo relacij med procesnimi področji. Sorodna procesna področja so tudi informativna komponenta.

Posebni cilji

Pri posebnih ciljih so opisane unikatne značilnosti, ki morajo predstavljati zadovoljitev procesnega območja. Posebni cilji so zahtevana komponenta modela in so v pomoč pri ugotovitvah, če je procesno področje zadovoljeno. Samo posebni cilji so obvezna komponenta modela.

Splošni cilji

Splošni cilji se imenujejo splošni zato, ker ugotovitve za en cilj lahko veljajo na več procesnih področjih. Splošni cilj je opisan kot cilj v sedanjosti. Je nujen sestavni del pri oceni, če je procesno območje zadovoljeno. Obvezen sestavni del splošnih ciljev je poročilo.

Posebni postopki

Poseben postopek je opis aktivnosti ki je pretehtan pomemben v doseganju združenega posebnega cilja. Posebni postopki opisujejo aktivnosti, da so pričakovani rezultati v doseganju posebnih ciljev procesnega področja. Posebni postopek je pričakovana komponenta modela. Le poročilo posebnega postopka je pričakovan sestavni del modela. Naslov posebnega postopka in zabeležke skupaj s posebnim postopkom so pomemben sestavni del modela.

Tipični produkti

Pod tem terminom je razumljen katerikoli produkt, ki se ga naredi pri razvoju programske opreme (datoteke, dokumenti, posamezni deli izdelkov, storitve, procesi, specifikacije, sezname itd.). Tipični produkti so lahko preprosti sezname, ki jih dobimo kot izhod iz posebnih postopkov. Tipičen produkt se imenujejo zato, ker so pogosto na tem mestu tudi drugi produkti, ki so ravno tako učinkoviti, vendar niso ustrezno dokumentirani.

Podpostopki

Podpostopek je podroben opis, ki zagotavlja vodila za pojasnjevanje in izvajanje posebnih ali splošnih postopkov. Podpostopki so lahko potrjeni, kot predpisani, in so dejansko informativna komponenta za ideje, ki so lahko uporabne za izboljšanje postopkov.

Splošni postopki

Splošni postopki se imenujejo „splošni“, ker se lahko enak postopek nanaša na več procesnih območij. Splošni postopek je opis aktivnosti, ki je preudarjeno pomemben v doseganju povezanih splošnih ciljev. Splošen postopek je pričakovan sestavni del modela. Za zmanjšanje števila ponavljajočih se informacij, je dovolj, da se prikaže samo naslov splošnega postopka, poročilo in dopolnitve o procesnih območjih.

Splošni postopki izdelave

Splošni postopek izdelave se pojavi po splošnem postopku in v procesnem območju nudi usmeritve, kako morajo biti splošni postopki uporabljeni edinstveno v procesnem področju. Splošni postopek izdelave je informativna komponenta modela. Na več mestih so nadaljnje informacije potrebne za opis zamisli. To informativno gradivo je sestavljeno iz zabeležke, primerov, razširitev in priporočil.

CMMI metoda je predstavljena na dva enakovredna načina:

- nivojsko (staged), kjer je predstavljenih več nivojev zrelosti procesa organizacije ter na vsakem nivoju procesna področja in cilji in
- zaporedno (continous), kjer je definirana zrelost posameznih procesnih področij glede na dosežene cilje oziroma uporabljene tipične postopke.

Ta dva načina sta združena na dveh ravneh in sicer v **zmožnostni** in **zrelostni** ravni. Zmožnostna in zrelostna raven sta sestavljeni iz več nivojev, in sicer zmožnostna iz šestih, zrelostna pa iz petih nivojev. Primerjava obeh ravni nam pokaže, da se razlikujeta samo v prvih dveh nivojih in sicer ima zmožnostna raven nivo 0 – nedokončan nivo, zrelostna raven pa tega nivoja nima. Nivo 1 je pri zmožnostni ravni izvršilni nivo, pri zrelostni ravni pa je to začetni nivo. Ostali nivoji so potem enaki za obe ravni in sicer: nivo 2 – upravljalni, nivo 3 – definiran, nivo 4 – kvantitetno voden in nivo 5 – optimiziran nivo.

Zmožnostna raven opisuje osnovno zaporedje za nepretrgan proces izboljšav in določitev meril za merjenje zmožnosti procesov in poslovnih ciljev v organizaciji. Omogoča primerjavo med organizacijami na procesnih področjih z uporabo organizacijskih področij. Sestavlja jo 6 nivojev:

0. **Nedokončan nivo.** Ta je predstavljen na nivoju 0. Nedokončan proces je proces, ki ni dokončan oziroma je delno končan. En ali več posebnih ciljev procesnega področja ni izpolnjenih in noben splošni cilj na tem nivoju ni dodelan.

1. **Izvedbeni nivo.** Prvi zmožnostni nivo je opisan kot izvedbeni nivo. Ta nivo je namenjen uresničevanju posebnih ciljev procesnega področja. Podpira in omogoča izdelavo tipičnih produktov. Čeprav so rezultati izvedbenega nivoja pomembni pri izboljšanju poslovanja, pa ta izboljšanja lahko pomenijo izgubljen čas, če ne privedejo organizacije v višji nivo.
2. **Upravljalni nivo.** Upravljan postopek je izveden postopek iz izvedbenega nivoja in zagotavlja osnovno infrastrukturo za podprti postopek. Načrtovan in izveden je v skladu z politiko; zaposlujejo se izkušeni ljudje in angažira se tiste, ki imajo primerne vire za izdelavo kontroliranih produktov; vključuje ustrezne partnerje; je spremljan, kontroliran, in ocenjen; in je ocenjen za privrženost k njegovemu postopkovnemu opisu. Postopkovna disciplina na drugem sposobnostnem nivoju pomaga zagotoviti, da se obstoječi postopki obdržijo v času večje napetosti.

Drugi zmožnostni nivo je označen kot upravljalni proces. Upravljeni proces je predstavljen proces, ki ima osnovno infrastrukturo na mestu za podporo procesu. Je planiran in izvršen skladno s politiko. Lahko zaposljuje ljudi, ki imajo aдекватne resurse za proizvodno kontrolirane izhode, vključno z verodostojnimi osebami, so spremljani, kontrolirani in ponovno preverjeni, so podvrženi ocenjevanju opisov procesov. Vsebina procesa se odraža v zmožnostnem nivoju 2, ki pomaga zagotoviti, da se obnese tudi v izjemnih situacijah in pod velikimi obremenitvami.

Organizacije imajo vgrajene osnovne programske upravljalne kontrole. Programski managerji na projektu spremljajo stroške, terminski plan in funkcionalnosti. Produkti, razviti za zadovoljevanje programskih potreb, so v skladu z osnovno linijo plana in njihova integriteta je kontrolirana. Standardi programskih projektov so definirani in organizacija zagotavlja upoštevanje teh standardov. Programski projekti, ki se izdelujejo s podizvajalci, prispevajo k trdnemu odnosu med naročnikom in izvajalcem. Sposobnost programskih procesov na drugem nivoju je disciplinirana zaradi planiranja in sledenja. Programski projekti so stabilni in uspešni, nadaljnji projekti pa se lahko ponavljajo. Projektni procesi so pod kontrolo projektnega vodenja, sledijo realnim planom, osnovanim na osnovi predhodnih projektov.

3. **Definiran nivo.** Je standardni proces za razvoj in vzdrževanje programske opreme in je prilagojen organizaciji s standardnimi procesi. Prispeva delovne izdelke, mere in druge postopke izboljšanja podatkov pri organizacijskem postopku.

Vključuje oboje, programski inženiring in managerske procese, ki so dokumentirani, standardizirani in vgrajeni v nepretrgano celoto programerskega procesa. Ta nivo je standarden proces CMMI modela. V tem

se tudi razlikuje od nivoja 2, saj zahteva na enak način natančen opis in definira postopke.

Postopki so pri zmožnostni ravni 3 opisani bolj strogo, kot je to pri zmožnostni ravni 2. Definiran postopek izkazuje jasen namen, vhode, vhodne kriterije, aktivnosti, vloge, izmere, verifikacijske korake, izhode in izhodne kriterije.

4. **Kvantitetno upravljalni nivo.** Organizacija določi kvalitativne in kvantitativne cilje za programske produkte in procese. Produktivnost in kvaliteta se merita pri vseh pomembnih aktivnostih programskih procesov na vseh projektih kot del merskega programa organizacije. Kvantitetno upravljalni nivo je definiran postopek, ki je statistično kontroliran z uporabo statističnih in drugih kvantitetnih tehnik. Programski procesi so merjeni z določenimi merili. Ta merila so temelj za kvantitetno ovrednotenje projektov programskih procesov in produktov. Kakovostna in procesna predstavitev je predstavljena v statističnih odnosih in je vodena čez ves življenjski cikel.

Kontrola projektov se izvaja preko produktov in procesov. Programski proces na tem nivoju je predvidljiv, ker je proces merjen. Ta nivo omogoča organizaciji, da predvidi trende v procesih in kvaliteti produktov. Ko se limit odstopanj prekorači, se sproži akcija za spremembo situacije. Programski produkti so predvidljivo visoke kvalitete.

5. **Optimiziran nivo.** Optimiziran postopek je kvantitetno uspešen postopek (sposobnostna raven 4), ki je izboljšan na podlagi razumevanja skupnih razlogov za odstopanja neločljivo povezanih v postopek. Pri optimiziranju se je potrebno osredotočiti na izboljšanje programskih postopkov od procesnih predstav do progresivnih in inovativnih izboljšanj.

Celotna organizacija je osredotočena na kontinuirano zaupanje v proces. Cilj organizacije je odkriti slabosti in prednosti procesa s ciljem preventive pred napakami. Podatki o učinkovitosti programskih procesov se uporabljajo za predstavitev poslovnega rezultata nove tehnologije in so podlaga za spremembe programskega procesa v organizaciji.

Programski projektni tim analizira napake, da lahko določi razloge zanje. Programski procesi so ovrednoteni za zaščito pred ponovitvami znanih tipov napak. Naučene ugotovitve tvorijo bazo znanja za ostale projekte.

Zmožnostni nivoji procesnega področja so doseženi preko uporabe splošnih postopkov ali ustreznih alternativ. Podjetje, ki se nahaja na nivoju 0 nima dodelanih procesov oziroma so procesi delno opisani. Dodelanih nima niti splošnih ciljev in jih zaradi tega tudi ne izpolnjuje. Zmožnostni nivo 1 doseže podjetje, ki svoje postopke vodi na način, kjer se izvedba nečesa samo zahteva. Opredeljenih ni nobenih postopkov, niti pravil, temveč je opredeljen le končni cilj. Drugi nivo doseže podjetje, ki ima že opredeljeno, kako se postopki opisujejo in ima

opredeljene naloge za spremljanje, kontroliranje in ocenjevanje opravljenega dela. Zmožnostni nivo 3 doseže podjetje, ki ima procese standardno opredeljene. Četrty nivo doseže organizacija, ki poleg spremljanja, kontroliranja in ocenjevanja opravljenega dela uporablja še kvantitativne in statistične tehnike. Na tem nivoju se organizacija lahko primerja s ostalimi, saj so njeni procesi in produkti tudi kvantitativno ovrednoteni. Na petem nivoju pa se nahajajo organizacije, ki so svoje procese in produkte standardizirale, jih kvantitativno in statistično ovrednotile ter jih tudi optimirale. To ne pomeni, da je optimiran vsak podproces, temveč da so optimirani tisti, ki vplivajo na poslovne cilje podjetja.

Zrelostna raven opisuje razvojno pot od ad-hoc, kaotičnih procesov, do zrelih, urejenih programskih procesov. Opisuje osnovno zaporedje nepretrganega procesa izboljšav in določitev meril za merjenje zrelosti organizacijskih programskih procesov. Omogoča primerjavo med organizacijami na procesnih področjih z uporabo zrelostnih nivojev. Sestavlja jo 5 nivojev:

1. **Začetni nivo.** To je prvi zrelostni nivo in ustreza organizaciji, v kateri se ne uporablja skoraj nobenega pristopa, ki bi zagotavljal obvladovanje razvojnega okolja. Na tem nivoju organizacija običajno ne zagotavlja stabilnega okolja za razvoj in vzdrževanje programske opreme. Programska oprema se razvija priložnostno in neurejeno. Nekateri procesi sicer so definirani, uspeh pa je odvisen od posameznikov in njihovega truda.

Med krizo se projekti oddaljijo od planirane procedure in se spremenijo v kodiranje in testiranje. Uspeh je odvisen v celoti od sposobnosti vodje in učinkovitega programerskega tima.

Razvoj programske opreme na začetnem nivoju je nepredvidljiv, ker se stalno spreminja, glede na razvoj dela. Termiski plan, proračun, funkcionalnosti in kvaliteta dela so nepredvidljivi. Projekt je odvisen od posameznikov, njihovega znanja in motivacije, ne pa od organizacijske sposobnosti.

2. **Upravljalni nivo.** Za upravljanje razvoja programske opreme je na tem nivoju vzpostavljena osnovna projektna organizacija. Planiranje in upravljanje novih projektov temelji na izkušnjah iz ostalih projektov. Drugi nivo je namenjen učinkovitemu upravljanju razvoja programske opreme.

Organizacije imajo vgrajene osnovne upravljalne kontrole. Spremljajo stroške projekta, termiski plan in funkcionalnosti. Standardi razvoja programske opreme so definirani in organizacija zagotavlja upoštevanje teh standardov. Projekti, ki se izdelujejo s podizvajalci, prispevajo k trdnemu odnosu med naročnikom in izvajalcem. Projekti so stabilni in uspešni, nadaljnji projekti pa se lahko ponavljajo. Projektni procesi so pod kontrolo projektnega vodenja, sledijo realnim planom, osnovanim na predhodnih projektih.

3. **Definiran nivo.** Je standardni proces za razvoj in vzdrževanje programske opreme. Vključuje programski inženiring in managerske procese, ki so dokumentirani, standardizirani in vgrajeni v nepretrgano celoto programerskega procesa. Ker je programerski proces točno določen, ima management dober vpogled v tehnični razvoj na vseh projektih. Organizacija izkorišča učinkovite programersko inženirske postopke, kadar standardizira njihove programske procese. Obstajajo skupine, ki so odgovorne za programerske procese v organizaciji. Pridobljeno znanje na tem nivoju je dobra podlaga za prihodnost.

Ta nivo je standardni proces CMMI modela. Organizacija, ki se nahaja na tem nivoju je standardna, ker so aktivnosti inženiringa in managementa stabilne in ponovljive, so znotraj proizvodne linije, stroškov in terminskega plana, funkcionalnosti so pod kontrolo, kvaliteta programske opreme pa je spremljana. Ta procesna sposobnost je osnovana na skupnem, vseorganizacijskem razumevanju aktivnosti, vlog in odgovornosti v določenem programerskem procesu.

4. **Kvantitetno voden.** Organizacija določi kvalitativne in kvantitativne cilje za programske produkte in procese. Produktivnost in kvaliteta se merita za pomembne aktivnosti programskih procesov preko vseh projektov kot del merskega programa organizacije. Podatkovna baza vseorganizacijskega programskega procesa se uporablja za zbiranje in analizo podatkov, ki so na razpolago iz določenih projektnih programskih procesov. Programski procesi so merjeni z določenimi merili. Ta merila so temelj za količinsko ovrednotenje projektov programskih procesov in produktov. Vsi procesi in aktivnosti so statistično spremljane.

Kontrola projektov se izvaja preko produktov in procesov. Programski proces na tem nivoju je predvidljiv, ker je proces merjen. Ta nivo omogoča organizaciji, da predvidi trende v procesu in kvaliteti produkta. Ko se limit prekorači, se sproži akcija za spremembo situacije. Programski produkti so predvidljivo visoke kvalitete.

5. **Optimiziran nivo.** Celotna organizacija je osredotočena na kontinuirano zaupanje v proces. Cilj organizacije je odkriti slabosti in prednosti procesa s ciljem preventive pred napakami. Podatki o učinkovitosti programskih procesov se uporabljajo za predstavitev poslovnega rezultata nove tehnologije in so podlaga za spremembe programskega procesa v organizaciji.

Programerski projektni tim analizira napake, da lahko določi razloge zanje. Programski procesi so ovrednoteni za zaščito pred ponovitvami znanih tipov napak. Naučene ugotovitve tvorijo bazo znanja za ostale projekte.

Glede na uvrstitev organizacije v ustrezen nivo, lahko predvidimo, kakšna bo verjetnost, da bo razvoj programske opreme uspešen.

CMMI model obravnava 22 procesnih področij, ki jih lahko združimo v štiri kategorije in sicer:

- upravljanje procesov,
- upravljanje projektov,
- inženiring in
- podpora.

Vsakemu procesnemu področju ustrezajo splošni cilji in splošni postopki s katerimi se organizacija uvrsti v določen nivo. Za prehod iz enega nivoja na drug nivo mora organizacija izpolniti **vse** zahtevane cilje prejšnjega nivoja.

Procesna področja, njihove združene kategorije in zrelostni nivoji so prikazani v naslednji tabeli:

TABELA 1: PROCESNA PODROČJA, NJIHOVE ZDRUŽENE KATEGORIJE IN ZRELOSTNI NIVOJI

<i>Procesno področje</i>	<i>Kategorija</i>	<i>Zrelostn i nivo</i>
1. Vzorčna analiza in zaključki	Podpora	5
2. Konfiguracijski management	Podpora	2
3. Odločitvena analiza in zaključki	Podpora	3
4. Integrirano vodenje projektov	Upravljanje projektov	3
5. Merjenje in analiza	Podpora	2
6. Organizacijska prenova in razvoj	Upravljanje procesov	5
7. Organizacijska definicija procesov	Upravljanje procesov	3
8. Izboljšanje procesov v organizaciji	Upravljanje procesov	3
9. Izvedba organizacijskega procesa	Upravljanje procesov	4
10. Izobraževanje v organizaciji	Upravljanje procesov	3
11. Produktna integracija	Inženiring	3
12. Projektno spremljanje in nadzor	Upravljanje projektov	2
13. Planiranje projektov	Upravljanje projektov	2
14. Zagotavljanje kakovosti procesov in proizvodov	Podpora	2
15. Kvantitativno projektno vodenje	Upravljanje projektov	4
16. Razvojne zahteve	Inženiring	3
17. Vodstvene zahteve	Inženiring	2
18. Rizično vodenje	Upravljanje projektov	3
19. Upravljanje s pogodbami dobaviteljev	Upravljanje projektov	2
20. Tehnične rešitve	Inženiring	3
21. Validacija	Inženiring	3
22. Verifikacija	Inženiring	3

Organizacije se lahko odločajo za napretrgano ali pa za fazno predstavitev iz CMMI modela. Primerjava med obema modeloma je prikazana v spodnji tabeli.

TABELA 2: PRIMERJAVA NEPRETRGANE IN FAZNE PREDSTAVITVE

Nepretrgana predstavitev	Fazna predstavitev
Organizacija izbira procesna področja in možnostni nivo, na podlagi njenih ciljev.	Organizacija izbira procesna področja na podlagi zrelostnih nivojev.
Izboljšanje je merjeno s pomočjo možnostnih nivojev. Zmožnostni nivoji so: • merilo zrelosti posameznega procesa organizacije • rangirani od 0 do 5	Izboljšanje je merjeno s pomočjo zrelostnih nivojev. Zrelostni nivoji so: • merilo zrelosti več procesov v organizaciji • rangirani od 1 do 5
Zmožnostni nivoji se uporabljajo za doseganje ciljev in uporabo postopkov za izboljšanje izvedbe.	Zrelostni nivoji se uporabljajo za doseganje ciljev in uporabi postopkov za izboljšanje izvedbe.
Enakovredna primerjava z drugimi organizacijami dopušča organizaciji uporabo nenehnega izboljševanja za uvrstitev v željen zrelostni nivo.	Tu ni potrebe po enakovrednem mehanizmu za vrnitev v nepretrgano predstavitev.

CMMI - je načrtovan za podporo razvoja programske opreme bližje poslovnim zahtevam. S pomočjo CMMI metode se skušajo doseči naslednji cilji:

- bolj konkretno povezati vodstvene in tehnične dejavnosti s svojimi poslovnimi cilji
- razširiti uporabnost na celoten življenjski cikel produkta in tehničnih dejavnosti ter poskrbeti, da produkt izpolnjuje pričakovanja uporabnika
- vključiti izkušnje prejšnjih oz. ostalih projektov
- upoštevati zrelostne prakse
- preslikati organizacijske funkcije na produkte in storitve
- upoštevati ustrezne ISO standarde

CMMI je opisna metoda, ki opisuje bistvene lastnosti, ki jih lahko pričakujemo od programske opreme glede na nivo, v katerega organizacija sodi. S pomočjo uporabe CMMI metode je možno določiti in izboljšati procesne pomanjkljivosti v organizaciji.

2.2.2 COBIT⁹

Leta 1996 je organizacija revizorjev informacijskih sistemov (ISACA) izdala publikacijo Cobit (Control Objectives For Information and Related Technology – kontrolni cilji za informacijsko in sorodne tehnologije) in jo v kasnejših letih še izpopolnila v naslednjih izdajah. V Cobit-u so zbrani in urejeni cilji in postopki revidiranja informacijskih sistemov.

Cobit je orodje za pomoč managerjem, da razumejo in upravljajo nevarnosti, ki se pojavljajo z izvrševanjem nove informacijske tehnologije, vodjem informatike in kontrolorjem oz. revizorjem pri reviziji informacijskih sistemov. Temelji na najboljših mednarodnih postopkih v informacijskem managementu in kontroli.

Ogrodje metodologije predstavljajo podatki in informacije, procesi informacijske tehnologije pa morajo z uporabo virov informacijske tehnologije (ljudje, aplikativni sistemi, tehnologija, oprema in podatki) omogočiti, da so informacije v pravem trenutku na voljo poslovnemu procesu in da ustrezajo kriterijem kakovosti, zakonitosti in varnosti.

Cobit je nekakšen križanec med ostalimi upravljavskimi metodologijami. Pokriva širše območje kot CMMI. Pokriva infrastrukturne postopke in razvoj aplikacij. Poleg tega je Cobit prevzel CMMI-jevo razvojno ocenjevanje za organizacijske sporazume storitvenih standardov.

Po metodologiji Cobit so cilji revidiranja informacijske tehnologije razdeljeni v štiri področja in na 34 IT procesov, le-ti pa še na 304 kontrolne cilje (CIO 2007).

Za vsak kontrolni cilj Cobit definira:

- seznam faktorjev kritičnega uspeha, ki zagotavljajo netehnično najboljši postopek za vsak IT proces,
- ključni cilj in ključni kazalec ocene (izid meritev in predstavitev gonilnikov za vse IT procese) in
- zrelostne modele za pomoč pri merjenju in sprejemanju odločitev za izboljšanje sposobnosti.

Za vsak proces je izdelana metodologija pregleda, opis poslovnih zahtev, ki jih omogoča proces, način izvedbe kontrol procesa in kaj je potrebno proučiti oz. pregledati. Področja ciljev revidiranja informacijske tehnologije so (Micro Process 2007):

- načrtovanje in organiziranje informacijskih sistemov, strateško planiranje, definiranje informacijske arhitekture, tehnoloških usmeritev in upravljanje investicij,

⁹ Povzeto po: Cobit 4.0. 2005.

- pridobivanje, izbor, izvedba in uvajanje informacijskih sistemov, vzdrževanje,
- upravljanje in zagotavljanje delovanja, pomoč, servisi, varnost, zagotavljanje neprekinjenosti poslovanja, izobraževanje in usposabljanje uporabnikov ter
- opazovanje in nadzor procesov.

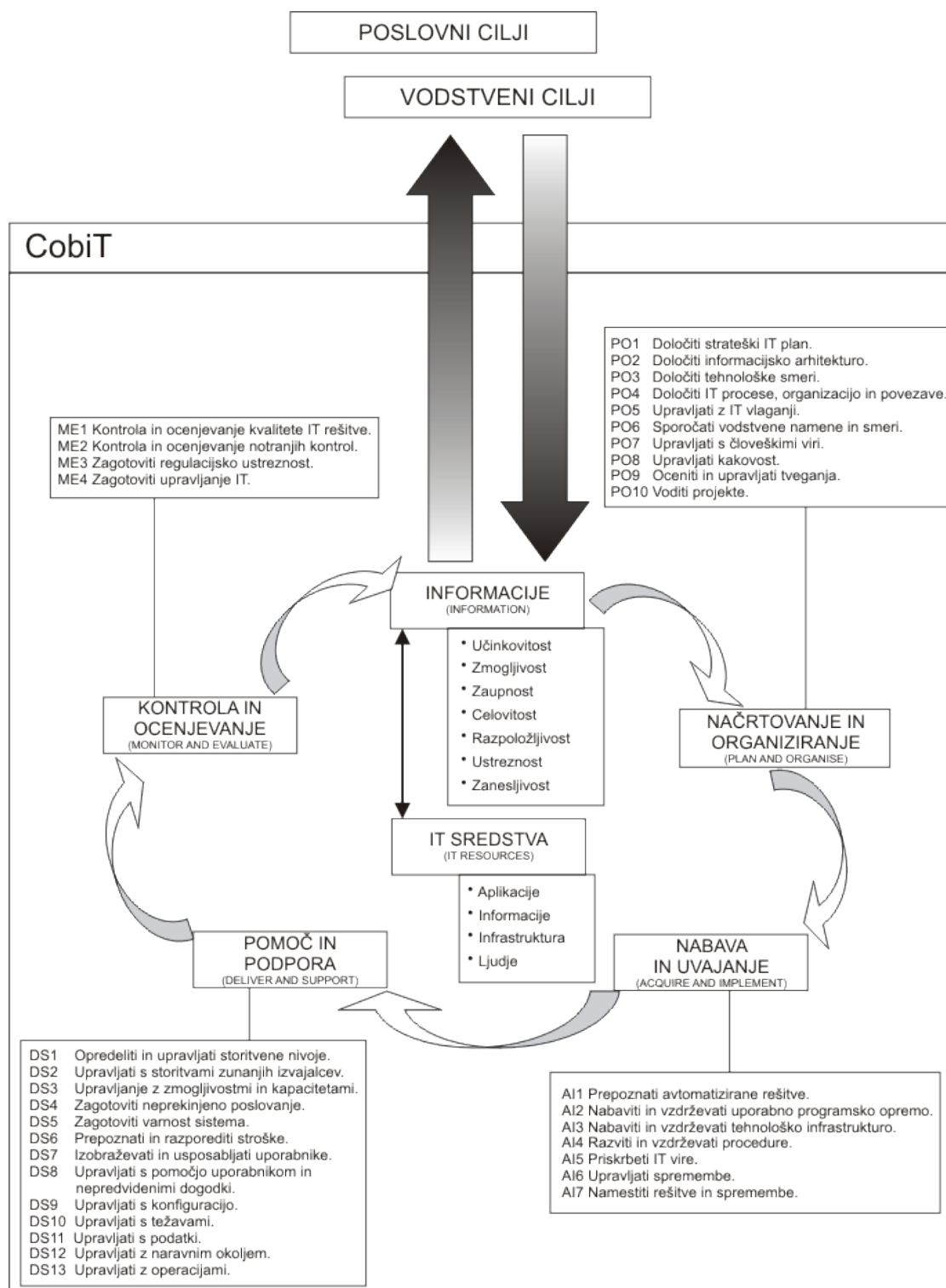
Cobit IT proces je definiran znotraj štirih področij: načrtovanja in organiziranja, nabave in uvajanja, pomoči in podpore ter kontrole in ocenjevanja. Osnova vseh štirih področij so informacije, ki vplivajo na poslovne in vodstvene cilje. Da zadovoljijo te cilje, morajo informacije izpolnjevati določena merila:

- **učinkovitost** – ustrezati morajo poslovnim procesom in biti pravočasne, pravilne in uporabne,
- **zmogljivost** – pripravljene morajo biti z optimalno uporabo sredstev informacijske tehnologije,
- **zaupnost** – varovane morajo biti pred razkritjem nepooblaščenim osebam,
- **celovitost** – biti morajo natančne in zadostne za obravnavan poslovni proces,
- **razpoložljivost** – biti morajo vedno na voljo,
- **skladnost** – biti morajo skladne z zakoni, predpisi, pogodbami, ki urejajo področje poslovnih procesov in
- **zanesljivost** – za odločanje.

Pridobivajo se s pomočjo sredstev informacijske tehnologije. Ta sredstva so:

- **aplikacije** – računalniške rešitve (avtomatizirani postopki poslovnih funkcij)
- **tehnologija** – strojna in sistemska programska oprema, sistemi za opravljanje baz podatkov, računalniške mreže...,
- **infrastruktura** – pomožne računalniške naprave in
- **ljudje** – njihovo znanje in izkušnje.

Vsa področja so med seboj zaporedno povezana. Na podlagi informacij, poslovnih in vodstvenih ciljev ter odločitev, se lahko prične z načrtovanjem in organiziranjem izvedbe informacijske rešitve. Na podlagi te priprave se pristopi k nabavi ali izdelavi in uvedbi informacijske rešitve. Naslednje področje, ki sledi, je pomoč in podpora uporabnikom IT rešitve, zadnje področje pa je kontrola in ocenjevanje rešitve. S tem je Cobit krogotok sklenjen. Vse pa se lahko ponovi, saj se na osnovi kontrole in ocenjevanja pridobijo nove informacije, ki ponovno vplivajo na poslovne in vodstvene cilje. Opisan proces je prikazan na naslednji sliki:

SLIKA 4: PRIKAZ COBIT PROCESA

Vir: Cobit 4.0 (2005, 25)

Vsak opis procesa vsebuje štiri odseke in sicer:

1. Odsek 1 vsebuje postopkovni opis povzetka procesnih ciljev z na najvišji ravni predstavljenimi kontrolnimi cilji.
2. Odsek 2 vsebuje podrobnejše kontrolne cilje za posamezen proces.
3. Odsek 3 vsebuje vhode v in izhode iz procesa, RACI¹⁰ grafe (z opredelitvijo odgovornosti, ocene, posvetovanja in/ali informiranja), cilje in merila.
4. Odsek 4 vsebuje zrelostni model za proces (po CMMI metodi).

V prvem odseku se opredeli pomembnost poslovnih zahtev in sicer se za vsak kriterij določi ali je primarnega ali sekundarnega pomena ali pa ne vpliva na proces. Poslovne zahteve so: uspešnost, učinkovitost, zaupnost, celovitost, razpoložljivost, skladnost in zanesljivost. Določi se kontroliranje IT procesa, določijo se poslovne zahteve za IT, določi se, na kaj je potrebno biti osredotočen, na kakšen način se zahteve dosežejo in kakšne meritve se bodo opravljale. Poleg tega se še označi, katera sredstva informacijske tehnologije se bodo potrebovala.

V drugem odseku se določijo podrobnejši kontrolni cilji.

V tretjem odseku se določijo vhodi, ki so lahko rezultat ostalih Cobit procesov ali vhodi, ki ne izhajajo iz Cobit procesov. Izходом je potrebno določiti, kateremu procesu bodo služili, kot vhod. RACI grafi so matrice med aktivnostmi in funkcijami, kjer je potrebno na presečiščih določiti ustrezne RACI opredelitve. Določijo se še cilji z merili in sicer: delovni cilji, ki se merijo s kazalniki učinkov, postopkovni cilji, ki se merijo s postopkovno-ciljnimi kazalniki in informacijsko tehnološki cilji, ki se merijo s kazalniki IT ciljev.

Sestavni del Cobit metode je, v četrtem odseku, tudi zrelostni model s svojimi šestimi nivoji (od 0 do 5), ki je bil opisan že pri CMMI metodi. Ta se opredeljuje v četrtem odseku in odgovarja na vprašanje, kje se podjetje nahaja sedaj, omogoča primerjavo z ostalimi podjetji in omogoča odločitve, kje podjetje želi biti. Vsak od teh 34 procesov lahko napreduje v naslednji nivo šele takrat, ko izpolni vse pogoje nižjega nivoja.

Procesi, ki jih opredeljuje Cobit model so naslednji:

Področje načrtovanja in organiziranja

- PO1 Določanje strateškega IT plan
- PO2 Določanje informacijske arhitekture
- PO3 Določanje tehnološke smeri
- PO4 Določanje IT procesov, organizacije in povezav
- PO5 Upravljanje z IT vlaganji
- PO6 Sporočanje vodstvenih namenov in smeri
- PO7 Upravljanje s človeškimi viri

¹⁰ Responsible, Accountable, Consulted and/or Informed

- PO8 Upravljanje kakovosti
- PO9 Ocenjevanje in upravljanje tveganj
- PO10 Vodenje projektov

Področje nabave in uvajanja

- AI1 Prepoznavanje avtomatiziranih rešitev.
- AI2 Nabavljanje in vzdrževanje uporabne programske opreme.
- AI3 Nabavljanje in vzdrževanje tehnološke infrastrukture.
- AI4 Razvijanje in vzdrževanje procedur.
- AI5 Zagotavljanje IT virov.
- AI6 Upravljanje sprememb.
- AI7 Nameščanje rešitev in sprememb.

Področje pomoči in podpore

- DS1 Opredeljevanje in upravljanje storitvenih nivojev.
- DS2 Upravljanje s storitvami zunanjih izvajalcev.
- DS3 Upravljanje z zmogljivostmi in kapacitetami.
- DS4 Zagotavljanje neprekinjenega poslovanja.
- DS5 Zagotavljanje varnosti sistema.
- DS6 Prepoznavanje in razporejanje stroškov.
- DS7 Izobraževanje in usposabljanje uporabnikov.
- DS8 Upravljanje pomoči uporabnikom in nepredvidenih dogodkov.
- DS9 Upravljanje konfiguracije.
- DS10 Upravljanje težav.
- DS11 Upravljanje podatkov.
- DS12 Upravljanje naravnega okolja.
- DS13 Upravljanje operacij.

Področje kontrole in ocenjevanja

- ME1 Kontroliranje in ocenjevanje kvalitete IT rešitve.
- ME2 Kontroliranje in ocenjevanje notranjih kontrol.
- ME3 Zagotavljanje regulacijske skladnosti.
- ME4 Zagotavljanje upravljanja IT.

SLIKA 5: PLANIRANJE IT PROCESOV NA IT VODSTVENIH PODROČJIH, COSO¹¹ PODROČJIH, COBIT IT SREDSTVIH IN COBIT INFORMACIJSKIH KRITERIJIH

		Pomembnost	IT vodstvena področja					COSO					Cobit IT sredstva				Cobit informacijski kriteriji							
			Strateška usmeritev	Vrednostna opredelitev	Upravljanje tveganj	Krizni management	Storitveni management	Obvladovanje okolja	Obvladovanje tveganja	Obvladovanje aktivnosti	Informacije in komunikacije	Spremljanje	Ljudje	Informacije	Aplikacije	Infrastruktura	Uspešnost	Učinkovitost	Zaupnost	Celovitost	Razpoložljivost	Skladnost	Zanesljivost	
Načrtovanje in organiziranje																								
PO1	Določiti strateški IT plan	V	P		S	S			P		S	S	✓	✓	✓	✓	P	S						
PO2	Določiti informacijsko arhitekturo	N	P	S	P	S			P	P			✓	✓			S	P	S	P				
PO3	Določiti tehnološke smeri	SR	S	S	P	S			S	P	S				✓	✓	P	P						
PO4	Določiti IT procese, organizacijo in povezave	N	S		P	P		P			S	S	✓				P	P						
PO5	Upravljalni z IT vlaganji	SR	S	P	S		S		S	P			✓		✓	✓	P	P						S
PO6	Sporočati vodstvene namene in smeri	SR	P			P		P			P		✓	✓			P					S		
PO7	Upravljalni s človeškimi viri	N	P		P	S	S	P			S		✓				P	P						
PO8	Upravljalni kakovost	SR	P	S		S		P		P	S	P	✓	✓	✓	✓	P	P		S			S	
PO9	Oceniti in upravljalni tveganja	V	P			P		P					✓	✓	✓	✓	S	S	P	P	P	S	S	
PO10	Voditi projekte	V	P	S	S	S	S	S	S	P		S	✓		✓	✓	P	P						
Nabava in uvajanje																								
AI1	Prepoznati avtomatizirane rešitve	SR	P	P	S	S			P						✓	✓	P	S						
AI2	Nabaviti in vzdrževati uporabno programsko opremo	SR	P	P		S			P						✓		P	P		S				S
AI3	Nabaviti in vzdrževati tehnološko infrastrukturo	N			P				P							✓	S	P		S	S			
AI4	Razviti in vzdrževati procedure	N	S	P	S	S			P	S		✓			✓	✓	P	P		S	S	S	S	
AI5	Priskrbeti IT vire	SR		S	P				P			✓	✓	✓	✓	✓	S	P					S	
AI6	Upravljalni spremembe	V		P	S			S	P		S	✓	✓	✓	✓	✓	P	P		P	P			S
AI7	Namestiti rešitve in spremembe	SR	S	P	S	S	S		P	S	S	✓	✓	✓	✓	✓	P	S		S	S			
Dobava in podpora																								
DS1	Opredeliti in upravljalni storitvene nivoje	SR	P	P	P		P	S		P	S	S	✓	✓	✓	✓	P	P	S	S	S	S	S	S
DS2	Upravljalni s storitvi zunanjih izvajalcev	N		P	S	P	S	P	S	P		S	✓	✓	✓	✓	P	P	S	S	S	S	S	S
DS3	Upravljanje z zmogljivostmi in kapacitetami	N	S	S	P	S	S		P		S				✓	✓	P	P			S			
DS4	Zagotoviti neprekinjeno poslovanje	SR	S	P	S	P	S	S		P	S		✓	✓	✓	✓	P	S			P			
DS5	Zagotoviti varnost sistema	V				P			P	S	S	✓	✓	✓	✓	✓			P	P	S	S	S	S
DS6	Prepoznati in razporediti stroške	N		S	P		S		P			✓	✓	✓	✓	✓		P						P
DS7	Izobraževati in usposabljalni uporabnike	N	S	P		S		P			S	✓					P	S						
DS8	Upravljalni s pomočjo uporabnikom in nepredvidenimi dogodki	N	S	P			S	S			P	P	✓		✓		P	P						
DS9	Upravljalni s konfiguracijo	SR		P		S			P				✓	✓	✓	✓	P	S			S			S
DS10	Upravljalni s težavami	SR		P		S			P	S	S	✓	✓	✓	✓	✓	P	P			S			
DS11	Upravljalni s podatki	V		P	P	P			P				✓							P				P
DS12	Upravljalni z naravnim okoljem	N			S	P		S	P						✓					P	P			
DS13	Upravljalni z operacijami	N			P				P	S		✓	✓	✓	✓	✓	P	P		S	S			
Kontrola in ocenjevanje																								
ME1	Kontrola in ocenjevanje kvalitete IT rešitve	V					P				S	P	✓	✓	✓	✓	P	P	S	S	S	S	S	S
ME2	Kontrola in ocenjevanje notranjih kontrol	SR		P		P						P	✓	✓	✓	✓	P	P	S	S	S	S	S	S
ME3	Zagotoviti regulacijsko skladnost	V	P			P			P	S	S	✓	✓	✓	✓	✓							P	S
ME4	Zagotoviti upravljanje IT	V	P	P	P	P	P	P	P		S	P	✓	✓	✓	✓	P	P	S	S	S	S	S	S

N - nizka pomembnost
SR - srednja pomembnost
V - visoka pomembnost

P - primarna relacija
S - sekundarna relacija
Prazno pomeni, da relacije ni

✓ - prosta odločitev organizacije

Vir: Cobit 4.0, str. 175.

¹¹ COSO – mednarodno priznan model notranjega kontroliranja

Po Cobit metodi se izvajajo tudi revizije informacijskih sistemov. Naročnik revizije in revizor informacijskih sistemov skupaj določita cilje revizije. Cilj revizije je lahko celovit pregled uporabe informacijske tehnologije, aplikacij, skladnosti z zakoni, pregled varnosti in zaščit, pregled postopkov za zagotavljanje neprekinjenosti poslovanja, pregled razvoja novih informacijskih sistemov v povezavi s kontrolo kvalitete, pregled usposobljenosti kadrov, ki uporabljajo informacijsko tehnologijo. V posebnih primerih se lahko dogovorita le za podrobni pregled ozkega področja oziroma cilja po metodologiji COBIT, na primer pregled postopkov pri vzdrževanju programske opreme.

Metoda Cobit zagotavlja razvojni model za nadzor celotnega IT procesa, da lahko vodstvo ugotovi, kje je organizacija sedaj, kje stoji v relaciji do najboljših in kje bi želela biti. Ključni kazalci, ki definirajo mero, povedo vodstvu, če IT proces izvaja poslovne zahteve.

Revizija informacijskih sistemov po Cobit metodi lahko vodstvu poslovnih sistemov odgovori na vprašanja, ali uporaba računalniške tehnologije zagotavlja:

- rešitve, ki podpirajo dolgoročne poslovne cilje,
- učinkovitost razvojnih projektov in prenov informacijskih sistemov,
- funkcionalnost računalniških rešitev za izvajanje in upravljanje poslovnih procesov,
- ustreznost izrabe računalniške tehnologije,
- delovanje računalniške podpore brez prekinitev oziroma vzpostavitve ponovnega delovanja brez izgub podatkov v primernem času in
- preprečevanje zlorab, poneverb, kraj in razkritij poslovnih skrivnosti.

Vodstva poslovnih sistemov lahko naročijo revizije informacijskih sistemov kot sestavni del revizije računovodskih izkazov ali kot samostojno revizijo.

Projekte prenove informacijskega sistema lahko opredelimo kot del strategije informacijskega razvoja. Zaradi tega lahko Cobit metodo učinkovito uporabimo na projektih nabave oz. izdelave programske opreme. Sama metoda omogoča v vsakem trenutku ocenitev kakovosti rezultata posamezne aktivnosti, kar pripomore h kakovostnejšemu razvoju programske opreme in hitrejšemu odpravljanju napak in pomanjkljivosti.

2.3 Specifikacija zahtev za programsko opremo¹²

Specifikacija zahtev za programsko opremo (SZPO) je namenjena opisu zahtev programske opreme. Lastnosti dobre SZPO so pravilnost, nedvoumnost, popolnost, skladnost, rangiranost po pomembnosti in/ali stabilnosti, preverljivost, prilagodljivost in sledljivost.

¹² Povzeto po: IEEE. 1998a. in Bregar 2004a.

SZPO je pravilna le, če je vsaka zahteva v njej navedena. Za ugotavljanje pravilnosti ne obstaja nobeno orodje ali postopek. SZPO mora biti primerljiva s katero koli nadrejeno specifikacijo, kot je na primer s specifikacijo sistemskih zahtev, z drugo projektno dokumentacijo in z drugimi uporabnimi standardi. Stranka ali uporabnik lahko določi alternativo, če specifikacija zahtev pravilno prikazuje dejanske potrebe. Sledljivost naredi ta postopek lažji in manj nagnjen k napakam.

SZPO je nedvoumna, če ima vsaka zahteva v dokumentu SZPO samo eno možno interpretacijo. Najmanj, kar je potrebno zagotoviti je, da vsako značilnost končnega produkta opiše z uporabo enega samega izraza. Kadar ima izraz pri uporabi v določenem kontekstu več pomenov (razlag), ga je potrebno vključiti v slovar, kjer se natančno določi njegov pomen.

Popolnost SZPO pomeni, da mora SZPO vključevati vse pomembne zahteve, ki se nanašajo na funkcionalnost, zmogljivost, omejitve pri načrtovanju, attribute in zunanje vmesnike. Definirati je potrebno odziv programske opreme na vse možne razrede vhodnih podatkov in na vse možne situacije. Potrebno je opisati odziv na veljavne in neveljavne vhodne vrednosti. Označiti in navesti je potrebno vse slike, tabele in diagrame ter definirati vse izraze in merske enote.

SZPO je skladna, če zahteve v njej niso protislovne. Najpogostejša nasprotja v SZPO so, da več zahtev opisuje isti objekt realnega sveta, vendar za ta objekt uporabljajo različne izraze, da so specificirane značilnosti v protislovju in da obstaja logično ali časovno protislovje med dvema ali več specificiranimi akcijami.

Vsaka zahteva v SZPO mora biti rangirana po pomembnosti in/ali stabilnosti. Vse zahteve pri programski opremi namreč niso enako pomembne. Nekatere so pomembnejše zlasti življenjsko kritične aplikacije.

SZPO je preverljiva, če je preverljiva vsaka posamezna zahteva. Zahteva pa je preverljiva, če obstaja končen, stroškovno omejen postopek, po katerem lahko človek ali stroj preveri, ali programski izdelek ustreza zahtevi.

SZPO je prilagodljiva, če sta njena zgradba in slog takšna, da lahko katerokoli spremembo zahtev izvedemo zlahka, dosledno in v celoti. SZPO mora biti skladno in uporabno organizirana ter mora vsebovati pregled vsebine, stvarno kazalo in prečne reference. Mora biti neredundantna. To pomeni, da ista zahteva ne nastopa na več kot enem mestu v SZPO. Redundanca sama zase ni napaka, ampak zlahka pripelje do napak. Pomaga lahko izboljšati čitljivost SZPO, težave pa se pojavijo, kadar želimo takšen dokument ažurirati. Predpostavimo, na primer, da je določena zahteva prikazana na dveh mestih. Čez nekaj časa moramo spremeniti takšno zahtevo, vendar spremembo izvedemo le na eni od obeh lokacij. Takrat SZPO postane neskladna. Kadarkoli je redundanca potrebna, moramo v SZPO vključiti eksplicitne prečne reference, ki omogočijo boljšo prilagodljivost.

SZPO je sledljiva, če je znan izvor vsake zahteve in če v bodočem razvoju ali pri dodelavi dokumentacije omogoča sklicivanje na vsako zahtevo. Priporočljiva sta dva tipa sledenja:

- a) Sledenje nazaj (to je k prejšnjim fazam) je omogočeno, če se lahko pri vsaki zahtevi eksplicitno sklicujemo na njen izvor v prejšnjih dokumentih.
- b) Sledenje naprej (to je k vsem dokumentom, izdelanim na podlagi SZPO) je omogočeno, kadar ima vsaka zahteva v SZPO enotno ime ali referenčno številko.

SZPO ima naslednja poglavja:

Vsebina

1. Uvod

1.1 Namen

1.2 Cilj

1.3 Definicije, akronimi, okrajšave

1.4 Reference

1.5 Pregled vsebine

2. Splošen opis

2.1 Perspektive produkta

2.2 Funkcije produkta

2.3 Značilnosti uporabnikov

2.4 Splošne omejitve

2.5 Domneve in odvisnosti

3. Specifične zahteve

Priloge

Stvarno kazalo

S pomočjo tako pripravljenih specifikacij se programerjem natančno opredelijo zahteve in opišejo funkcionalnosti programskega produkta.

2.4 Kakovost in testiranje programske opreme

Zagotavljanje kakovosti je pri razvoju nove programske opreme planiran, sistematičen proces, ki zagotavlja, da je programska oprema in postopki njenega razvoja v skladu s sprejetimi standardi, načinom dela in postopki (Ruthberg 1991, B-2).

Kakovost programske opreme se izraža v njeni zanesljivosti in v zadovoljstvu uporabnikov pri njeni uporabi. Ocenjevanje programske opreme je z vidika kakovosti težko, saj ni kvantitativnih meril s katerimi bi kakovost lahko ocenili. Da bi lahko ocenjevali kakovost, je potrebno predhodno pripraviti merila za kakovost.

Pri tem so lahko v pomoč standardi in metodologije, ki vključujejo tudi merila za kakovost.

Proces testiranja se izvaja z namenom, da bo programska oprema delovala zanesljivo, zadovoljila funkcionalne in tehnične zahteve ter zagotavljala točne rezultate obdelav. Uspešno testiranje vključuje načrtovanje testiranja, izvajanje testiranja ter ocenjevanje rezultatov testiranja. Izbiramo lahko med več metodami testiranja, med katere najpogostejše sodi (Vallabhaneni 1996, 887):

- performančno testiranje glede na predefinirane parametre,
- testiranje funkcionalnosti izdelka (Black-Box),
- testiranje interne strukture in logike sistema (White-Box).

Testiranje programskega paketa je sestavljeno iz naslednjih vrst testiranja (Black 1999, 2-10):

- testiranje posameznih enot,
- testiranje skupine enot oziroma podsistema,
- testiranje izdelka in integracijski testi,
- prevzemno testiranje.

Testiranje programske opreme je neločljivo povezano s pojmom verifikacija in validacija programske opreme. S testiranjem se namreč preverja vse kar verifikacija in validacija pomenita. Preverja se, če posamezne faze programske opreme ustrezajo postavljenim zahtevam, če program pravilno deluje in če je programska oprema skladna z zahtevami.

V nadaljevanju bosta opisana dva standarda, ki lahko služita kot pomoč pri zagotavljanju kakovosti programske opreme in testiranju le te.

2.4.1 ISO 9000¹³

Standardi serije ISO 9000:2000 so namenjeni sistemu vodenja. Ker poslovni svet doživlja neprestane in hitre spremembe, podjetja iščejo najrazličnejše možnosti za preživetje na trgu. Vodilna podjetja v svetu svoj uspeh povezujejo z raznimi koncepti vodenja, vse večje število je tistih, ki kot temelj svoje poti v vodenje celovite kakovosti izbirajo model vodenja kakovosti, opisan v standardih serije ISO 9000.

¹³ Povzeto po ISO 9001:2000

2.4.1.1 Standardi skupine ISO 9000:2000

Standardi te skupine so bili sprejeti v letu 2000. Sestavljajo jo naslednji standardi:

- 1
- ISO 9000: Sistemi vodenja kakovosti – Načela in slovar
- ISO 9001: Sistemi vodenja kakovosti – Zahteve
- ISO 9004: Sistemi vodenja kakovosti – Smernice
- ISO 19011: Smernice za presojanje sistemov kakovosti

Standardi ISO 9000:2000 se zavzemajo za privzem procesnega pristopa pri razvijanju, uvajanju in izboljševanju sistema vodenja kakovosti. Procesni pristop se zrcali v strukturi standarda ISO 9004:2000, *Sistem vodenja kakovosti – Smernice za izboljševanje delovanja* in tudi v ISO 9001:2000, *Sistem vodenja kakovosti – Zahteve*. Struktura „20 elementov“ standarda ISO 9001:1994 je bila zamenjana s tem sistemom vodenja kakovosti, ki temelji na procesih in je shematično prikazana na spodnji sliki.

SLIKA 6: MODEL SISTEMA VODENJA KAKOVOSTI, KI TEMELJI NA PROCESU (VZET IZ ISO 9000:2000)



Vir: SIST 2006

V točki 0.2 v uvodu standarda ISO 9001:2000 je glede procesnega pristopa navedeno:

„Pri uporabi znotraj sistema vodenja kakovosti tak pristop poudari pomen:

1. razumevanja in izpolnjevanja zahtev,
2. potrebe po obravnavanju procesov z vidika dodane vrednosti,
3. pridobivanje rezultatov delovanja in učinkovitosti procesov in
4. nenehnega izboljševanja procesov na podlagi objektivnega merjenja.”

Nadaljnje smernice so zapisane v točki 2.3 standarda ISO 9000:2000.

Znotraj konteksta standarda ISO 9001:2000 procesni pristop vključuje procese, potrebne za realizacijo proizvoda in druge procese, potrebne za učinkovito uvajanje sistema vodenja kakovosti, kot so: proces notranje presoje, proces vodstvenega pregleda, proces analize podatkov ter proces vodenja virov.

Zahteve za te procese so navedene v naslednjih točkah standarda ISO 9001:2000:

- 4 Sistem vodenja kakovosti
- 5 Odgovornost vodstva
- 6 Vodenje virov
- 7 Realizacija proizvoda
- 8 Merjenje, analize in izboljševanje

Splošne zahteve za sistem vodenja kakovosti so opredeljene v točki 4.1 standarda ISO 9001:2000. V nadaljevanju so navedeni nekateri napotki glede vprašanj, ki naj si jih organizacija zastavi pri obravnavanju teh zahtev. Poudarjeno je, da so ta vprašanja le primeri in naj se ne bi interpretirala kot edini način pri izpolnjevanju zahtev.

a) Identificirati procese, potrebne za sistem vodenja kakovosti in njihovo uporabo v celotni organizaciji:

- Kateri so potrebni procesi za sistem vodenja kakovosti?
- Kdo so odjemalci za posamezen proces (notranji ali zunanji odjemalci)?
- Kakšne so zahteve teh odjemalcev?
- Kdo je „lastnik“ procesa?
- Ali katerega od teh procesov izvajajo zunanji izvajalci?
- Kakšni so vhodi in izhodi za vsak proces?

b) Določiti zaporedje in medsebojne vplive teh procesov:

- Kakšen je celotni potek procesov?
- Kako lahko to opišemo (zemljevidi procesov ali diagrami poteka)?
- Kakšne so povezave med procesi?
- Kakšno dokumentacijo potrebujemo?

c) **Določiti kriterije in metode, potrebne za zagotovitev učinkovitega delovanja in tudi učinkovitega obvladovanja teh procesov:**

- Kakšne so karakteristike namernih in nenamernih rezultatov procesa?
- Kakšni so kriteriji za spremljanje, merjenje in analize?
- Kako lahko to vključimo v načrtovanje sistema vodenja kakovosti in procese – realizacije proizvoda?
- Kakšne so gospodarske postavke (stroški, čas, odpad itd.)?
- Katere metode so primerne za zbiranje podatkov?

d) **Zagotoviti, da so na voljo viri in informacije, potrebni za podporo pri delovanju in nadzorovanju teh procesov:**

- Kateri viri so potrebni za vsak proces?
- Kakšni so kanali komuniciranja?
- Kako lahko zagotovimo zunanje in notranje informacije o procesu?
- Kako dobimo povratne informacije?
- Katere podatke moramo zbirati?
- Katere zapise moramo shranjevati?

e) **Meriti, nadzorovati in analizirati te procese:**

- Kako lahko spremljamo delovanje procesa (zmožnost procesa, zadovoljstvo odjemalcev)?
- Katera merjenja so potrebna?
- Kako lahko najbolje analiziramo zbrane informacije (statistične tehnike)?
- Kaj nam rezultati analiz povedo?

f) **Izvajati ukrepe, potrebne za doseganje planiranih rezultatov in nenehnega izboljševanja teh procesov:**

- Kako lahko izboljšamo proces?
- Kateri korektivni in/ali preventivni ukrepi so potrebni?
- So bili ti korektivni/preventivni ukrepi izvedeni?
- So učinkoviti?

Omejitve tega modela so, da se ne nanaša samo na softver, ne vključuje tehnološkega razvoja, dejanska sposobnost procesa je odvisna od njegove zrelosti. Skladnost z ISO 9001 ne zagotavlja, da v procesu niso uzakonjene napake.

Standard ISO 9001:2000 je plod dolgoletnih izkušenj in spoznaj o nujnih dejavnostih za doseganje ustreznega poslovanja razvojno usmerjenih podjetij.

Sam standard se ne more direktno uporabljati za razvoj programske opreme. Temu so namenjene smernice, ki so opisane v ISO/IEC 90003. ISO/IEC 90003 nič

ne spreminja in ne dodaja standardu 9001:2000, temveč samo podaja smernice, kako ta standarda uporabiti pri programski opremi.

2.4.1.2 Karakteristike kakovosti po ISO 9126

Mednarodni standard ISO 9126 opisuje 6 karakteristik kakovosti programskega proizvoda. Te se po določilih standarda minimalno prekrivajo in so osnova za nadaljnjo razdelitev na podkarakteristike. Karakteristike in možne podkarakteristike so v standardu določene le opisno (Pivka 1996, 31-34).

Funkcionalnost

Funkcionalnost proizvoda (storitev) opredeljuje množica tistih karakteristik, ki vplivajo na posamezne funkcije in njihove lastnosti. Funkcionalnost opredeljuje kaj proizvod (storitev) dela, da zadovolji potrebe uporabnika.

Zanesljivost

Zanesljivost je množica karakteristik, ki se nanašajo na zmožnost programskega proizvoda, da je v danih pogojih v določenem času uporaben.

Uporabnost

Uporabnost je tista karakteristika, ki opredeljuje množico podkarakteristik, ki se nanašajo na oceno uporabnika, ali je programski proizvod zanj uporaben ali ne.

Učinkovitost

Učinkovitost sestavlja množica atributov, ki opredeljuje odnos med funkcijami, ki jih je programska oprema sposobna izvajati pod določenimi pogoji in za to potrebno količino virov.

Vzdrževalnost

Vzdrževalnost je množica tistih lastnosti programske opreme, ki se nanaša na njeno sposobnost, da lahko v programski opremi izvajamo korekture, spremembe in dopolnitve. Ta lastnost kaže, kako in na kakšen način se je programska oprema sposobna prilagajati uporabnikovim zahtevam za spremembe.

Prenosljivost

Prenosljivost je lastnost, ki opredeljuje zmožnost prenašanja programske opreme v različna organizacijska okolja, strojna okolja in okolja različnih operacijskih sistemov.

2.4.2 Testiranje programske opreme po IEEE standardu

Proces testiranja se izvaja z namenom, da bo programska oprema delovala zanesljivo, zadovoljila funkcionalne in tehnične zahteve ter zagotavljala točne rezultate obdelav. Uspešno testiranje vključuje načrtovanje testiranja, izvajanje testiranja ter ocenjevanje rezultatov testiranja. Izbiramo lahko med več metodami testiranja, med katere najpogosteje sodi (Vallabhaneni 1996, 887):

- performančno testiranje glede na predefinirane parametre,
- testiranje funkcionalnosti izdelka (Black-Box),
- testiranje interne strukture in logike sistema (White-Box),

Testiranje programskega paketa je sestavljeno iz naslednjih vrst testiranja (Black 1999, 2-10):

- testiranje posameznih enot,
- testiranje skupine enot oziroma podsistema,
- testiranje izdelka in integracijski testi in
- prevzemno testiranje.

Zagotavljanje kakovosti pri projektih razvoja novih informacijskih sistemov je planiran, sistematičen proces, ki zagotavlja, da so informacijski sistemi in procesi njihovega razvoja v skladu s sprejetimi standardi, načinom dela in postopki (Ruthberg 1991, B-2). Na zunaj se izraža kakovost projekta v zanesljivosti novega informacijskega sistema in v zadovoljstvu uporabnikov pri njegovi uporabi. Da bi lahko ocenjevali kakovost, je potrebno predhodno pripraviti merila za kakovost. Te obsegajo (Warren in drugi 1998, B4-28):

- formulacijo sistemskih in programskih standardov,
- pregled dokumentacije načrtovanja informacijskih sistemov z namenom, da se ugotovi njihova skladnost s prej sprejetimi standardi in za zagotovitev, da bo imel novi sistem tudi nadzorne funkcije,
- pregled testiranja posameznih programskih komponent in testiranja celotnega sistema, nadzor nad paralelnim ali poskusnim izvajanjem novega informacijskega sistema z namenom zagotovitve skladnosti s standardi,
- potrditev, da se pri kodiranju in izdelavi sistema uporabljajo standardi,
- zagotovitev, da novi sistem vsebuje ustrezne kontrole, postopke varnosti in zaščite ter revizijske sledi.

Nadzor kakovosti projekta lahko izvaja notranji revizor, zunanji revizor, neodvisna skupina za zagotavljanje kakovosti ali katerakoli kombinacija od naštetih kontrol.

IEEE standard¹⁴**1. Namen testirnega načrta**

S testirnim načrtom se opredeli obseg in razpored testirnih aktivnosti ter pristop k njihovem izvajanju. Opredelijo se vsi potrebni viri. Identificirajo se predmeti testiranja (izvorna in objektna koda, nadzorni podatki) ter njihove značilnosti (učinkovitost, prenosljivost, funkcionalnost), ki jih je potrebno preveriti. Določijo se naloge, opravljene v sklopu testiranja, osebe, ki so zanje odgovorne in tveganja, povezana z načrtom testiranja.

2. Osnutek načrta

Načrt naj ima naslednjo strukturo:

- a) Identifikator testirnega načrta;
- b) Uvod;
- c) Opis testiranega produkta;
- d) Značilnosti, ki bodo testirane;
- e) Značilnosti, ki ne bodo testirane;
- f) Pristop testiranja;
- g) Odobritvena/odpovedna kriterijska funkcija;
- h) Prekinitev in nadaljevanje testiranja;
- i) Testiranje dostavljive dokumentacije;
- j) Potrebne aktivnosti za izvedbo testiranja;
- k) Opis potrebnega testirnega okolja;
- l) Identifikacija odgovornosti;
- m) Človeški viri in potrebno izobraževanje;
- n) Urnik;
- o) Tveganja in posledice;
- p) Potrditev testirnega načrta;

Poglavja naj bodo razvrščena v podanem zaporedju. Dodatna poglavja smejo biti vključena pred točko p („potrditev testirnega načrta“). Če je del ali celotna vsebina poglavja navedena v poljubnem drugem dokumentu, mora biti podana referenca na ta dokument.

a) Identifikator testirnega načrta

Specificiran je enoličen identifikator, prirejen načrtu testiranja.

b) Uvod

Podpoglavje povzema predmete/postavke in značilnosti programskega produkta, ki jih bomo testirali. V testirnim načrtu najvišjega nivoja so podane reference na zunanje dokumente, kot so: projektni načrt, načrt zagotavljanja kakovosti, načrt upravljanja konfiguracije, standardi...

¹⁴ Povzeto po: IEEE 1998b in Bregar 2004b.

Pri večnivojskih načrtih naj vsak načrt nižjega nivoja vsebuje referenco na načrt neposredno višjega nivoja.

c) Opis testiranega produkta

Identificirati je potrebno predmete/postavke, ki bodo podvržene testiranju. V opisu se je potrebno sklicevati na naslednjo dokumentacijo:

- Specifikacijo zahtev;
- Specifikacijo načrtov programske opreme;
- Uporabniški priročnik;
- Priročnik delovanja;
- Priročnik namestitve.

Na kratko naj bodo opisane komponente programskega sistema, ki ga razvijamo in testiramo. Predstavljeni naj bodo vmesniki med morebitnimi podsistemi. Prav tako mora biti pojasnjena narava sistema.

d) Značilnosti, ki bodo testirane

Z namenom testiranja se identificirajo značilnosti programske opreme in/ali kombinacije teh značilnosti. Za vsako mora biti specificiran ločen testirni plan.

e) Značilnosti, ki ne bodo testirane

Identificirajo se značilnosti programske opreme in/ali kombinacije teh značilnosti, ki ne bodo testirane. Obrazložijo se razlogi, zakaj bo testiranje izpuščeno.

f) Pristop testiranja (Bregar 2004b, 3)

„Poglavje služi opredelitvi pristopa k testiranju. Opis mora biti dovolj podroben, da bo omogočil identifikacijo vseh bistvenih testirnih nalog kakor tudi ocenitev potrebnega časa za izvršitev le-teh. Identificirani naj bodo tipi testiranja. Pri tem naj bodo za vsak tip definirane ustrezne metode in postopki ter opredeljeni zahtevani vhodi in izhodi. V povezavi z vhodi je nujno specificirati njihove izvore, medtem ko mora biti posameznemu izhodu pripisan namen in format. Prav tako morajo biti definirani kriteriji za vrednoteneje dobljenih rezultatov testov ter kriteriji za zaključitev aktivnosti testiranja (na primer: stopnja pogostosti pojavljanja napak). Določena mora biti tudi najmanjša še sprejemljiva stopnja izčrpnosti testiranja (na primer: kolikšen odstotek kontrolnih poti je potrebno zajeti pri testiranju po metodi bele skrinje¹⁵).

Uporabljeni pristopi naj bodo prirojeni posameznim skupinam/kombinacijam značilnosti. Za posamezen nivo testiranja mora obstajati testirni načrt kot tudi množica izročljivih produktov. Prav tako je priporočljivo na različnih nivojih poseči po različnih tehnikah testiranja, kakršni sta od spodaj navzgor (bottom-up) ali od zgoraj navzdol (top-down). Odgovoriti moramo na številna vprašanja. Kakšne

¹⁵ Pri metodi bele skrinje se testirajo interne strukture in logika sistema.

testne gonilnike (test drivers) ali nadomestila (test stubs) bomo sprogramirali? Bomo posegli po metodi črne¹⁶ ali bele skrinje? Kako bomo analizirali kontrolne tokove, če se bomo odločili za belo skrinjo? Kakšno particijo ekvivalenčnih razredov bomo izbrali v primeru črne skrinje?”

g) Odobritvena/odpovedna kriterijska funkcija

Specificirajo se kriteriji, na podlagi katerih se preveri ali so posamezni „predmeti” produkta (software items) opravili test. Analogno se specificirajo kriteriji za ugotavljanje ustreznosti značilnosti produkta (software features).

h) Prekinitev in nadaljevanje testiranja

Podajo se kriteriji za prekinitev vseh ali zgolj nekaterih aktivnosti testiranja ter specificirajo aktivnosti, ki morajo biti ponovljene, ko se testiranje nadaljuje.

i) Testiranje dostavljive dokumentacije

Identificirana mora biti vsa dostavljiva (možna in dostavljena) dokumentacija. Vključeni naj bodo naslednji dokumenti:

- (a) Načrt testiranja;
- (b) Specifikacije metod testiranja;
- (c) Specifikacije testnih primerkov;
- (d) Specifikacije postopka testiranja;
- (e) Predstavitev testiranih „predmetov” in značilnosti programskega produkta;
- (f) Dnevnik testiranja (test log);
- (g) Poročila o napakah;
- (h) Povzetek testiranja.

Kot dostavljiva dokumentacija so lahko identificirani tudi vhodni in izhodni podatki ter testirna orodja (na primer: gonilniki in nadomestila modulov).

j) Potrebne aktivnosti za izvedbo testiranja

Identificirati je potrebno nabor nalog in aktivnosti za pripravo in izvedbo testiranja ter določiti vse medsebojne odvisnosti. Določiti je potrebno tudi znanja, ki jih testerji potrebujejo.

k) Opis potrebnega testirnega okolja

Specificirane morajo biti potrebne ter zaželenne značilnosti testirnega okolja. Specifikacija naj obravnava fizične lastnosti strojne opreme in sistemsko-komunikacijske programske opreme, način uporabe (na primer: „stand-alone”) in kakršnokoli drugo programsko opremo, potrebno za izvajanje testiranja. Prav tako

¹⁶ Pri metodi črne skrinje se testirajo funkcionalnosti programske opreme.

naj določa zahtevani nivo varnosti za testirne pripomočke, sistemsko programsko opremo in drugo lastniško opremo. Identificiran naj bo izvor orodij, ki so potrebna za testiranje, vendar niso na voljo.

l) Identifikacija odgovornosti

Določiti je potrebno skupine, ki bodo odgovorne za upravljanje, načrtovanje, pripravljanje, izvajanje, preverjanje in izboljševanje postopka testiranja. Prav tako je potrebno določiti skupine, ki bodo odgovorne za izbiro testiranih „predmetov“ in značilnosti programskega produkta.

m) Človeški viri in potrebno izobraževanje

Potrebe po testirnem osebju naj bodo specificirane glede na nivo zahtevanega znanja. Podane naj bodo možnosti izobraževanja – pridobivanja potrebnega znanja.

n) Urnik

Upoštevati je potrebno mejnike, postavljene v urniku celotnega procesa razvoja programskega sistema. Definirati je potrebno dodatne mejnike, vezane na fazo testiranja. Oceniti je potrebno čas, ki je potreben za izvajanje vsake posamezne naloge testiranja, oblikovati je potrebno časovni raspored vseh aktivnosti in določiti končni rok testiranja. Posameznemu viru (orodja, osebje...) je potrebno določiti časovni interval aktivnosti.

o) Tveganja in posledice

Identificirati je potrebno tveganja, ki izvirajo iz načrta testiranja. Za vsako ugotovljeno tveganje je potrebno navesti možne posledice (na primer: zamik roka dobave produkta).

p) Potrditev testirnega načrta

Določiti je potrebno osebe, ki bodo načrt testiranja potrdile. Ravno tako je potrebno določiti podpisnike, roke...

2.5 Metodologija vodenja projektov v državni upravi

Po definiciji (Hauc 1995, 3) je projekt zaključen proces izvajanja določenih del – aktivnosti, ki so med seboj logično povezane za doseganje ciljev projekta, z nadaljnjim povezovanjem aktivnosti pa se postopoma uresničita objektni in namenski končni cilj.

Projekt je (Hauc 1995, 2):

- ciljno usmerjen časovno omejen proces,
- proces doseganja ciljev,
- proces izvajanja aktivnosti za doseganje ciljev,
- proces izvajanja logično med seboj povezanih aktivnosti, kar predstavlja tehnologijo izvedbe projekta,
- proces ustvarjanja,
- proces med tistim, kar želimo ustvariti in tistim, kar bo ustvarjeno,
- proces angažiranja naročnika projekta in izvajalcev v omejenem času,
- proces časovno omejenega finančnega vlaganja
- proces integracije znanj in izkušenj za izvedbo njegove namere,
- proces udejanjanja strategij,
- proces, ki ne sovпада s kontinuiranim poslovanjem in proizvodnjo, ju pa na razne načine dopolnjuje.

Po Kerznerju (1992) je klasična specifikacija projektnega managementa sestavljena iz:

- planiranja,
- organiziranja,
- upravljanja,
- kontroliranja in
- ukazovanja.

Za vodenje projektov v državni upravi se uporablja metodologija¹⁷, ki so jo pripravili na Centru vlade Republike Slovenije za informatiko (CVI). Predstavitev metodologije se nahaja na spletni strani CVI (CVI 2006) in je dosegljiva samo uporabnikom znotraj omrežja CVI. V nadaljevanju je predstavljena ta metodologija:

2.5.1 Uvod

Metodologija vodenja projektov v državni upravi je nastala na podlagi originalnih priročnikov o vodenju projektov po metodologiji **PRINCE** (Project **IN** Controlled Environment). Metodologijo PRINCE je na zahtevo angleške vlade razvila agencija **CCTA** (Central Computer and Telecommunication Agency) za potrebe nadzora porabe denarja, kakovosti izdelkov projekta in pravočasnosti izdelanih rešitev. Prva verzija metodologije je bila razvita za projekte s področja informacijske tehnologije (IT). Kasneje se je metodologija, zaradi svoje prilagodljivosti, uveljavila na vseh področjih projektnega dela. Leta 2005 je Office of Government Commerce (OGC), ki je zamenjala CCTA, izdala revizijo priročnika PRINCE 2.00.

¹⁷ Metodologija vodenja projektov v državni upravi: projekti informacijske tehnologije: priročnik. 2001. Ljubljana: Center vlade RS za informatiko.

Metodologija se, kot ena izmed treh standardnih metodologij vodenja projektov, uporablja tudi v državah evropske skupnosti in je v skladu s standardi ISO 9000.

2.5.2 *Namen*

Z uvedbo predlagane metodologije bo presežena dosedanja praksa, ko se naloge, kjer bi bil primeren projektni pristop, lansirajo premalo organizirano, z vnaprej nejasnimi parametri.

Prav tako bodo preseženi glavni razlogi za neuspeh nekaterih projektov:

- pomanjkanje koordinacije pri planiranju in uporabi virov,
- pomanjkanje koordinacije pri planiranju in izvajanju aktivnosti,
- pomanjkanje komunikacije med udeleženci na projektu,
- nepravilno planiranje časa, finančnih in ostalih virov,
- pomanjkanje nadzora nad napredkom projekta,
- pomanjkanje nadzora nad kakovostjo izdelkov projekta in
- pomanjkanje podpore vodstva projektom.

Vsi naštet razlogi vodijo v to, da so izdelki projekta neprimerni za uporabnika; da projekti trajajo dalj, kot bi lahko in da stanejo več denarja, kot bi bilo potrebno.

Na srečo je danes že prodrlo spoznanje, da je edini primerni način izvajanja velikih nalog projektni pristop in z njim uporaba metodologije, ki zagotavlja planiranje, spremljanje in nadziranje vseh potrebnih izdelkov, aktivnosti in virov ter primerjavo s cilji projekta in merili kakovosti.

Predlagana metodologija obsega postopke in dokumente, ki pokrivajo vse faze življenjskega cikla projekta: pred začetkom izvajanja projekta, tekom izvajanja projekta in tudi po končanju projekta. Vse delo poteka v organiziranem okolju, ki ga sestavljajo: jasna organizacija z natančno definiranimi nalogami in odgovornostmi za vse vloge ter natančno definirani postopki.

Metodologija je primerna tudi pri projektih, kjer sodelujejo zunanji izvajalci, saj vnaša enotna pravila sodelovanja med naročnikom (uporabnikom) in izvajalci. Tako harmonizira odnose med udeleženci na projektu, rezultate in tudi okolje projekta.

2.5.3 *Struktura gradiva Metodologija vodenja projektov v državni upravi*

Osnovni priročnik opisuje elemente metodologije, ki so primerni za izvajanje vseh projektov ne glede na področje in vrsto, v aneksu pa so dodani elementi, ki so specifični za področje informacijske tehnologije.

Struktura obeh gradiv je zaradi komplementarnosti enaka in je naslednja:

1. Uvod v metodologijo
2. Organiziranost projekta
3. Planiranje projekta
4. Nadziranje projekta in poročanje
5. Postopki vodenja projekta
6. Projektna dokumentacija
7. Nadziranje kakovosti
8. Spremljanje razvoja izdelkov
9. Obvladovanje tveganj
10. Izdelki
11. Obrazci
12. Slovar
13. Indeks
14. Literatura

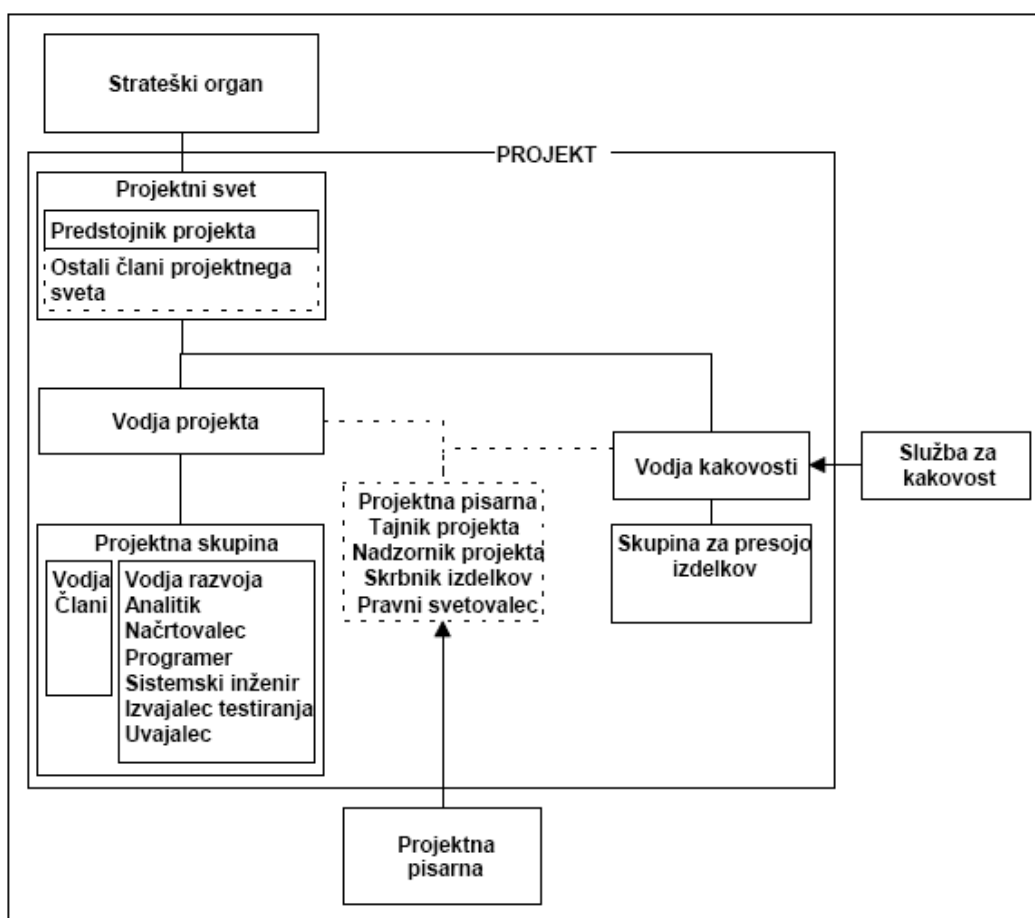
2.5.4 Predstavitev metodologije

Metodologija predvideva planiranje in vodenje izvajanja projektov z definiranimi aktivnostmi in rezultati, ki ob planiranih ciljih, virih in merilih kakovosti, poteka v nadzorovanem okolju. Temelj nadzorovanega okolja je jasno določena organizacijska struktura, kjer so naštetih nosilci vlog, njihove naloge in odgovornosti ter pravila poročanja in nadziranja.

Metodologija je razdeljena v več med seboj prepletenih sklopov:

- 1. Organiziranost projekta:** Metodologija predlaga organizacijsko strukturo, v okviru katere lahko iščemo možnosti za organizacijsko strukturo konkretnega projekta. Nekatere vloge v organizacijski strukturi morajo biti obvezno zasedene (projektni svet, vodja projekta), druge pa so aktualne samo pri kompleksnejših projektih oziroma pri projektih, kjer so poudarjeni določeni elementi (npr. vodja faze, analitik). Za vsako od vlog na projektu metodologija opisuje osnovne odgovornosti in naloge.

SLIKA 7: POLNA ORGANIZACIJSKA STRUKTURA PROJEKTA



Vir: CVI (2001, 35)

- 2. Planiranje projekta:** Metodologija opisuje postopek in tehnike planiranja, ki temeljijo na izdelkih projekta. Rezultat planiranja, ki se izvaja skozi celoten življenjski cikel projekta so plani na različnih nivojih. Plan projekta in plane posameznih faz izdelamo na projektu vedno, plane na nižjih nivojih pa samo po potrebi.
- 3. Nadziranje projekta in poročanje:** Metodologija opredeljuje mehanizme nadzora in poročanja na projektu. Opisane so različne vrste nadzornih točk, postopki nadziranja, oblika in vsebina poročil ter odgovornosti nosilcev vlog v organizacijski strukturi pri nadziranju in poročanju .
- 4. Postopki vodenja projekta:** metodologija natančno določa aktivnosti in zaporedje, od začetka do zaključka vodenja projekta. Opredeljuje tudi naloge in odgovornosti, izdelke in obrazce znotraj teh aktivnosti.
- 5. Vodenje projektne dokumentacije:** Ustrezno vodenje projektne dokumentacije je predpogoj za uspešno delo na projektu in za varnost podatkov o projektu. V metodologiji so specificirani postopki za: izdelavo, posredovanje, shranjevanje in arhiviranje dokumentov, ki nastajajo na projektu. Opredeljene so tudi projektne mape in njihova struktura ter njihov skrbnik.

- 6. Nadziranje kakovosti izdelkov projekta:** Pomembno je, da izdelki projekta ustrezajo merilom kakovosti, ki so bili postavljeni ob vzpostavitvi projekta. Metodologija priporoča postopke, ki omogočajo ustrezno preverjanje kakovosti izdelkov, poleg tega pa predvideva tudi posebne vloge na projektu, ki so zadolžene za kakovost.
- 7. Spremljanje razvoja projekta:** Pri projektih z veliko izdelki predstavlja pomemben del vodenja tudi spremljanje razvoja izdelkov. Spremljanje razvoja izdelkov obsega evidentiranje izdelkov, nadzor nad njimi ter spremljanje statusa in verzije. Metodologija predlaga postopke, ki se izvajajo v teku projekta in tudi po zaključku projekta.
- 8. Obvladovanje tveganja na projektu:** Evidentiranje dejavnikov tveganja in ocena skupnega tveganja projekta imata pomemben vpliv na izvedbo projekta. Pomemben metodološki pripomoček pri oceni tveganja je kontrolni seznam tveganj, ki je lahko skupen za določeno vrsto projektov ali pa specifičen za projekt.
- 9. Izdelki projekta:** Metodologija pozna tri vrste izdelkov:
- **izdelki vodenja:** Izdelki vodenja nastajajo pri planiranju in nadzoru projekta. Med izdelki vodenja je še posebej pomemben Vzpostavitevni dokument projekta, ki združuje vse elemente za uspešen začetek projekta (organizacijska struktura, plani, pravila nadzora in poročanja),
 - **izdelki kakovosti:** Izdelki kakovosti nastanejo pri ocenjevanju kakovosti izdelkov projekta,
 - **tehnični izdelki:** Tehnični izdelki so izdelki, zaradi katerih se projekt izvaja. Za vsako posamezno področje, vrsto projekta ali celo posamezen projekt jih je potrebno posebej opredeliti.

2.5.5 Prednosti in pričakovane koristi uporabe metodologije

Kot je bilo že omenjeno v poglavju o namenu metodologije, uporaba predstavljene metodologije zagotavlja ustrezno planiranje, spremljanje in nadziranje izdelkov, aktivnosti in virov na projektu.

Glavne prednosti uporabe metodologije so naslednje:

- usmerjena je v izdelke projekta,
- zagotavlja vključevanje uporabnika, naročnika in izvajalca,
- zahteva pripravo na projekt - faza vzpostavitve projekta,
- zahteva plan zagotavljanja kakovosti izdelkov projekta,
- zahteva plan izvajanja projekta,
- zahteva natančno opredeljene naloge in odgovornosti vlog na projektu,
- predpisuje enotni način vodenja projektne dokumentacije,
- je splošno uporabna in prilagodljiva za vse velikosti in vrste projektov,
- predvideva nadzor nad dejavniki tveganja in
- predvideva spremljanje razvoja izdelkov.

Pričakovane koristi uporabe metodologije se bodo pokazale kmalu po pričetku njenega uvajanja v organe državne uprave. Nekatere koristi so implicitno navedene že v prejšnjih poglavjih, zato bodo v nadaljevanju podane samo najvažnejše:

- povečanje učinkovitosti projektov v celoti (sprejem te filozofije pri vsem delu),
- poslovna utemeljitev projektov (harmonizacija ciljev, vložkov in rezultatov) in
- stalen nadzor nad viri (kadri, sredstva, čas) skozi celotni življenjski cikel projekta.

S poenotenjem vseh elementov projekta je dosežena preglednost vseh projektov v nekem projektnem okolju, kar omogoča neposredne primerjave in odpravo šibkih mest.

Skozi celotno trajanje življenjskega cikla projekta je zagotovljen ustrezen nadzor nad uporabo virov, kar omogoča njihovo enakomerno in optimalno izrabo. Metodologija predvideva delitev odgovornosti za uspeh projekta med uporabnika (naročnika) in izvajalca. Tako vsi vpleteni skrbijo za koristi projekta in ne samo ena ali druga stran.

Poenotenje postopkov in dokumentov v nekem okolju v veliki meri zmanjša možnost nesporazumov in s tem odpravi nepotrebne stroške, ki pri tem nastanejo.

Nad dokumentacijo projekta se izvajajo enotni postopki upravljanja z dokumenti (kreiranje, ocenjevanje, shranjevanje, posredovanje). Dokumentacija se shranjuje na enem mestu, omogočena njena uporaba po zaključku projekta, kreira pa se tudi zbirka znanja, ki ostane tudi potem, ko nek strokovnjak zapusti to okolje.

Obseg administrativnih del, je prilagodljiv velikosti in vrsti projekta. Sčasoma, ko uporaba metodologije postane vsakodnevna praksa, se lahko posameznik na projektu posveti nemotenemu opravljanju strokovnih nalog in ne izgublja časa z odvečno administracijo.

Zaposleni, ki se bodo srečevali z uporabo metodologije, bodo brez težav lahko prehajali iz enega v drugo projektno okolje znotraj državne uprave. S tem pa bo dosežen prihranek pri času in finančnih sredstvih, ki bi se pojavilo ob ponovnem učenju specifičnih metod in tehnik, ki se uporabljajo v posameznih okoljih.

Sodelovanje pri mednarodnih projektih, na mednarodnih razpisih in podobno, bo veliko lažje saj je metodologija v Evropi dobro poznana in priznana. Izhajajoč iz izkušenj pa se tujci mnogo lažje odločajo za sodelovanje na projektih, če le to poteka na urejen način.

2.6 Sklep

Današnji računalniški sistemi so zapleteni in težko razumljivi v popolnosti. Zaradi tega se sistemi modelirajo, s čimer poskušamo na poenostavljen in razumljiv način prikazati kompleksno realnost. Izbira modelov ima velik vpliv na to, kako pristopimo k reševanju problema in kako oblikujemo rešitev. En sam model nikoli ni dovolj. Najboljši pristop za modeliranje zapletenih sistemov je uporaba nekaj različnih modelov.

Metode naročanja oziroma izdelave programske opreme so namenjene standardizaciji postopkov naročila, izdelave, nadzorovanja in izboljševanja le te. Ne glede na to, kateri model uporabimo, je pomembno, da se ga upošteva. Le v tem primeru je verjetnost uspeha maksimalna. Razmišljanje o tem, kateri model je boljši oz. najboljši je brezpredmetno, kljub temu, da nekateri avtorji v ospredje postavljajo Cobit, ki naj bi združeval najboljše iz vseh ostalih modelov. V različnih situacijah pridejo v poštev različni modeli, precej je odvisno tudi od tega, kakšno organizacijo ima podjetje. Če na primer ima podjetje v celoti vgrajen model CMMI, potem bi bilo nesmiselno in neracionalno, da bi za primer izgradnje IS uporabilo model Cobit. Modeli so bili narejeni z namenom olajšati in standardizirati postopke in tako jih je tudi potrebno uporabljati.

3 METODA IZDELAVE ALI NAROČANJA PROGRAMSKE OPREME

3.1 Vzpostavitev projektne organizacije

Osnova za vzpostavitev projekta je problem, ki ga imajo uporabniki. Po Haucu (2002, 234) se lahko zagon projekta izvede s strateško konferenco, s projektnim zagonskim sestankom ali z izstavitvijo delovnega naloga projekta. Vodstvo Zavoda s sklepom (delovni nalog) ustanovi projektno in usmerjevalno skupino ter opredeliti projektne zahteve (cilje, stroške, projektno organizacijo, dejavnike tveganja, način nadzora...). Vodja projekta mora v skladu z usmeritvami pripraviti terminski plan v katerem morajo biti prikazane aktivnosti projekta in njihovo trajanje za izpolnitev zahtev in doseg končnega cilja. Terminski plan mora potrditi usmerjevalna skupina.

Osnovne aktivnosti projekta so:

- analiza stanja,
- izdelava procesnega modela,
- izdelava ekranskih vmesnikov,
- izdelava podatkovnega modela,
- programiranje,
- testiranje,
- zagotavljanje kakovosti in
- izobraževanje.

3.2 Analiza stanja

Temeljito je potrebno analizirati obstoječe stanje. Možni sta dve varianti:

1. Aplikacija že obstaja, vendar je neustrezna.
2. Aplikacija še ne obstaja.

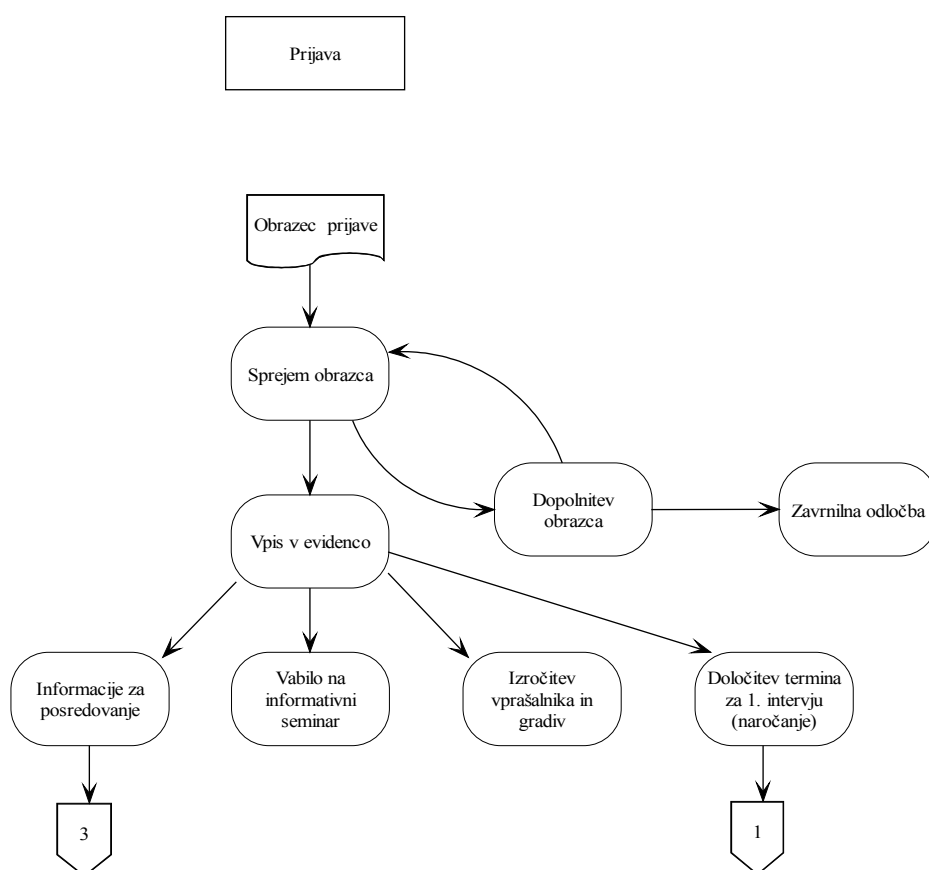
Če aplikacija že obstaja in je neustrezna, je potrebno najprej popisati vse funkcije, ki jih aplikacija ima. Narediti je potrebno diagram, ki grafično prikazuje postopek dela aplikacije. Vse rešitve, ki so v obstoječi aplikaciji neustrezne je potrebno odstraniti, da nas pri nadaljnjem delu ne bodo ovirale in obremenjevale. Tu je zelo pomembno, da to fazo vodi nekdo, ki zna uporabnike ustrezno voditi. Ni dovolj, če uporabnike vprašamo, ali določeno funkcijo potrebujejo ali ne. Če to naredimo, bomo skoraj vedno dobili odgovor, da funkcijo potrebujejo. Uporabnikom je potrebno za vsako funkcijo postaviti naslednja vprašanja:

1. Zakaj se ta funkcija potrebuje?
2. Kako pogosto se uporablja?

3. Kaj bi se zgodilo, če bi to funkcijo odstranili?
4. Česa funkcija ne naredi tako, kot bi bilo najboljše, najhitreje, bolj enostavno?
5. Kakšna bi bila boljša rešitev?

Obvezno je potrebno postavljati odprta vprašanja, na katera uporabnik ne more odgovoriti na kratko. Če opazimo, da so uporabniki pri teh vprašanjih nejasni, zmedeni, nimajo kaj povedati, delujejo nervozno, to pomeni, da s to funkcijo nekaj ni v redu in jo je potrebno odstraniti. S tem tudi odstranimo napake, ki so bile v njej vgrajene. To ne pomeni, da smo z odstranitvijo funkcije odstranili tudi potrebo oz. postopek, kjer se funkcija potrebuje. To bomo namreč zaznali kasneje pri pripravi postopkovnega modela. Na takšen način že dobimo grobo skico postopkovnega modela, v katerem so na več mestih lahko opazne prekinitve postopka.

SLIKA 8: GROBA SKICA POSTOPKOVNEGA MODELA ZA PRIMER PRIJAVE BREZPOSELNE OSEBE V EVIDENCO



V primeru, da aplikacije še ni, je potrebno narediti analizo postopkov dela. Tu je potrebno postopke dela spoznavati od začetka. V tem primeru nimamo še nobenih opornih točk, kot v prvem primeru. Tudi tu je potrebno postopke najprej grobo skicirati.

3.3 Postopkovni (procesni) model

Procese¹⁸ pogosto zelo težko ločimo od postopkov¹⁹. Zaradi tega lahko pri pripravi modela procese in postopke združimo. Na Zavodu, pri razvoju programske opreme, razumemo proces, kot način dela po materialnih predpisih (zakonu, organizacijskem pravilniku ali ostalih predpisih). Kot postopek razumemo način dela za izvajanje procesov. Pri pripravi postopkovnega (procesnega) modela, je potrebno poznati vse postopke operativnega izvajanja posameznih nalog. Informacije in manjkajoče podatke je potrebno pridobiti pri uporabnikih oziroma vsebinskih nosilcih. Skico (model) dopolnimo z manjkajočimi postopki ter jih kratko opišemo.

Primer kratkega opisa postopka:

Prijava

Brezposelna oseba (BO) se prijavi pri referentu. Osebi se izpiše prijavitni obrazec z vsemi podatki, ki so za osebo pomembni in ga le ta tudi podpiše. Določi se ji svetovalac zaposlitve in termin za prvi intervju. Zraven dobi oseba še vso potrebno gradivo. V primeru manjkajočih dokumentov se tudi ti izpišejo in določi se rok za dopolnitev. V primeru, da oseba ne izpolnjuje pogojev za vpis se ji izda zavrnilna odločba.

Skica in postopek se nenehno dopolnjujeta in popravljata. Iz kratkega opisa nastaja postopoma podroben opis postopka, iz grobe skice pa podrobna skica.

Primer podrobnega opis za prijavo oseb v evidenco z zahtevami uporabnikov:

Opis postopka dela:

Brezposelna oseba se javi v sprejemni pisarni. Tu dobi informacije, kaj potrebuje za prijavo v evidenco (za BO osebni dokument, delavsko knjižico, dokument o prenehanju delovnega razmerja, za evidence po drugih zakonih pa osebni dokument in dokazilo o statusu), če seveda oseba že prej ni pridobila vseh informacij na samopostrežen način. Ko oseba zadosti pogojem za prijavo, se prične s postopkom prijave oz. s postopkom prijave se prične tudi, ko oseba to zahteva. Tudi če oseba ne izpolnjuje vseh pogojev, se s postopkom prijave lahko prične. Rezultat pa je lahko delna prijava do izpolnitve pogojev ali pa zavrnilna odločba. V postopku prijave se preveri pogoje za prijavo in se izpiše iz aplikacije prijavitni obrazec. Pogoji za prijavo so opisani v organizacijskem predpisu številka CS 041 03 015 – Način in vsebina vodenja uradnih evidenc brezposelnih oseb,

¹⁸ Proces – med seboj povezani pojavi, ki se vrstijo po določenih zakonitostih; celota del, delovanja za doseg kakega cilja (SSK 1995).

¹⁹ Postopek – oblika načrtnega, premišljenega dela, delovanja, ravnanja ali mišljenja za doseg kakega cilja (SSK 1995).

oseb prijavljenih pri Zavodu na podlagi drugih zakonov, oseb, ki jim pravice po tem zakonu mirujejo in oseb vključenih v ukrepe aktivne politike zaposlovanja.

Opisu posameznih postopkov sledijo še opombe, predlogi in kritike, kamor vsi, ki so sodelovali pri opisu, podajajo svoje opombe in predloge, ki niso bili zajeti v opis oz. jih niso mogli ali znali definirati. Tu so zabeležene tudi dileme, ki jih mora vodstvo projekta razčistiti skupaj z usmerjevalno skupino projekta. Tukaj se tudi zabeležijo kritike, ki so jih imeli člani usmerjevalne skupine in jih bo potrebno pri naslednji dopolnitvi opisa upoštevati. Primer opomb, predlogov in kritik:

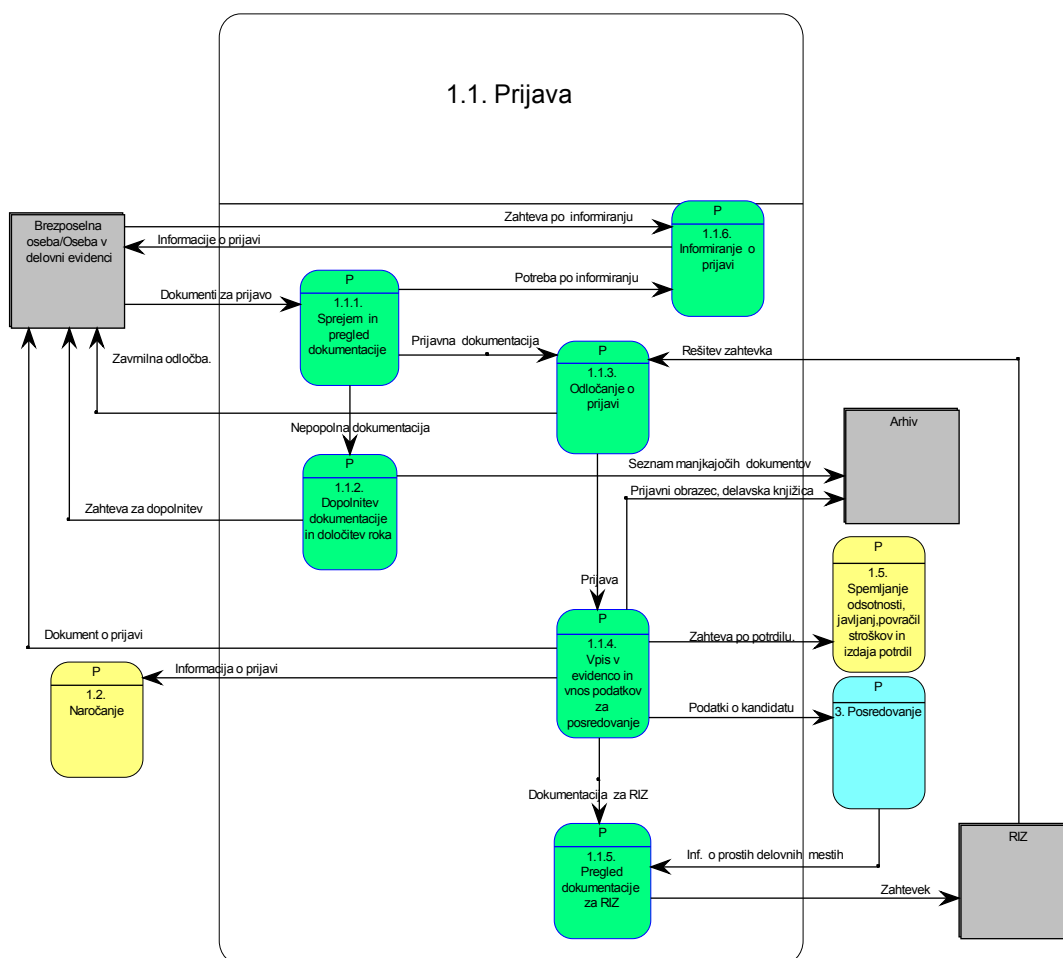
Opombe, predlogi, kritike

Zagotoviti je potrebno beleženje dostopov, vnosa in spreminjanja podatkov v aplikacijo. Podatki se trajno hranijo v arhivu. Osnovno načelo je, da ima tisti, ki brezposelno osebo obravnava tudi pravico do vpogleda, vnosa in spreminjanja podatkov.

Potrdila o neprijavljenosti bodo na razpolago v sprejemni pisarni na predtiskanem obrazcu ter na internetu. Oseba sama izpolni obrazec. Referent preveri ime in priimek osebe iz osebnega dokumenta in obrazca, preveri morebitni vpis v evidenci in obrazec potrdi. Cilj pa je, da se pristojni z ustreznimi institucijami dogovorijo, da teh potrdil ne bodo zahtevali - sprememba zakonodaje.

Vsi dokumenti o brezposelni osebi se shranjujejo v posebnem dosjeju – (mapi) v arhivu, ki se nahaja na uradu za delo. Vsi dokumenti, ki se o osebi zberejo na območni službi, se v najkrajšem možnem času dostavijo na urad, kjer je oseba prijavljena. Vsi delavci Zavoda dokumente, ki se nanašajo na določeno brezposelno osebo, shranjujejo v to mapo, ki je tem delavcem po potrebi tudi na vpogled.

Na osnovi opisa se pripravijo podrobne skice (diagrami tokov podatkov). Primer diagrama tokov podatkov (DTP diagram):

SLIKA 9: DTP DIAGRAM PRIJAVE

3.4 Podatkovni model

Pripravi se opis podatkov s skrajšano vsebino šifrantov. Opis pripravijo vsebinski nosilci na naslednji način:

Splošni (identifikacijski podatki) o brezposelni osebi in razlog vpisa v evidenco:

- Priimek
- Ime
- Naslov stalnega bivališča (ulica + številka, poštna številka in naziv kraja)
- EMŠO (enotna matična številka občana) – iz EMŠO je razviden tudi datum rojstva in spol
- Šifra UE (upravna enota)
- Razlog vpisa v evidenco in status osebe ob prijavi

Šifrant razlogov vpisa:

0. oz. prazno: oseba v postopku razvrščanja (manjkajoča dokumentacija)

1. brezposelna oseba
 - 1.1 iskalec prve zaposlitve
 - 1.2 prenehanje delovnega razmerja
 - 1.3 lastnik ali solastnik gospodarske družbe
 - 1.4 samozaposlen
 - 1.5 lastnik, zakupnik, najemnik ali drug uporabnik kmetijskih ali gozdnih zemljišč
 - 1.6 iztek javnega dela
 - 1.7 prekinitev javnega dela
 - 1.8 iztek izobraževanja po 53. b členu²⁰
 - 1.9 prekinitev izobraževanja po 53. b členu
2. oseba v evidenci po drugih zakonih
 - 2.1. delovnih invalidov, ki so prejemniki denarnega nadomestila na ZPIZ
 - 2.2. Invalid II. in III. kategorije
 - 2.3. Delavec, katerega delo je postalo trajno nepotrebno v skladu s predpisi o delovnih razmerjih
 - 2.4. Vključen v vladni projekt P-11
 - 2.5. Oseba, ki ji je delo ogroženo zaradi potresa
 - 2.6. Presežni delavci zaradi prisilne poravnave
 - 2.7. Nezmožni za delo iz zdravstvenih razlogov (nad 30 dni, ob nastanku brezposelnosti)
 - 2.8. Tujci prejemniki DN (denarnega nadomestila) oz. DP (denarne pomoči) brez osebnega delovnega dovoljenja

Na podlagi opisa podatkov se pripravi podatkovni model. Podatkovni model pripravijo informatiki. Definirajo se tabele v katere se bodo podatki zapisovali in šifranti. Vsaka tabela oz. šifrant vsebuje podatke o atributih, formatu zapisa, kratek opis posameznega atributa, podatek o obveznosti vpisa in opombo za posamezen atribut.

Primer tabel in šifrantov za prijavo brezposelne osebe v evidenco:

Prijava

Tabela prijav v evidenco brezposelnih oseb oziroma evidenco po drugih zakonih.

²⁰ 53. b člen Zakona o zaposlovanju in zavarovanju za primer brezposelnosti.

Atribut	Format	Opis	Obvezen vpis	Opombe
GUIDprijave	uniqueidentifier	GUID ²¹ št. prijave osebe v evidenco BO/DZ ²²	Da	PK ²³ , default
GUIDosebe	uniqueidentifier	GUID št. osebe	Da	FK ²⁴
DatPrijave	datetime	Datum prijave v evidenco	Da	PV ²⁵ = getdate()
DatStatPrijave	char (6)	Datum statistične prijave v evidenco (v obliki MMLLLL)	Da	
IDuvrstitve	smallint	ID ²⁶ št. uvrstitve v eno izmed evidenc (BO, DZ)	Ne	FK PV= 1
DatUvrstitve	datetime	Datum uvrstitve v evidenco BO/DZ	Ne	
IDSvetovalca	int	ID št. svetovalca zaposlitve BO	Ne	FK
CrtnaKoda	varchar (20)	Črna koda prijave v evidenco	Ne	
InetObjava	bit	Ali je BO/DZ privolil v objavo svojih podatkov na internetu?	Ne	
Referent	tipReferent	Referent, ki je zadnji spremenil zapis	Da	
IDenote	smallint	ID št. org. enote Zavoda iz katere je prišla sprememba	Da	
DATSP	smalldatetime	Datum spremembe podatka	Da	PV= getdate()

Primarni ključ: GUIDprijave

Tuji ključi: FK_ZPPrijava_Osebe
FK_ZPPrijava_SifZPUvrsEvidenca
FK_ZPPrijava_SifZaposleni

Entitete in attribute je potrebno med seboj še povezati. Za prikaz povezav med njimi se uporablja entitetno-relacijski diagram (E-R diagram). Primer takšnega E-R diagrama prikazuje spodnja slika:

²¹ GUID: enoznačna identifikacijska številka.

²² DZ – evidenca po drugih zakonih

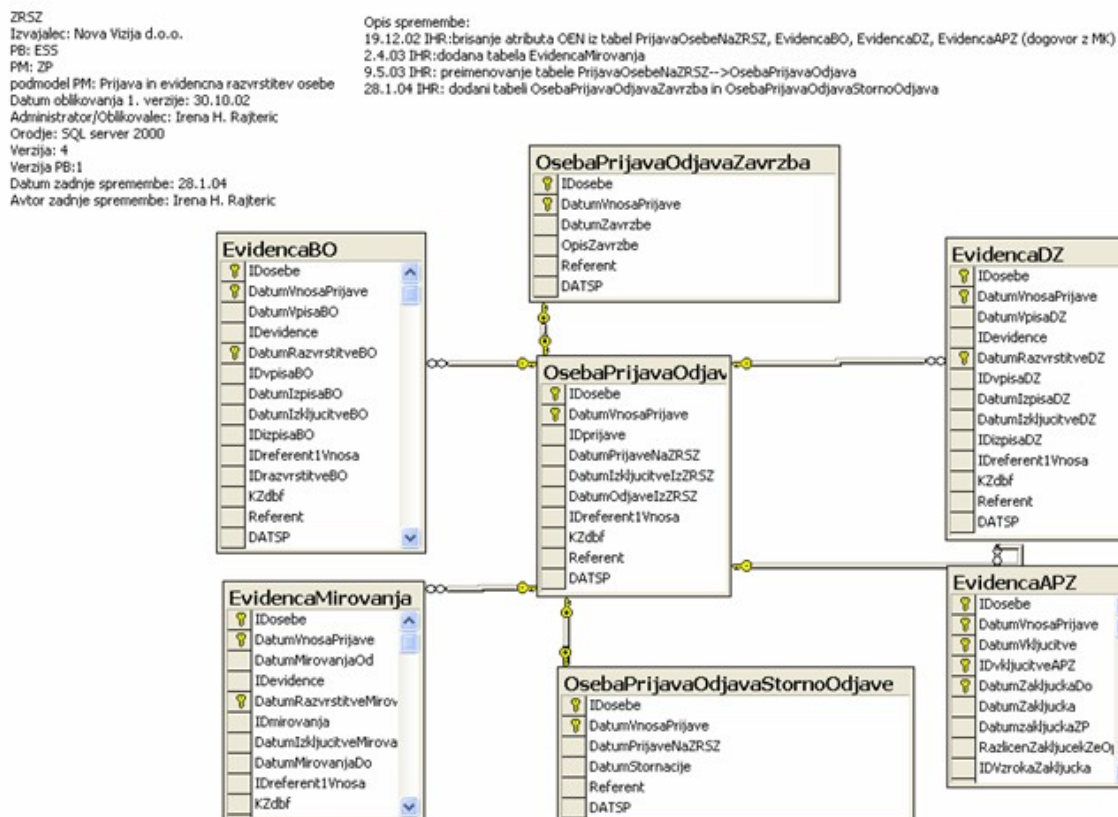
²³ PK: primarni ključ.

²⁴ FK: tuji ključ.

²⁵ PV: privzeta vrednost

²⁶ ID št.: identifikacijska številka

SLIKA 10: E-R DIAGRAM



3.5 Funkcionalnost aplikacije (tipke)

Za vsako aplikacijo je potrebno opredeliti pomen funkcijskih tipk. Na primer:

F1	Pomoč
F2	Odpiranje šifrant (referenčni podatek)
F3	Iskanje osebe
F4	Zaposlitveni načrt

3.6 Ekranski vmesniki

Ekranski vmesnik je namenjen samo prikazu ter približni razvrstitvi podatkov, ki jih želimo prikazati. Izvedba samega vmesnika je prepuščena programerjem, možna pa so tudi večja odstopanja od same zasnove.

Primer predlaganega ekranskega vmesnika je prikazan na naslednji sliki.

SLIKA 11: PREDLAGAN EKRANSKI VMESNIK ZAVIHKKA OSEBE

Osebe. | Izobrazba in izkušnje | Delovna Doba | Prijava. | Dosegljivost | Zadnja zaposlitev | Naročanje | Izposoja DK | Intervju | Intervju 1 | Zaposlitveni načrt

Osební podatki:

Priimek: Davčna številka:
 Ime: Bančna številka:
 EMŠO: Pokl./strok. izobrazba:
 Dat. Rojstva: Stopnja izobrazbe:
 Spol: Telefonska številka:
 Državljanstvo: Elektronski naslov:

☒ Posredovanje da/ne

Naslovi:

Naslov bivališča	Hišna št.	Pošta	TipNaslova	Upravna enota	Občina
Tržaška	5	1000	Ljubljana	stalni	Lj. Vič Rudnik
Podmilškakova	5	1000	Ljubljana	kontaktni	Lj. Bežigrad

Zadnja prijava:

Evidenca: Prijava od: Do: Vzrok:

Referent: Datum spremembe:

3.7 Specifikacija zahtev za programsko opremo

Za lažje programiranje in v pomoč programerjem se pripravi še specifikacija zahtev za programsko opremo. Specifikacijo pripravijo uporabniki skupaj z informatiki. Obvezna poglavja, ki jih mora takšna specifikacija na Zavodu vsebovati so:

Naslovna stran

1. Predstavitev izhodiščnega stanja
 - 1.1. Namen
 - 1.2. Sopomenke in okrajšave
 - 1.3. Reference
 - 1.4. Sosledje verzij
2. Opis predlagane programske opreme oz. posamezne funkcije programske opreme
 - 2.1. Opisi in definicije
 - 2.2. Splošna pravila
 - 2.3. Podatkovni model
 - 2.4. Uporabniški vmesniki
 - 2.5. Informacijski pogledi
 - 2.6. Referenčni podatki (šifranti)

2.7. Funkcije in procesi

3. Zaključki in opombe analitika

Naslovna stran vsebuje podatke o naročniku (Zavodu), naslov specifikacije zahtev, podatke o dokumentu (tip dokumenta, identifikator, verzijo, komu je dokument namenjen, naziv projekta, datum oblikovanja, avtorje) in podatek o zaščiti dokumenta.

V predstavitvi izhodiščnega stanja je potrebno opredeliti namen programske opreme in cilj, ki ga mora produkt zadovoljiti. Našteti je potrebno vse sopomenke, ki bodo uporabljene in opisati vse kratice. Pri referencah je potrebno našteti vse dokumente na katere se specifikacija zahtev sklicuje oziroma se v njih nahajajo dodatna pojasnila. Sosledje verzij pomeni, da se spremlja in v dokumentu označuje vsaka nastala sprememba v njemu.

Pri opisu programske opreme je najprej potrebno natančno opisati proces in način dela. Opisati je potrebno definicije vseh pojmov, ki se bodo v dokumentu uporabljale. Podati je potrebno vsa splošna pravila, ki veljajo na nivoju atributne, domenske in referenčne integritete oziroma na nivoju vnosa, sprememb ali brisanja podatkov. Pri podatkovnem modelu morajo biti opisane vse entitete (tabele) z opisom atributov, ki jih vsebujejo.

Našteti in opisani morajo biti vsi uporabniški vmesniki, ki jih bodo uporabniki potrebovali (ekranski vmesniki za potrebe vnosa, popravljanja, spreminjanja, iskanja itd.). Ravno tako morajo biti predstavljeni vsi želeni in pričakovani pogledi v podatke (informacijski pogledi). Našteti morajo biti vsi šifranti, ki bodo uporabljeni v programski rešitvi.

Preglednica funkcij ali procesov podaja seznam, opis in prioriteto tistih funkcij IS, ki jim mora le-ta zadovoljiti. Implementacija funkcij je predmet načrta in implementacije programske opreme.

V zaključku morajo biti podane vse opombe, ki še niso razrešene. Specifikacija zahtev ne more biti podlaga za programiranje, dokler vse opombe nimajo podanih rešitev.

Dokument mora biti pregledan in potrjen s strani vsebine, kar mora biti v njemu tudi napisano (datum in kdo je dokument potrdil).

3.8 Javni razpis

V primeru naročanje aplikacije pri zunanjem izvajalcu je potrebno opraviti javni razpis in opraviti izbor najugodnejšega ponudnika (po veljavni zakonodaji – Zakonu o javnih naročilih).

3.9 Prenos podatkov

V primeru, da gre pri prenovi programske opreme tudi za spremembo baze podatkov in se bodo pri novi programski opremi uporabljali tudi stari podatki, je potrebno podatke iz stare baze prenesti v novo. Potrebno je pripraviti prenosni postopek, kjer je potrebno opredeliti, kam se kateri podatek prenese. Podatke je potrebno večkrat preveriti na testni bazi, dokler se z novo programsko opremo ne preveri pravilnost podatkov. Zadnji prenos se opravi na produkcijsko bazo zadnji dan pred uvedbo nove programske opreme.

3.10 Programiranje

Na osnovi razpisne dokumentacije in specifikacij zahtev za programsko opremo se prične s programiranjem aplikacije. Programiranje se izvaja po posameznih zaključenih sklopih (modulih), ki se testirajo. Napake se javljajo programerjem takoj. Prejeti popravki, se ponovno testirajo. To se ponavlja do zadnje verzije, ki izpolnjuje vse zahteve. V času programiranja je potrebno z izvajalci tesno sodelovati in glede na napredovanje programiranja tudi prilagajati oz. spreminjati zahteve. Predlogi za izboljšanje se lahko prejmejo tudi od izvajalcev, ki lahko pogosto vidijo boljše rešitve, kot so bile predlagane.

3.11 Testiranje

Testiranje verzij aplikacije se izvaja po sklopih, kot prihajajo v testiranje. Za vsak sklop je potrebno pripraviti testirni načrt, ki mora vsebovati vse, kar se bo testiralo. Obvezno je potrebno preveriti, če:

1. sklop deluje v skladu z opisanim postopkom,
2. se zaporedje postopka aplikacije ujema s postopkom dela,
3. se vsi vpisani podatki shranijo in zapišejo v prave tabele,
4. sklop vsebuje vse zahtevane podatke,
5. vse deluje brez napak,
6. vse funkcijske tipke delujejo v skladu s specifikacijo,
7. sklop deluje v skladu z vsemi zahtevami specifikacije programske opreme,
8. so vsi izpisi popolni (glede postavljenih zahtev) in
9. je sklop učinkovit, enostaven, razumljiv.

Vse pomanjkljivosti je potrebno opisati v posebnem dokumentu – zahtevku programerjem, ki mora vsebovati naslednje podatke: opis problema, predlog rešitve, ime osebe, ki lahko poda dodatne informacije, in opombe.

Testiranje se prekine v primeru hujših napak, ko programska oprema zaradi njih ne omogoča nadaljnjega testiranja. Po prejemu popravkov je potrebno pričeti s testiranjem sklopa od začetka.

Po zaključenem programiranju celotne aplikacije je potrebno ponoviti testiranje celotne aplikacije. Pri tem se testira funkcionalnost celotnega sistema, testira se uporabniška dokumentacija in preveri tehnična dokumentacija.

Po končanem testiranju vsakega sklopa je potrebno pripraviti poročilo o preverjanju sklopa, na koncu testiranja celotne aplikacije pa poročilo o testiranju aplikacije.

3.12 Kakovost

Kakovost se preverja pri uporabnikih, če to res tisto, kar so želeli. Vsak delni rezultat procesa se kritično pregleda, oceni ali celo izmeri, rezultati pa se primerjajo z želenimi rezultati.

Preveriti je potrebno, če je programska oprema funkcionalna, zanesljiva, uporabna, učinkovita, omogoča vzdrževanje in je prenosljiva.

3.13 Izobraževanje

Osnovno izobraževanje mora biti izvedeno vsaj tri mesece pred uvedbo programske opreme v produkcijo. S tem si zagotovimo dovolj časa, da je možno programsko opremo še ustrezno popraviti oziroma dopolniti z uporabniškimi zahtevami. Pogoji za izvedbo izobraževanja je uporabniška dokumentacija, ki mora biti popolna. Programska oprema ne sme podpirati nobene funkcije, ki ni opisana v uporabniški dokumentaciji.

Po usposabljanju (seznanjanju) za delo s celotno programsko opremo se pri zahtevnejših aplikacijah, do uvedbe v produkcijo, izvajajo še izobraževanja po sklopih, ki se jih udeležijo uporabniki, ki bodo te sklope uporabljali. Pri manj zahtevnih aplikacijah pa se še enkrat izvede celotno usposabljanje.

V primeru, da se programska oprema uvaja v produkcijo po modulih, se pred vsakim uvajanjem izvede izobraževanje za ta modul.

3.14 Uvajanje v produkcijo

Programska oprema se lahko v produkcijo uvede v celoti, na en dan, za vse uporabnike ali postopno po modulih. Ravno tako se lahko programska oprema

uvede v produkcijo za vse uporabnike ali pa po posameznih organizacijskih enotah.

3.15 Upravljanje in vzdrževanje

Programska oprema mora zagotavljati čim bolj enostavno upravljanje in vzdrževanje. To pomeni, da lahko inštalacijo novih verzij opravi vsak informatik (morda celo uporabnik) po navodilih v tehnični dokumentaciji. Programska koda mora biti dobro dokumentirana, da jo bodo lahko popravljali tudi tisti programerji, ki pri razvoju niso sodelovali. Spremembe v šifrantih morajo biti neodvisne od programske kode, kar pomeni, da če se v šifrantih karkoli spremeni, to ne sme vplivati na delovanje aplikacije (dodajanje novih šifer, atributov, spreminjanje dolžine atributov...).

Programska oprema mora omogočati enostavno izvajanje popravkov, sprememb in dopolnitev. Pri teh spremembah mora ostati programska oprema stabilna, v testiranje sprememb pa mora biti vloženega kar najmanj napora.

Upravljanje in vzdrževanje je zadnja faza pri izdelavi programske opreme. Razvoj programske opreme namreč nikoli ni dokončan. Vedno se namreč pojavi kakšna nova zahteva (zakonska sprememba, uporabniška zahteva, odkritje napake, lepotni popravek, sprememba tehnologije itd.). V takšnem primeru je nujen poseg v programsko kodo in sprememba programa. Vzdrževanje mora biti izvedeno s čim manjšimi stroški in v čim krajšem možnem času. V primeru, da programiranje izvaja zunanji izvajalec morajo biti stroški in čas opredeljeni v pogodbi o vzdrževanju, v primeru notranjih programerjev pa se čas opredeli v zahtevi.

4 PREVERJANJE UPORABNOSTI METODE

4.1 Projekt naročila aplikacije ZP_Net

12. septembra leta 2000 je vodstvo Zavoda sprejelo odločitev, da se pristopi k prenovi osnovne aplikacije Zavoda – ZP (zaposlovanje). Obstoječi program je bil napisan v Clipperju, uporabljala pa se je DB2 podatkovna baza. Sam program je bil že na sami meji zmogljivosti operacijskega sistema DOS, saj je zasedel več kot 600 KB RAM, meja pa je bila 640 KB. Da smo lahko poleg te aplikacije poganjali še kaj drugega, smo morali uporabljati programe za optimizacijo spomina. Prenova je bila zaradi tega nujno potrebna, saj se je v program redno dodajalo nove funkcionalnosti.

Za namen prenove je bila ustanovljena delovna skupina z nazivom: „Prenova aplikacije zaposlovanja in priprava postopkovnega modela kot podlage za prenovo računalniške aplikacije za podporo dela svetovalcev zaposlitve“, ki bi jo lahko poimenovali tudi projektna skupina, in usmerjevalna skupina, ki bi se po metodi projektnega vodenja lahko imenovala tudi management projekta. Delo skupine se ni imenovalo projekt zato, ker skupina ni imela v naprej določenega terminskega plana in stroškov. Naloge te skupine so bile analizirati obstoječe stanje, pripraviti postopkovni model, pripraviti podatkovni model in pripraviti razpis za izbiro izvajalca. Postopkovni model lahko v tem primeru enačimo delom procesnega pristopa po ISO 9001:2000, saj je bilo potrebno identificirati procese, določiti zaporedje procesov in procese izboljšati.

Delovno skupino so sestavljali:

Vodja delovne skupine za pripravo postopkovnega in podatkovnega modela – Drago Perc

Člani:

Služba za zaposlovanje: Janez Arko, Valerija Okorn, Tadeja Džordžević, Marja Sedlaček, Duša Grižon, Valentina Plemenitaš.

Služba za analitiko: Alenka Gorše Dolinar, Jože Giodani.

Služba za izvajanje zavarovanja za primer brezposelnosti: Metka Barbo Škerbinc.

Služba za informatiko: Irena Hribar Rajterič, Slavko Kancilija, Judita Erjavec, Branka Kužnik in Marijan Prelec.

Usmerjevalno skupino so sestavljali:

Vodja: Mavricija Batič

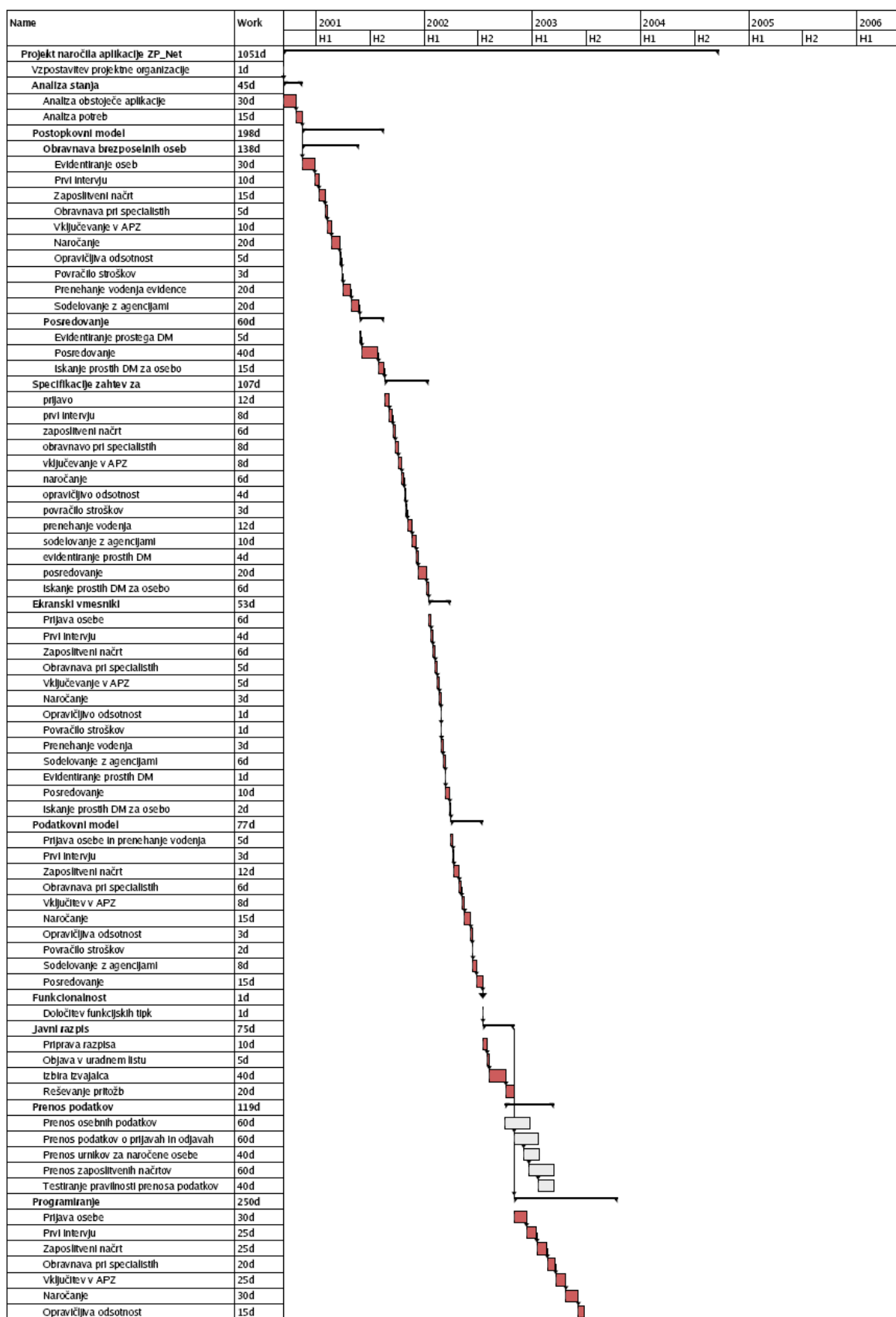
Člani: Sonja Pirher, Janez Urbas, Tone Furlan, Franc Belčič in Cvetka Sreš.

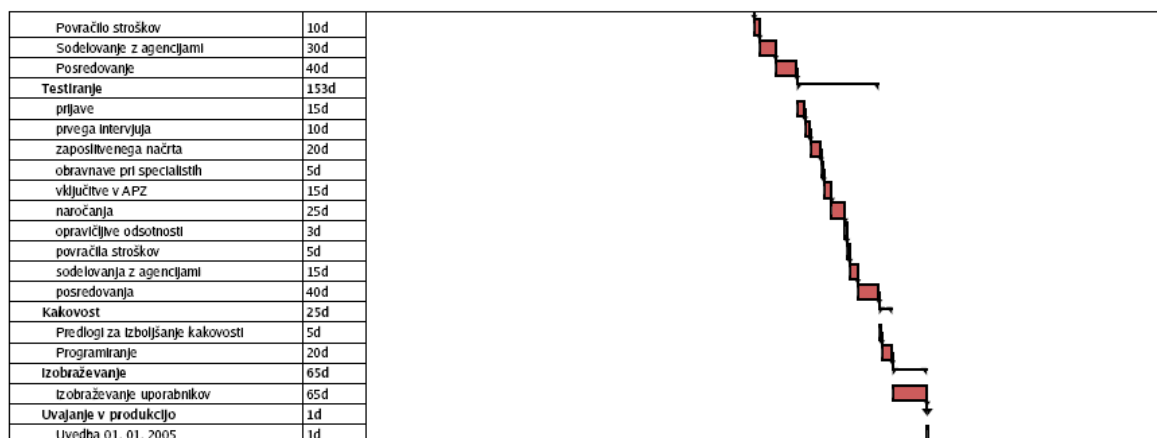
Delovna skupina je organizirala svoje delo kot projektna skupina, saj je pripravila seznam aktivnosti projekta, terminski plan pa je bil na žalost odvisen od drugih in ne od same skupine. Največjo oviro je namreč predstavljala prav usmerjevalna skupina, ki se tudi po dva meseca ni sestajala in je zato delo delovne skupine stalo.

4.1.1 Priprava projektne dokumentacije s terminskim planom

Vodja delovne skupine je pripravil terminski plan. Plan s korekcijami, ki so se izvedle med projektom ima naslednjo vsebino in obliko:

SLIKA 12: GANTOV DIAGRAM PRIKAZA PROJEKTA





Načrtovane aktivnosti projekta so prikazane v spodnji tabeli. V stolpcu „Slack“ so označene časovno kritične aktivnosti, ki vplivajo na čas trajanja projekta in število kritičnih delovnih dni do zaključka projekta.

TABELA 3: AKTIVNOSTI PROJEKTA

WBS	Name	Start	Finish	Work	Duration	Slack
1	Projekt naročila aplikacije ZP_Net	sep 12	sep 21	1051d	1051d	0
1.1	Vzpostavitev projektne organizacije	sep 12	sep 12	1d	1d	0
1.2	Analiza stanja	sep 13	nov 14	45d	45d	1005d
1.2.1	Analiza obstoječe aplikacije	sep 13	okt 24	30d	30d	0
1.2.2	Analiza potreb	okt 25	nov 14	15d	15d	0
1.3	Postopkovni model	nov 15	avg 17	198d	198d	807d
1.3.1	Obravnava brezposelnih oseb	nov 15	maj 25	138d	138d	867d
1.3.1.1	Evidentiranje oseb	nov 15	dec 26	30d	30d	0
1.3.1.2	Prvi intervju	dec 27	jan 9	10d	10d	0
1.3.1.3	Zaposlitveni načrt	jan 10	jan 30	15d	15d	0
1.3.1.4	Obravnava pri specialistih	jan 31	feb 6	5d	5d	0
1.3.1.5	Vključevanje v APZ	feb 7	feb 20	10d	10d	0
1.3.1.6	Naročanje	feb 21	mar 20	20d	20d	0
1.3.1.7	Opravičljiva odsotnost	mar 21	mar 27	5d	5d	0
1.3.1.8	Povračilo stroškov	mar 28	mar 30	3d	3d	0
1.3.1.9	Prenehanje vodenja evidence	apr 2	apr 27	20d	20d	0
1.3.1.10	Sodelovanje z agencijami	apr 30	maj 25	20d	20d	0
1.3.2	Posredovanje	maj 28	avg 17	60d	60d	807d
1.3.2.1	Evidentiranje prostega DM	maj 28	jun 1	5d	5d	0
1.3.2.2	Posredovanje	jun 4	jul 27	40d	40d	0
1.3.2.3	Iskanje prostih DM za osebo	jul 30	avg 17	15d	15d	0
1.4	Specifikacije zahtev za	avg 20	jan 15	107d	107d	700d
1.4.1	prijavo	avg 20	sep 4	12d	12d	0
1.4.2	prvi intervju	sep 5	sep 14	8d	8d	0

WBS	Name	Start	Finish	Work	Duration	Slack
1.4.3	zaposlitveni načrt	sep 17	sep 24	6d	6d	0
1.4.4	obravnavo pri specialistih	sep 25	okt 4	8d	8d	0
1.4.5	vklučevanje v APZ	okt 5	okt 16	8d	8d	0
1.4.6	naročanje	okt 17	okt 24	6d	6d	0
1.4.7	opravičljivo odsotnost	okt 25	okt 30	4d	4d	0
1.4.8	povračilo stroškov	okt 31	nov 2	3d	3d	0
1.4.9	prenehanje vodenja	nov 5	nov 20	12d	12d	0
1.4.10	sodelovanje z agencijami	nov 21	dec 4	10d	10d	0
1.4.11	evidentiranje prostih DM	dec 5	dec 10	4d	4d	0
1.4.12	posredovanje	dec 11	jan 7	20d	20d	0
1.4.13	iskanje prostih DM za osebo	jan 8	jan 15	6d	6d	0
1.5	Ekranski vmesniki	jan 16	mar 29	53d	53d	647d
1.5.1	Prijava osebe	jan 16	jan 23	6d	6d	0
1.5.2	Prvi intervju	jan 24	jan 29	4d	4d	0
1.5.3	Zaposlitveni načrt	jan 30	feb 6	6d	6d	0
1.5.4	Obravnava pri specialistih	feb 7	feb 13	5d	5d	0
1.5.5	Vključevanje v APZ	feb 14	feb 20	5d	5d	0
1.5.6	Naročanje	feb 21	feb 25	3d	3d	0
1.5.7	Opravičljivo odsotnost	feb 26	feb 26	1d	1d	0
1.5.8	Povračilo stroškov	feb 27	feb 27	1d	1d	0
1.5.9	Prenehanje vodenja	feb 28	mar 4	3d	3d	0
1.5.10	Sodelovanje z agencijami	mar 5	mar 12	6d	6d	0
1.5.11	Evidentiranje prostih DM	mar 13	mar 13	1d	1d	0
1.5.12	Posredovanje	mar 14	mar 27	10d	10d	0
1.5.13	Iskanje prostih DM za osebo	mar 28	mar 29	2d	2d	0
1.6	Podatkovni model	apr 1	jul 16	77d	77d	570d
1.6.1	Prijava osebe in prenehanje vodenja	apr 1	apr 5	5d	5d	0
1.6.2	Prvi intervju	apr 8	apr 10	3d	3d	0
1.6.3	Zaposlitveni načrt	apr 11	apr 26	12d	12d	0
1.6.4	Obravnava pri specialistih	apr 29	maj 6	6d	6d	0
1.6.5	Vključitev v APZ	maj 7	maj 16	8d	8d	0
1.6.6	Naročanje	maj 17	jun 6	15d	15d	0
1.6.7	Opravičljiva odsotnost	jun 7	jun 11	3d	3d	0
1.6.8	Povračilo stroškov	jun 12	jun 13	2d	2d	0
1.6.9	Sodelovanje z agencijami	jun 14	jun 25	8d	8d	0
1.6.10	Posredovanje	jun 26	jul 16	15d	15d	0
1.7	Funkcionalnost	jul 17	jul 17	1d	1d	569d
1.7.1	Določitev funkcijskih tipk	jul 17	jul 17	1d	1d	0
1.8	Javni razpis	jul 18	okt 30	75d	75d	0
1.8.1	Priprava razpisa	jul 18	jul 31	10d	10d	0
1.8.2	Objava v uradnem listu	avg 1	avg 7	5d	5d	0
1.8.3	Izbira izvajalca	avg 8	okt 2	40d	40d	0
1.8.4	Reševanje pritožb	okt 3	okt 30	20d	20d	0
1.9	Prenos podatkov	okt 1	mar 14	119d	119d	397d
1.9.1	Prenos osebnih podatkov	okt 1	dec 23	60d	60d	179d
1.9.2	Prenos podatkov o prijavah in odjavah	okt 31	jan 22	60d	60d	237d
1.9.3	Prenos urnikov za naročene osebe	dec 2	jan 24	40d	40d	297d
1.9.4	Prenos zaposlitvenih načrtov	dec 20	mar 13	60d	60d	337d
1.9.5	Testiranje pravilnosti prenosa	jan 20	mar 14	40d	40d	397d

WBS	Name	Start	Finish	Work	Duration	Slack
	podatkov					
1.10	Programiranje	okt 31	okt 15	250d	250d	244d
1.10.1	Prijava osebe	okt 31	dec 11	30d	30d	0
1.10.2	Prvi intervju	dec 12	jan 15	25d	25d	0
1.10.3	Zaposlitveni načrt	jan 16	feb 19	25d	25d	0
1.10.4	Obravnava pri specialistih	feb 20	mar 19	20d	20d	0
1.10.5	Vključitev v APZ	mar 20	apr 23	25d	25d	0
1.10.6	Naročanje	apr 24	jun 4	30d	30d	0
1.10.7	Opravičljiva odsotnost	jun 5	jun 25	15d	15d	0
1.10.8	Povračilo stroškov	jun 26	jul 9	10d	10d	0
1.10.9	Sodelovanje z agencijami	jul 10	avg 20	30d	30d	0
1.10.10	Posredovanje	avg 21	okt 15	40d	40d	0
1.11	Testiranje	okt 16	maj 17	153d	153d	91d
1.11.1	prijave	okt 16	nov 5	15d	15d	0
1.11.2	prvega intervjuja	nov 6	nov 19	10d	10d	0
1.11.3	zaposlitvenega načrta	nov 20	dec 17	20d	20d	0
1.11.4	obravnav pri specialistih	dec 18	dec 24	5d	5d	0
1.11.5	vključitve v APZ	dec 25	jan 14	15d	15d	0
1.11.6	naročanja	jan 15	feb 18	25d	25d	0
1.11.7	opravičljive odsotnosti	feb 19	feb 23	3d	3d	0
1.11.8	povračila stroškov	feb 24	mar 1	5d	5d	0
1.11.9	sodelovanja z agencijami	mar 2	mar 22	15d	15d	0
1.11.10	posredovanja	mar 23	maj 17	40d	40d	0
1.12	Kakovost	maj 18	jun 21	25d	25d	66d
1.12.1	Predlogi za izboljšanje kakovosti	maj 18	maj 24	5d	5d	0
1.12.2	Programiranje	maj 25	jun 21	20d	20d	0
1.13	Izobraževanje	jun 22	sep 20	65d	65d	1d
1.13.1	Izobraževanje uporabnikov	jun 22	sep 20	65d	65d	0
1.14	Uvajanje v produkcijo	sep 21	sep 21	1d	1d	0
1.14.1	Uvedba 01. 01. 2005	sep 21	sep 21	1d	1d	0

4.1.2 Analiza obstoječega stanja in izdelava postopkovnega modela

Delovna skupina je svoje delo opravljala v dveh podskupinah in sicer podskupini za pripravo postopkovnega modela in podskupini za pripravo podatkovnega modela. Skupino za pripravo postopkovnega modela so sestavljali vsebinski nosilci področij in uporabniki obstoječe aplikacije, skupino za pripravo podatkovnega modela pa informatiki. Skupina za pripravo postopkovnega modela se je najprej dogovorila za metodo dela, nato pa so se razdelili še v manjše skupine, ki so pripravljale postopkovni model. Smiselna delitev je bila na skupino za delo z brezposelnimi osebami, na skupino za posredovanje v zaposlitev (brezposelne osebe, delodajalci in potrebe po delavcih) ter na skupino za analitične in statistične izhode, ki pa je začela delovati kasneje.

Ena od odločitev, ki jo je delovna skupina sprejela, je bila tudi ta, da se na staro aplikacijo popolnoma pozabi, ter da se vse pripravi od začetka. Pri tem je bilo potrebno upoštevati postopek dela na uradih za delo (pripraviti je bilo potrebno

analizo postopkov na različnih uradih in upoštevati najboljše postopke), zakonodajo in ostale predpise ter analitične in statistične izhode.

Aktivnosti so bile naslednje:

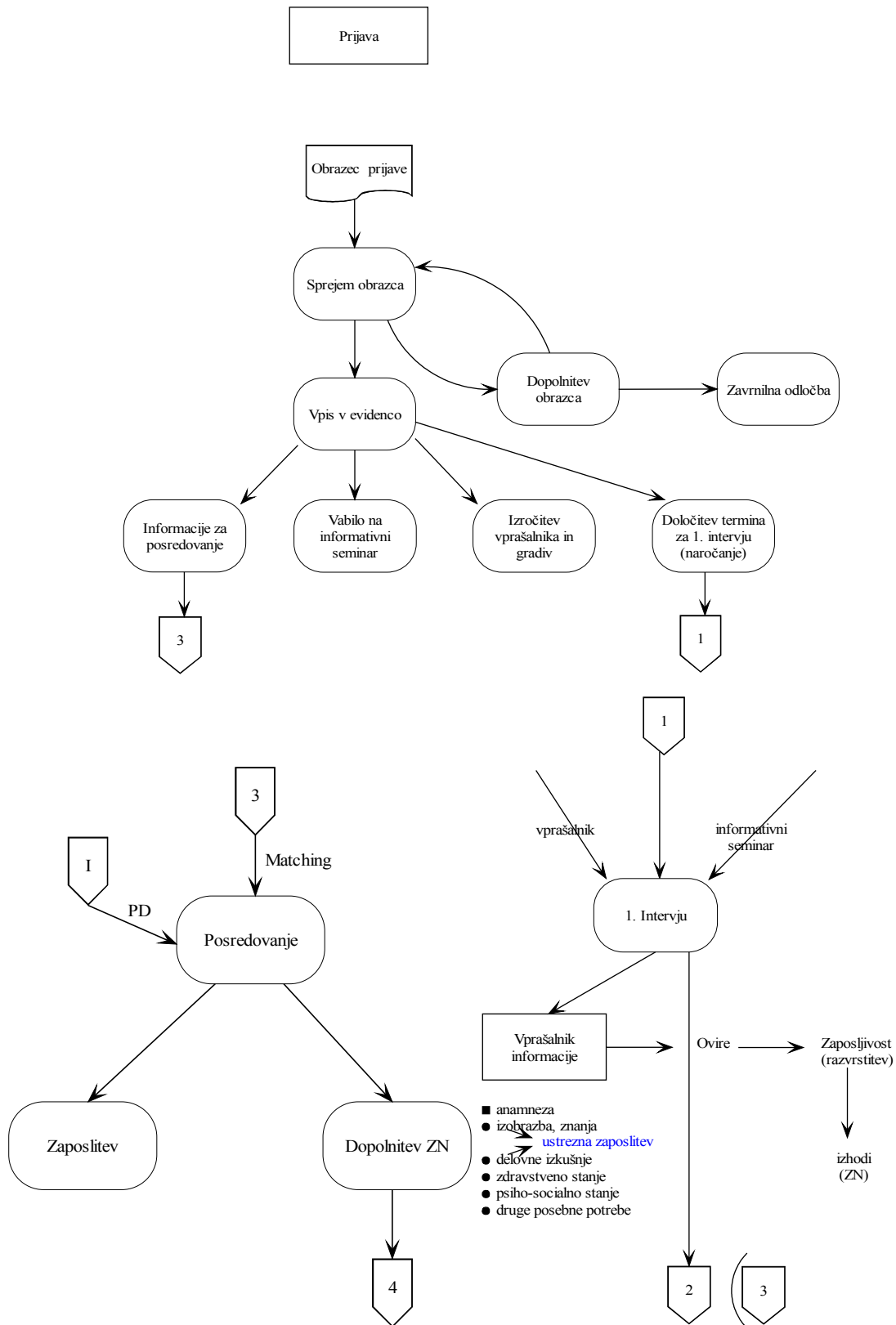
- analiza postopkov dela,
- izdelava grobe skice postopkovnega modela za celotno področje zaposlovanja,
- izdelava podrobnih skic in kratki opisi postopkov,
- podroben opis vsakega postopka,
- definiranje potrebnih podatkov,
- izdelava podatkovnega modela,
- priprava javnega razpisa,
- programiranje,
- prenos podatkov iz DB2 baze v MS-SQL,
- testiranje in
- uvajanje.

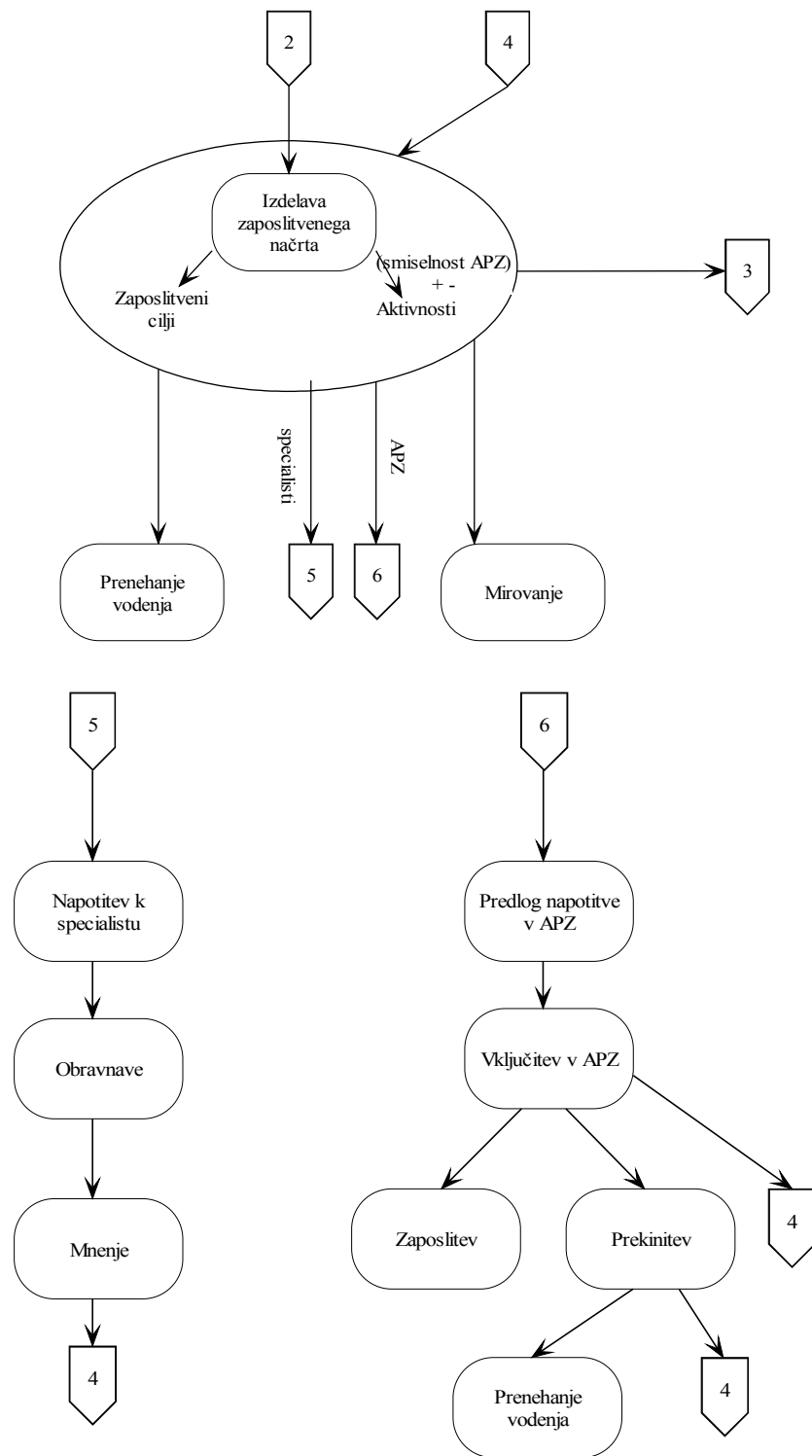
Pri analizi dela so bili upoštevani postopki dela na enem velikem uradu (Ljubljana), enem srednjem (Sevnica) in enem manjšem (Dravograd). Postopki dela, ki so bili enaki na vseh uradih so bili preneseni direktno v model, iz posebnosti, ki so izhajale zaradi različnih velikost uradov, pa so bile upoštevane najboljše rešitve. V primeru, da tudi to ni bilo možno, je bila sprejeta odločitev, da se podpre vse postopke.

Običajen postopek na manjših uradih je, da skrbnik vse potrebe po delavcih razdeli svetovalcem zaposlitve. Svetovalci se potem odločijo, katere brezposelne osebe bodo posredovali delodajalcem. Na Ljubljanskem uradu za delo pa so imeli zahtevo, da osebe posredujejo skrbniki sami. V tem primeru smo podprli oba načina dela.

Izdelane so bile grobe skice postopkovnega modela. Ena izmed njih je prikazana na spodnji sliki.

SLIKA 13: GROBA SKICA POSTOPKOVNEGA MODELA ZA PRIMER PRIJAVE IN OBRAVNAVE BREZPOSELNE OSEBE





Sledi kratek opis postopkov:

1. Prijava

Brezposelna oseba (BO) se prijavi pri referentu. Osebi se izpiše prijavitni obrazec z vsemi podatki, ki so za osebo pomembni in ga le ta tudi podpiše. Določi se ji

svetovalec zaposlitve in termin za prvi intervju. Zraven dobi oseba še vso potrebno gradivo. V primeru manjkajočih dokumentov se tudi ti izpišejo in določi se rok za dopolnitev. Če oseba ne izpolnjuje pogojev za vpis se ji izda zavrnilna odločba.

Po udeležbi na informativnem seminarju se oseba, na določen termin, zglaš pri svetovalcu zaposlitve, kjer opravi prvi svetovalni intervju. V primeru, da se informativnega seminarja oseba še ni udeležila se opravi individualno informiranje pri svetovalcu.

2. Prvi svetovalni intervju

Na prvem intervjuju svetovalec preveri podatke o izobrazbi²⁷, preveri, če je ustrezna zaposlitev pravilno določena glede na izobrazbo, preveri podatke o dodatnih znanjih in sposobnostih, delovnih izkušnjah, zdravstvenem stanju, psiho-socialnem stanju in drugih posebnih potrebah brezposelne osebe. Svetovalec določi tudi zaposljivost BO.

3. Zaposlitveni načrt

V zaposlitvenem načrtu (ZN) se določi ustrezna/primerna zaposlitev glede na poklic²⁸, zaposlitveni cilji in morebitna sprememba dosegljivosti. Osrednji del zaposlitvenega načrta so aktivnosti brezposelne osebe. Vsebina aktivnosti je odvisna o zaposljivosti BO. Ta je lahko neposredno zaposljiva, pogojno zaposljiva ali pa je to stranka s posebnimi potrebami.

Ob naslednjih intervjujih brezposelne osebe se spremlja realizacija aktivnosti. Po potrebi pa se ZN dopolni ali spremeni.

4. Obravnava pri specialistih

Svetovalec napoti BO k specialistu²⁹ na podlagi ZN. Če je specialist v Zavodu, svetovalec naroči BO na prosti termin specialista ter opredeli razlog napotitve. BO lahko obravnava tudi tim. Mnenje specialista oz. tima je namenjeno nadaljnji izdelavi ZN.

Iz kratkega opisa se je postopoma razvijal daljši in podroben opis postopkov.

²⁷ Izobrazba se pridobi v sistemu izobraževanja.

²⁸ Poklic je delo, ki ga človek lahko opravlja in je zanj potrebna določena usposobljenost (izobrazba).

²⁹ Notranji specialisti so poklicni svetovalci in rehabilitacijski svetovalci, zunanji specialisti, ki opravljajo delo za Zavod, pa so zdravniki svetovalci.

Podroben opis prijave v evidenco:

Brezposelna oseba se javi v sprejemni pisarni. Tu dobi informacije, kaj potrebuje za prijavo v evidenco (za evidenco BO osebni dokument, delavsko knjižico, dokument o prenehanju delovnega razmerja, za evidence po drugih zakonih pa osebni dokument in dokazilo o statusu), če seveda oseba že prej ni pridobila vseh informacij na samopostrežen način. Ko oseba zadosti pogojem za prijavo, se prične s postopkom prijave. S postopkom prijave se prične tudi, ko oseba to zahteva. Tudi, če oseba ne izpolnjuje vseh pogojev, se s postopkom prijave lahko prične. Rezultat je lahko delna prijava do izpolnitve pogojev ali pa zavrnilna odločba. V postopku prijave se preveri pogoje za prijavo. Pogoji za prijavo so opisani v organizacijskem predpisu številka CS 041 03 015 – Način in vsebina vodenja uradnih evidenc brezposelnih oseb, oseb prijavljenih pri Zavodu na podlagi drugih zakonov, oseb, ki jim pravice po tem zakonu mirujejo in oseb vključenih v ukrepe aktivne politike zaposlovanja.

Ob prijavi se izpiše prijavni obrazec. Osebi je prijavni obrazec dostopen tudi na internetu. Obrazec se lahko izpiše tudi, ko prijava še ni zaključena. Prijavni obrazec se izpiše po vnosu podatkov. Izpiše in izpolni ga oseba, ki dela v sprejemni pisarni – to je lahko katerakoli oseba, ki ima za to pooblastilo (referent, svetovalec, vodja urada...). Prijavni obrazec se ne izpiše samo v primeru, ko se s postopkom prijave ne prične. Osebi povemo, da ne izpolnjuje pogojev za prijavo in oseba odide – v tem primeru obrazca ne izpišemo. Obrazca tudi ne izpišemo, če oseba pri sebi nima osebnega dokumenta. Prijavni obrazec vsebuje tudi izjavo o vseh obveznostih (dosegljivost; obveznost aktivnega iskanja zaposlitve; obveznost izdelave zaposlitvenega načrta; obveznost dogovora/javljanja o opravičljivi odsotnosti; izjava o (ne)pridobivanju dodatnih dohodkov za osebe, ki uveljavljajo pravico do denarnega nadomestila; opsijsko izjava, da delodajalec ni izročil osebi delovne knjižice). Prijavni obrazec oseba podpiše. Oseba poleg prijavnega obrazca prejme tudi osnovne informacije - zloženke in vprašalnik. Vprašalnik prejmejo vse osebe, ki se prvič prijavljajo na Zavodu. Za ostale brezposelne osebe pa je potrebno vprašalnik preveriti in ugotoviti, če je še aktualen. Izpolnjenega vrnejo Zavodu (osebno ali po pošti) oz. ga prinesejo na prvi intervju pri svetovalcu. Določi se osebni svetovalec brezposelne osebe. Osebnega svetovalca določi oseba, ki prijavlja osebo v evidenco. Osebni svetovalec se nahaja na uradu, v katerem se oseba prijavlja. Sestavni del prijavnega obrazca je vabilo z datumom in uro prvega intervjuja pri svetovalcu (30 - 45 minut) ter vabilo na informativni seminar (v primeru, da je organizirano skupinsko informiranje). Oseba tako ob prijavi prejme le en dokument z vsemi izjavami (opcije) in vabili, ki ga podpiše. Če oseba prijavnega obrazca ne podpiše, zaključimo s postopkom prijave in je ne vpišemo v nobeno od evidenc. Če želi, lahko dobi tudi zavrnilno odločbo.

Če ima oseba vse potrebne dokumente in izpolnjuje pogoje, se jo vpiše v evidenco brezposelnih oseb oziroma v evidenco oseb prijavljenih pri Zavodu na podlagi drugih zakonov oz. v evidenco oseb, ki so vključene v program APZ (aktivne politike zaposlovanja). Pogoji se preverjajo ob vpisovanju podatkov v prijavni

obrazec. Če ne izpolnjuje pogojev oz. če nima vseh dokumentov, je oseba v postopku prijave. V evidenco se jo vpiše po dopolnitvi (v predvidenem roku za dopolnitev), drugače pa se osebi izda zavrnilna odločba in se jo v evidenco ne vpiše.) Osebo se vpiše v tisto evidenco za katero izpolnjuje pogoje. To je najprej v evidenco BO, če ne izpolnjuje pogojev za BO se osebo vpiše v evidenco po drugih zakonih, če pa tudi teh pogojev ne izpolnjuje se osebi izda zavrnilna odločba. Glej tudi organizacijski predpis CS 041 99 015 in pravilnik o vsebini in načinu vodenja uradnih evidenc s področja zaposlovanja (Ur. l. 85/2002).

V primeru, da oseba ne izpolnjuje pogojev za vpis, se ji izda zavrnilna odločba. Zavrnilno odločbo izda odgovorna oseba (skladno z Zakonom o splošnem upravnem postopku – ZUP). To je pooblaščen oseba, ki ima uspešno opravljeno preverja znanja iz ZUP.

Pri izdaji zavrnilne odločbe se v aplikacijo vnesejo samo naslednji podatki: ime in priimek, naslov, rojstni datum. Podatki o tej osebi se shranijo začasno za 6 mesecev. Po 6 mesecih se avtomatično izbrišejo. Oseba ima pravico do pritožbe na odločbo o zavrnitvi zahtevka za prijavo za vpis v evidenco brezposelnih oseb (ni dopolnil vloge, ni drugih pogojev za vpis) pri drugostopenjskem organu Zavoda. Če se oseba pritoži, se podatki ne brišejo. Vsaka zavrnilna odločba ima delovodno številko. Kaj se bo zgodilo, ko je pritožba obravnavana, je odvisno od sklepa drugostopenjskega organa Zavoda. Možni sta dve varianti. Ena je, da se pritožbi ugodi, druga pa je, da se pritožba zavrne.

Oseba ob izpolnjevanju pogojev za vpis v evidenco pridobi status brezposelne osebe oziroma status osebe prijavljene na Zavodu na podlagi drugih zakonov, ko je dokumentacija dopolnjena. S postopkom prijave pričnemo takoj, saj drugače osebi ne moremo izpisati prijavnega obrazca, na katerem je seznam manjkajočih dokumentov. Proces dopolnjevanja dokumentacije nima nobene omejitve glede števila ponovitev. Proces dopolnjevanja je odvisen od osebe. Oseba lahko prinese vso manjkajočo dokumentacijo naenkrat, lahko pa prinese samo del manjkajoče dokumentacije. V tem primeru moramo osebi še enkrat izpisati prijavni obrazec z manjkajočo dokumentacijo, ki ga mora oseba podpisati.

Ob prijavi pooblaščen oseba Zavoda od brezposelne osebe poizkuša pridobiti podatek o ustrezni zaposlitvi (izobrazba, ustreznost zaposlitev glede na poklic po SKP – standardna klasifikacija poklicev (šifrira se jo v celoti (6 mestna šifra), če to ne gre pa vsaj na 4 mesta natančno) – šifra ali opis dela v zadnjem letu pred nastankom brezposelnosti, tarifni razred). Če katerega od podatkov ni možno pridobiti takoj, se le-ta določi na prvem svetovalnem intervjuju. Namen je pridobiti osnovne podatke za posredovanje. Tako je lahko oseba že od prijave v evidenco vključena v posredovanje za zaposlitev.

Prijavljeni osebi se v prijavno - odjavni službi ali pri svetovalcu zaposlitve po potrebi izda potrdilo iz evidence (potrdilo o prijavi v evidenco).

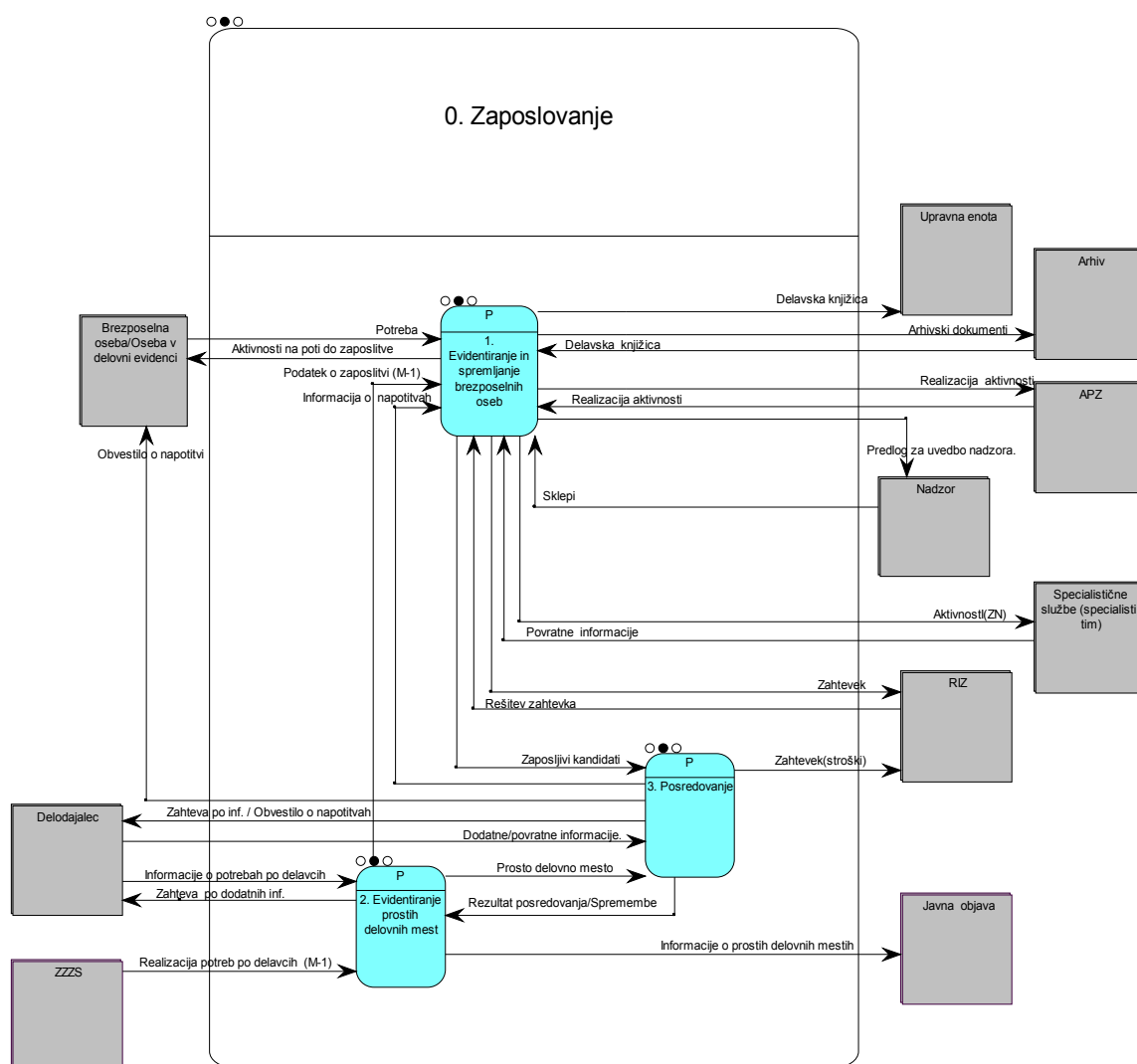
Opombe, predlogi, kritike

Zagotovi se beleženje dostopov, vnosa in spreminjanja podatkov v aplikacijo. Podatki se trajno hranijo v arhivu. Osnovno načelo je, da ima tisti, ki brezposelno osebo obravnava tudi pravico do vpogleda, vnosa in spreminjanja podatkov.

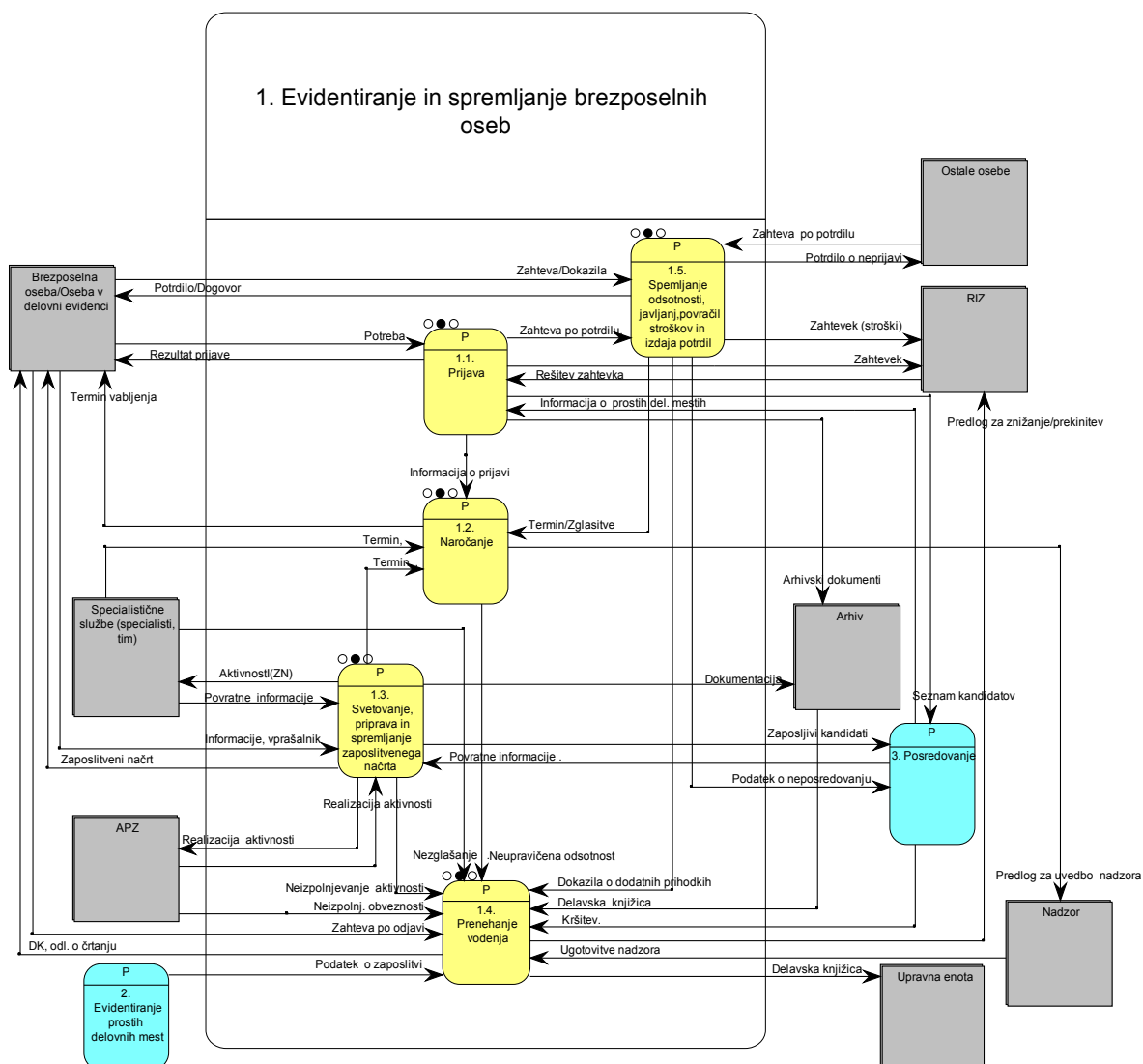
Potrdila o neprijavljenosti bodo na razpolago v sprejemni pisarni na predtiskanem obrazcu ter na internetu. Oseba sama izpolni obrazec. Referent preveri ime in priimek osebe iz osebnega dokumenta in obrazca, preveri morebitni vpis v evidenci in obrazec potrdi.

Vsi dokumenti o brezposelni osebi se shranjujejo v posebnem dosjeju – (mapi) v arhivu, ki se nahaja na uradu za delo. Vsi dokumenti, ki se o osebi zberejo na območni službi, se v najkrajšem možnem času dostavijo na urad, kjer je oseba prijavljena. Vsi delavci Zavoda dokumente, ki se nanašajo na določeno brezposelno osebo, shranjujejo v to mapo, ki je tem delavcem po potrebi tudi na vpogled.

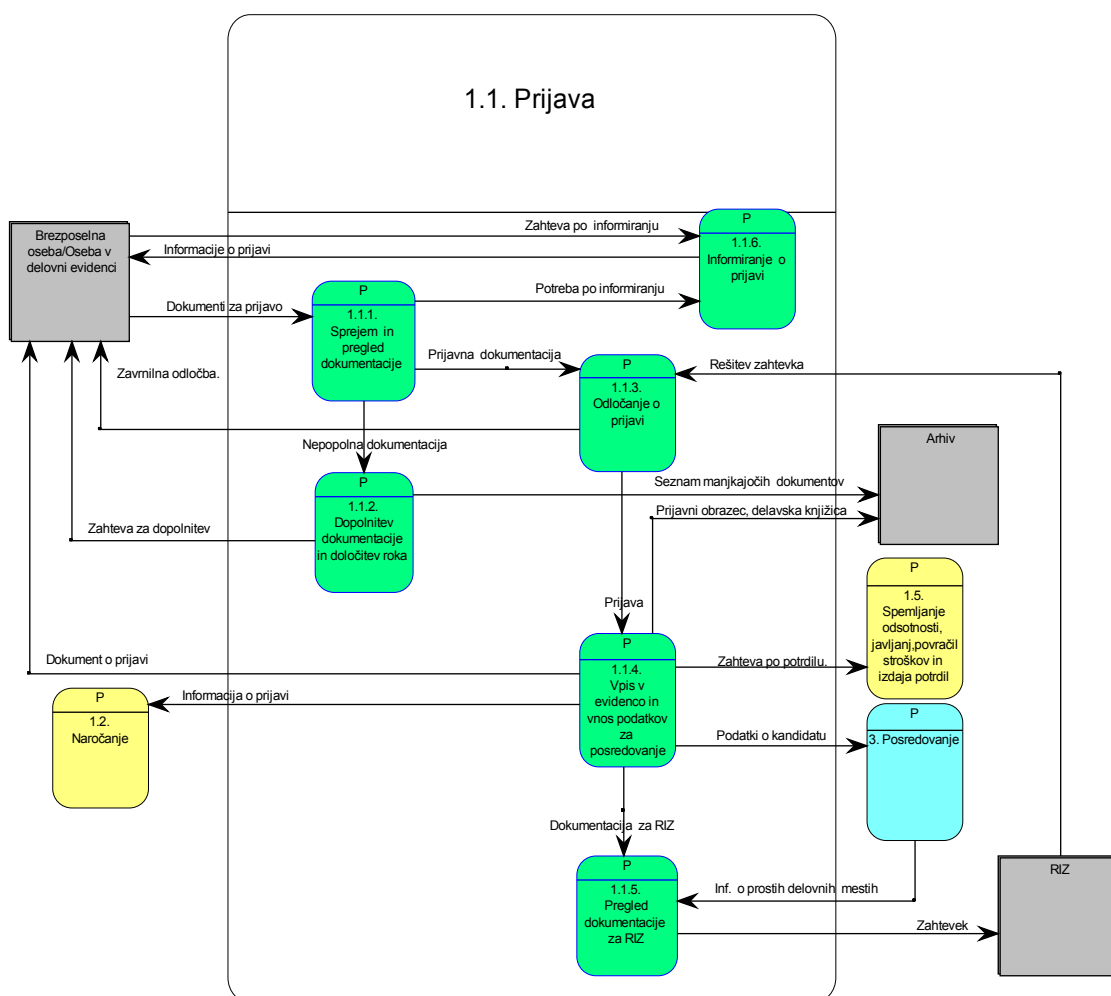
Na podlagi opisa so bili izdelani diagrami toka podatkov (DTP). Diagram „Proces Zaposlovanja“ nam prikaže glavne podprocese v procesu zaposlovanja in njihovo medsebojno povezavo. Glavni podprocesi so: evidentiranje in spremljanje brezposelnih oseb, evidentiranje prostih delovnih mest in posredovanje.

SLIKA 14: DIAGRAM PROCESA ZAPOSLOVANJA

Vsak od podprocesov je nadalje razgrajen na manjše samostojne podprocese. Na spodnji sliki je prikazan podproces „Evidentiranje in spremljanje BO“.

SLIKA 15: DTP DIAGRAM ZA EVIDENTIRANJE IN SPREMLJANJE BO

Tudi ti podprocesi se lahko še nadalje delijo. Na primer: podproces 1.1. Prijava se deli na podproces 1.1.1. Sprejem in pregled dokumentacije, 1.1.2. Dopolnitev dokumentacije in določitev roka, 1.1.3. Odločanje o prijavi itd. Ta delitev je prikazana na spodnji sliki:

SLIKA 16: DTP DIAGRAM ZA PRIJAVO OSEB V EVIDENCO

4.1.3 Definicije potrebnih podatkov

Za vse postopke dela je bilo potrebno definirati vse podatke. Primer za evidentiranje osebe in vpis v evidenco:

Splošni (identifikacijski podatki) o brezposelni osebi

- Priimek
 - Ime
 - Naslov stalnega bivališča (ulica + številka, poštna številka in naziv kraja)
 - EMŠO; iz EMŠO je razviden tudi datum rojstva in spol
 - Šifra UE
 - Razlog vpisa v evidenco in status osebe ob prijavi
0. oz. prazno: oseba v postopku razvrščanja (manjkajoča dokumentacija)
1. brezposelna oseba
- 1.1 iskalec prve zaposlitve

- 1.2 prenehanje delovnega razmerja
 - 1.3 lastnik ali solastnik gospodarske družbe
 - 1.4 samozaposlen
 - 1.5 lastnik, zakupnik, najemnik ali drug uporabnik kmetijskih ali gozdnih zemljišč.
 - 1.6 iztek javnega dela
 - 1.7 prekinitev javnega dela
 - 1.8 iztek izobraževanja po 53. b členu
 - 1.9 prekinitev izobraževanja po 53. b členu
2. oseba v evidenci po drugih zakonih
- 2.1. delovnih invalidov, ki so prejemniki denarnega nadomestila na ZPIZ
 - 2.2. Invalid II. in III. kategorije
 - 2.3. Nezmožni za delo iz zdravstvenih razlogov (nad 30 dni, ob nastanku brezposelnosti)
 - 2.4. Tujci prejemniki DN oz. DP brez osebnega delovnega dovoljenja

Razlog izdaje zavrnilne odločbe (obvezen vpis delovodne številke pri vseh odločbah)

1. v delovnem razmerju
2. lastnik ali solastnik gospodarske družbe
3. samozaposlena oseba
4. lastnik, zakupnik, najemnik ali drug uporabnik kmetijskih ali gozdnih zemljišč
5. upokojenec
6. invalid trajno in popolnoma nezmožen za delo
7. študent, dijak, vajenec ali udeleženec izobraževanja v skladu s 53. členom zakona
8. sodba - prestajanje zaporne kazni, daljše od šest mesecev
9. prejemanje odškodnine zaradi poteka ali prekinitve individualne pogodbe o zaposlitvi
10. prejemanje nadomestila zaradi konkurenčne prepovedi po pogodbi o zaposlitvi
11. nepopolna dokumentacija
12. tuj državljan brez osebnega delovnega dovoljenja
13. še ni preteklo 6 mesecev od odjave iz evidence zaradi kršitve obveznosti
14. izpolnil pogoje za upokojitev

Odločbe se shranjujejo v elektronski obliki 6 mesecev od datuma izdaje. V primeru pritožbe pa se odločba shrani za nedoločen čas.

Osnovni podatki, pomembni za posredovanje in ostali osnovni podatki

- Spol (podatek se generira iz EMŠO)
- Naslov dosegljivosti (ulica, številka in poštna številka)
- Šifra UE

- Čas in način dosegljivosti
- * čas: aplikacija ponudi čas od 11.00 do 14.00 z možnostjo spreminjanja.

Način dosegljivosti (možnost izbire več načinov dosegljivosti):

1. osebno in po pošti (privzeta in obvezna izbira na podlagi zakona)
 2. telefon
 3. elektronska pošta
- Kontaktni telefon (lahko mobilni telefon) (*podatek se vpisuje, ko je telefon definiran kot način dosegljivosti*)
 - Elektronska pošta (*podatek se vpisuje, ko je el. pošta definirana kot način dosegljivosti*)
 - Državljanstvo (*obstoječi šifrant je potrebno dopolniti z dvomestno šifro po standardu ISO 3166/ 1993*)

Primer šifranta:

Šifra	Šifra po ISO	Država
4	AF	AFGANISTAN
8	AL	ALBANIJA
705	SI	SLOVENIJA (privzeto)

Če državljanstvo ni enako slovenskemu je potrebno tujcu določiti naslednje podatke:

- bivanje

Šifrant vrst dovoljenj za prebivanje

Tekst
Začasno
Stalno
Brez

- upravna enota, ki je izdala dovoljenje za bivanje. Če je izdajatelj MNZ (ministrstvo za notranje zadeve) se vpiše UE, kjer se MNZ nahaja.
 - datum veljavnosti začasnega dovoljenja za bivanje
 - datum vložitve vloge za dovoljenje za bivanje (v primeru podaljševanja)
 - številka potnega lista
 - datum veljavnosti potnega lista
 - številka delovnega dovoljenja
 - veljavnost delovnega dovoljenja
- Bančna številka
 - Davčna številka

- Datum izposoje, oddaje ali pošiljanja delovne knjižice na upravno enoto.
- Datum prenehanja zadnje zaposlitve
- Mat. številka podjetja, naziv podjetja (prikaz na ekranu), v katerem je delovno razmerje prenehalo
- Razlog prenehanja zaposlitve:

Tekst
Pisna izjava ali sporazum
Nezmožnost opravljanja del oz. nedoseganje pričakovanih rezultatov dela
Neupravičene odsotnosti z dela 5 dni zaporedoma
Sklenitev delovnega razmerja v nasprotju z zakonom oz. neresnični podatki ob sklenitvi delovnega razmerja
Odklonitev razporeditve oz. pravic presežkov
Disciplinski ukrep prenehanja delovnega razmerja
Pisna odpoved delodajalca zaradi hujše kršitve delovne obveznosti ali namernega poškodovanja premoženja delodajalca
Neuspešno poskusno delo
Neopravljen izpit pripravnika
Prenehanje dejavnosti oz. obratovalnice zasebnega delodajalca
Prepoved opravljanja del po sodišču oziroma organu
Nastop prestajanja kazni zaporu ali VV ukrepa nad 6 mesecev
Razveza delavca zaradi neizpolnjevanja obveznosti delodajalca
Iztek delovnega razmerja za določen čas
Stečaj organizacije oziroma delodajalca
Likvidacija organizacije oziroma delodajalca
Trajni presežek
Prisilna poravnava
Ostali vzroki

- Poklicna / strokovna izobrazba, šifra izobrazbe
- Stopnja izobrazbe

Šifra	Tekst
01	nepismen
02	brez šole
03	I nepopolna osnovna šola
04	osnovna šola
05	II kategorija zahtevnosti
06	III kategorija zahtevnosti (2-letna šola)
07	IV kategorija zahtevnosti (3-letna šola)
08	V kategorija zahtevnosti (4-letna šola)
09	VI/1 kategorija zahtevnosti
10	VI/2 kategorija zahtevnosti
11	VII kategorija zahtevnosti
12	VII/2 kategorija zahtevnosti

13	VIII kategorija zahtevnosti (doktorji)
----	--

- Ustrezna zaposlitev po SKP (*šifra na ravni najmanj 4 - mestne šifre referent v sprejemni pisarni, nato došifrira svetovalec zaposlitve*)
- Tarifni razred (*če je razviden iz pogodbe o zaposlitvi ali iz drugih dokumentov; če ni, šifriranje naknadno po možnosti v istem dnevu kot datum prijave*)

Šifra	Tekst
1	enostavna dela
2	manj zahtevna dela
3	srednje zahtevna dela
4	zahtevna dela
5	bolj zahtevna dela
6	zelo zahtevna dela
7	visoko zahtevna dela
8	najbolj zahtevna dela
9	najbolj pomembna, najbolj zahtevna dela

- * *Podatek ali je oseba prejemnik socialne varnosti, se zagotovi z možnostjo vpogleda v evidenco prejemnikov in sicer neposredno iz aplikacije (prejemnik DN/DP, od do, datum prekinitve).*
- * *Podatke o delovni in zavarovalni dobi (matična številka ali naziv podjetja, trajanje zaposlitve od do, količnik delovnega časa, vrsta delovnega razmerja za delovno dobo) vnese v aplikacijo iz delavske knjižice referent ob prijavi v evidenco. Program izračuna skupno delovno in zavarovalno dobo. Poklic (SKP) vnese svetovalec zaposlitve ob prvem intervjuju.*

Šifrant vrste osnov za zavarovalno dobo

Šifra	Tekst
01	Redna zaposlitev
02	Prejemanje denarnega nadomestila
03	Javno delo
04	Delo v tujini
05	Porodniški dopust ali bolniška

Podatki o prijavi in razvrstitvi v evidenco

- Datum prijave na Zavod in datum vpisa (v evidenco BO ali evidenco po drugih zakonih)

Datum prijave na Zavodu je datum, od katerega bo oseba uradno vpisana v evidenco. S tem datumom osebi pripadajo vse pravice zaradi brezposelnosti. Ta

datum se lahko vpisuje tudi za več dni (tudi mesecev) nazaj. Privzeta vrednost pa je tekoči datum.

Datum vpisa je datum, ko osebo razvrstimo v eno od evidenc (po prejemu vse dokumentacije, po rešenem postopku pritožbe...). To je vedno tekoči datum in referent tega datuma ne vidi. To je datum, ki pomaga pri analitičnem in statističnem vodenju oseb.

- Manjkajoči dokumenti
 1. osebni dokument: osebna izkaznica ali potni list
 2. delovna knjižica
 3. pogodba o zaposlitvi
 4. sklep o prenehanju delovnega razmerja
 5. dokončna odločba o odmeri davka od dohodkov iz dejavnosti v predhodnem koledarskem letu
 6. napoved za odmero dohodnine v predhodnem koledarskem letu
 7. odločba o odmeri dohodnine
 8. bilanca uspeha družbe v predhodnem koledarskem letu
 9. katastrski dohodek v preteklem letu
 10. osebno delovno dovoljenje za nedoločen čas
 11. dokončna odločba o priznani pravici do razporeditve oz. zaposlitve na drugem delovnem mestu oz. do dela s skrajšanim delovnim časom po 123. členu Zakona o pokojninskem in invalidskem zavarovanju
 12. dokončen sklep o opredelitvi trajno presežnega delavca
 13. dovoljenje za prebivanje
 14. drugo (tekstovni opis)
- Rok za dopolnitev prijave (*opozorilo, da je rok za dopolnitev potekel*)
- Identifikacijski znak svetovalca/ referenta
- Datum in ura prvega intervjuja pri svetovalcu, svetovalec in številka sobe
- Datum in ura informativnega seminarja, kraj oz. številka sobe
- Urad prijave

Na podlagi opisa vseh podatkov se je izdelal podatkovni model.

4.1.4 Izdelava podatkovnega modela

Na osnovi opisa podatkov se je pripravil podatkovni model. Podatkovni model so pripravili informatiki. Definirale so se vse tabele in šifranti. Za vsako tabelo oz. šifrant se je pripravila tabela z opisom atributov, formatom atributa, opisom atributa, določitvijo obveznega vpisa in opombami.

Tabela izobrazb oseb (IzobrazbaOsebe) v evidenci Zavoda

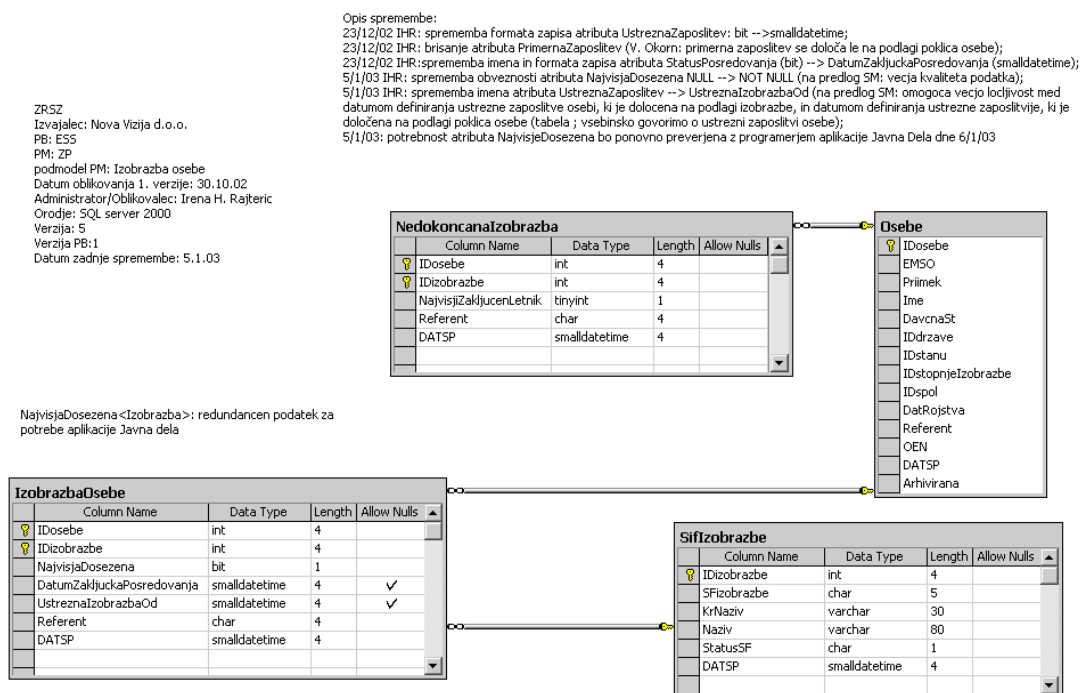
Atribut	Format	Opis	Obvezen vpis	Opombe
GUIDosebe	uniqueidentifier	GUID št. osebe	Da	PK, FK
IDizobrazbe	int	ID št. pokl./strok. izobrazbe	Da	PK, FK
NajvisjaDosezena	bit	Ali je izobrazba najvišje dosežena?	Ne	
DatumZakljucka Posredovanja	smalldatetime	Datum od kdaj se oseba v tej izobrazbi ne posreduje	Ne	
UstreznaIzobrazba Od	smalldatetime	Datum, od kdaj je ta izobrazba ustrezna	Ne	
Referent	tipReferent	Referent, ki je zadnji spremenil zapis	Da	
DATSP	smalldatetime	Datum spremembe podatka	Da	PV= getdate()

Primarni ključ: GUIDosebe
IDizobrazbe

Tuji ključi: FK_OsebeIzobrazba_Osebe
FK_OsebeIzobrazba_SifIzobrazbe

Za vse tabele in šifrante so bili izdelani relacijski modeli. Primer relacijskega modela za izobrazbo osebe:

SLIKA 17: ENTITETNO-RELACIJSKI MODEL – IZOBRAZBA OSEBE



4.1.5 Funkcionalnost aplikacije

Prva odločitev je bila, da se bo podprla tipka „Enter“ kot tipka za premik na naslednje vnosno polje. S tem smo spremenili funkcionalnost te tipke v Windows okolju. Za to smo se odločili zaradi navajenosti uporabnikov na staro DOS aplikacijo, kjer so se s tipko Enter premaknili na naslednje polje. Uporabnikov aplikacije je namreč 350, pretežno pa so to starejše osebe, ki v Windows okolju ne uporabljajo drugega kot Microsoft Outlook in Microsoft Word. Pri uporabi obeh pa uporabljajo miško, ki pa pri hitrem vnosu osebnih podatkov ne pride v poštev. Na tipko „Tab“, ki je v Windows okolju namenjena premiku na naslednje polje, uporabniki niso navajeni, vendar pa bo ta tipka to funkcionalnost ohranila. Ravno tako ohranijo svojo funkcionalnost ostale standardne Windows tipke in kombinacije tipk (Esc – prekliči, F1 – pomoč, Ctrl+C – kopiraj, Ctrl+V – prilepi itd.).

Odločili smo se, da bomo v aplikaciji podprli naslednje funkcije tipk:

Tipka	Pomen
Enter	Premik kazalca v naslednje polje
F1	Pomoč
F2	Odpiranje šifrant
F3	Iskanje osebe
F4	Zaposlitveni načrt
F5	Izpisi, dokumenti, dokumentacija
F6	Vnos zaposlitvenih ciljev (SKP)
F7	Podpisan zaposlitveni načrt
F8	Vpis podatkov v tekstovno polje (prosti tekst)
F9	Pregled podatkov o socialni varnosti
F10	
F11	
F12	
Ctrl+W	Shrani, gumb „V redu“
ESC	Ne shrani, gumb „Prekliči“

4.1.6 Ekranski vmesniki

Ekranski vmesnik je namenjen samo prikazu ter približni razvrstitvi podatkov, ki jih želimo prikazati. Izvedba samega vmesnika je prepuščena programerjem, možna so tudi večja odstopanja od same zasnove.

Primer vmesnika, ki je bil predlagan in izvedenega ekranskega vmesnika je prikazan na naslednjih slikah.

SLIKA 18: PREDLAGAN EKRANSKI VMESNIK OSNOVNIH PODATKOV O OSEBI

Osebe. | Izobrazba in izkušnje | Delovna Doba | Prijava. | Dosegljivost | Zadnja zaposlitev | Naročanje | Izposoja DK | Intervju | Intervju 1 | Zaposlitveni načrt

Osebnostni podatki:

Priimek: Davčna številka:
 Ime: Bančna številka: Vnos bančne št.
 EMSO: Pokl./strok. izobrazba:
 Dat. Rojstva: Stopnja izobrazbe:
 Spol: Telefonska številka:
 Državljanstvo: Elektronski naslov:

☐ Posredovanje da/ne

Naslovi:

Naslov bivališča	Hišna št.	Pošta	Tip Naslova	Upravna enota	Občina
Tržaška	5	1000	Ljubljana	stalni	Lj. Vič Rudnik
Podmilškakova	5	1000	Ljubljana	kontaktni	Lj. Bežigrad

Zadnja prijava:

Evidenca: Prijava od: Do: Vzrok:

Referent: Datum spremembe:

SLIKA 19: IZVEDEN EKRANSKI VMESNIK OSNOVNIH PODATKOV O OSEBI

ZP - Razvojna, Testna linija - [Prijava osebe] - Izobrazba: 57001, Svetovalec: [...]

Oseba Posredovanje Opravi Informacijski pogledi Izpisi Uredi Olna Pomoč

Urad: URAD ZA DELO CELJE

Status: Prijavljen, BO osebni dvig zaradi ZPZ

Trenutna prijava: 15.3.2005 BO

15.3.2005 Prva prijava po dokončanju / prekinitvi šolanja in daljši neaktivnosti (nad 2 leti)

Obisk: Zadnji obisk: 15.6.2005 izdaja potrdil / dokumentov

Starost, brezpos.: Starost: 26 let Trajanje brezposelnosti: 0 let 5 mesecev

Odsotnost: Prejemki: Opomba:

EMSO: Priimek: Ime: Datum rojstva:

Osnovni podatki | Prijava | Posebne skupine | Izobrazba | Delovna doba | DN/DP | Ostale evidence | Tujec | Odlombe | Opombe | Obisk osebe | Dosedanje prijave | Izdana potrdila

Oseba

Davčna številka: Spol: Državljanstvo: SLOVENIJA Star: samski (ka)

Dosedanje prijave

Nova	Evidenca	Datum vpisa	Razlog vpisa	Datum izpisa	Ra
☒	BO	15.3.2005	Prva prijava po dokončanju / prekinitvi šolanja in daljši neaktivnosti (nad 2 leti)		
☒	BO	12.11.2003	Prva prijava po dokončanju / prekinitvi šolanja	1.3.2004	por
☒	BO	2.10.2000	Prva prijava po dokončanju / prekinitvi šolanja	14.5.2001	ose

Naslovi

A.	Tip nasl.	Ulica	Telefon	Poštna številka	Kraj	Občina	Upravna enota
☒	stalni	Predstavnostna ulica		3000	CELJE	CELJE	CELJE

Dosegljivost

Telefon	Tip naslova	Ulica	Kraj	Začetna ura	Zaključna ura	E-naslov	Velja od
040 213 555	stalni	Predstavnostna ulica	CELJE	9	12		2.8.2005
040 308897	stalni	Predstavnostna ulica	CELJE	11	14		15.3.2005
	stalni	Predstavnostna ulica	CELJE	10	13		12.11.2003

Računi

Vrsta računa	Številka računa	Banka	Aktiven
transakcijski račun	6101010101010101	NLB-Nova ljublj. banka	☒

Prijava osebe

4.1.7 Javni razpis

Zavod je naročil izdelavo nove aplikacije z javnim razpisom, ki je bil objavljen v Uradnem listu. Razpis je vseboval naslednja poglavja:

1. Povabilo k oddaji ponudbe
2. Obrazec povabila k oddaji ponudbe
3. Pogoji za spremembo, dopolnitev in pojasnjevanje razpisne dokumentacije
4. Odpiranje ponudb
5. Usposobljenost in sposobnost ponudnika
6. Ponudbe v variantah
7. Navedba delov javnega naročila za katere je dopustno predlagati ponudbo
8. Zahteve glede finančnih zavarovanj
9. Zahtevek za revizijo
10. Rok veljavnosti ponudb
11. Resničnost in verodostojnost ponudb
12. Podizvajalci
13. Merila za ocenitev ponudbe
14. Obrazec ponudbe
15. Obvezni sestavni deli ponudbe (navodila o načinu dokazovanja usposobljenosti in sposobnosti ponudnikov ter ostala dokumentacija)
16. Vzorci izjav in obrazcev
 - a. Izjava, da ni bilo izdane sodbe oz. odločbe o prepovedi izvajanja dejavnosti
 - b. Izjava o poravnanih poslovnih obveznostih
 - c. Izjava o sprejemanju pogojev razpisa
 - d. Vzorec bančne garancije za resnost ponudbe
 - e. Vzorec bančne garancije za dobro izvedbo pogodbenih obveznosti
 - f. Vzorec bančne garancije za odpravo napak v garancijskem roku
 - g. Vzorec menične izjave, ki je priložena bianko menici kot finančno zavarovanje za resnost ponudbe
 - h. Vzorec menične izjave, ki je priložena bianko menici kot finančno zavarovanje za dobro izvedbo pogodbenih obveznosti
 - i. Vzorec menične izjave, ki je priložena bianko menici kot finančno zavarovanje za odpravo napak v garancijskem roku

V javnem razpisu je bilo tudi navedeno, kje in kdaj lahko zainteresirani prevzamejo razpisno dokumentacijo. V razpisni dokumentaciji so bile podrobno pojasnjene in opisane zahteve. Razpisna dokumentacija je vsebovala naslednja poglavja:

- 1 Splošni opis
 - 1.1 Opis področja dela
 - 1.2 Opis stanja na področju računalniške podpore informacijskega sistema ZRSZ in ciljev prenove računalniške podpore informacijskega sistema (RPIS)
- 2 Tehnične zahteve
 - 2.1 Arhitektura sistema
 - 2.2 Izpisi na papir

- 2.3 Spremljanje delovanja (uporabe) aplikacije
- 2.4 Navezave na integralni IS ZRSZ in uporaba skupne baze podatkov
- 2.5 Način vzdrževanja aplikacije
- 2.6 Dokumentacija
- 3 Vsebinske zahteve
- 3.1 Uvod
- 3.2 Obravnava brezposelnih oseb
- 3.3 Evidentiranje prostih delovnih mest in posredovanje na prosta delovna mesta
- 3.4 Izdelava osnovnih analitičnih in statističnih izhodov
- 4 Splošne vsebinske zahteve
- 4.1 Spremljanje dostopov do podatkov
- 4.2 Nivoji varovanja dostopov do podatkov
- 4.3 Arhiviranje podatkov (opredelitev podatkovnih sklopov, postopki arhiviranja, postopki dostopa do arhivov)
- 4.4 Opredelitev parametrov, ki bodo potrebni za spremljanje dela (svetovalcev, referentov)
- 4.5 Aktivnosti v procesu zaposlovanja
- 5 Priloge
- 5.1 Opisi podprocesov
- 5.2 Diagrami podatkovnih tokov procesa „Zaposlovanje“
- 5.3 Seznam in opis podatkov z omejitvami
- 5.4 Predloge
- 5.5 Osnutki ekranskih vmesnikov
- 5.6 Opis podatkovnega modela procesa „Zaposlovanje“
- 5.7 Definicija ustrezne in primerne zaposlitve
- 5.8 Delovna miza
- 5.9 Letno poročilo

Na razpis se je prijavilo šest podjetij. Med njimi smo izbirali najugodnejšega ponudnika po naslednjih kriterijih:

- rok izvedbe,
- garancijska doba,
- tekoči stroški (glede na opis predvidenega sodelovanja, cene odkupa izvirne kode, cene vzdrževanja in dodatnih del, zneska pavšala v 5 letih, čistega pavšala v 5 letih, cene svetovalne ure),
- kakovost (reference, prihodkov iz dejavnosti, prihodkov iz dejavnosti v državni upravi, naši oceni kadrovske usposobljenosti, števila ljudi na razvoju, števila ljudi na projektu, kadrovske usposobljenosti za projekt),
- poprodajne storitve in tehnična pomoč (cena svetovalne ure, cena programerske ure, odzivni čas, čas za odpravo napak),
- tehnične prednosti (ocena možnosti pridobitve znanja (glede na kadre), razvojna orodja, usklajenost ponujene rešitve z našim HW in SW, kompatibilnost rešitve z obstoječimi),
- estetske in funkcionalne lastnosti na podlagi referenc in opisanih rešitev in
- skupna vrednost ponudbe in cena pogarancijskega vzdrževanja.

Eno ponudbo smo izločili zaradi:

- nerealne ocene resursov (ni bila dobro preštudirana razpisna dokumentacija),
- nerealnega roka izvedbe,
- najkrajše garancijske dobe,
- opis rešitve in ponudba sta bili preveč splošni,
- ni bilo razvidnih referenc in
- kadrovska struktura za razvoj zahtevane rešitve je bila neustrezna.

Izmed vseh ponudnikov smo ponudnika, ki je bil časovno in cenovno daleč najugodnejši, izločili. Po kriterijih javnega razpisa, je imela cena 60% teže in ob upoštevanju tega kriterija ter časa izdelave, je bil ta ponudnik daleč najugodnejši in bi moral biti izbran na razpisu kot zmagovalec. Vendar bi si z izbiro le tega nakopali velike težave, saj nam nikakor ni bilo jasno, kako lahko neko podjetje s 1/2 manjšimi kadri v 1/2 časa in s 1/2 stroški izdela enako kvalitetno aplikacijo, kot naslednji najugodnejši ponudnik.

Kot najugodnejši ponudnik je bilo izbrano podjetje Nova Vizija iz Žalca. Na izbiro ponudnika smo prejeli tudi pritožbo in zahtevo po reviziji, ki pa je bila po utemeljitvi umaknjena.

4.1.8 Prenos podatkov

Zelo zahtevna naloga, ki je čakala informatike je bil prenos podatkov iz DB2 baze podatkov v SQL. Sama priprava prenosov je trajala skoraj eno leto. Po številnih prenosih in testiranjih v testno bazo je bilo ugotovljeno, da se podatki pravilno prenašajo. Šele po tej ugotovitvi, so bili podatki 31. 12. 2004 preneseni v produkcijsko bazo, da se je nova aplikacija lahko, s 3. januarjem 2005, pričela uporabljati.

4.1.9 Programiranje, testiranje, izobraževanje in zaključek projekta

Poleg opisov vseh postopkov, definicij podatkov in izdelanega podatkovnega modela se je, za pomoč pri programiranju, za vsak proces napisala še specifikacija zahtev, ki je na krajši način programerju predstavila, kaj se od aplikacije pričakuje. V nadaljevanju je prikazan skrajšan primer specifikacije zahtev za vpis naslova osebe.

Vnos in spreminjanje naslova osebe v evidencah ZRSZ

Razdelek podaja osnovne informacije o procesu vnosa in spreminjanja (administracije) naslova osebe, med katere prištevamo: tip naslova, ulica, pošta, občina, upravna enota.

Opisi in definicije

Definicija. Stalni naslov je naslov, kjer je oseba stalno prijavljena in je vpisan na osebem dokumentu.

Definicija. Začasni naslov je naslov, kjer je oseba začasno prijavljena in to dokazuje s potrdilom o začasnem bivanju.

Definicija. Kontakti naslov vpišemo, kadar se naslov bivanja razlikuje od stalnega oz. začasnega naslova.

Definicija. Naslov dosegljivosti je tisti naslov, kjer bo oseba dosegljiva za prejemanje pošte, telefonske klice... in je eden izmed zgoraj definiranih (stalni, začasni, kontaktni).

Splošna pravila

V nadaljevanju so podana nekatera splošna pravila, ki veljajo na nivoju atributne, domenske in referenčne integritete oziroma na nivoju vnosa, spremembe ali brisanja podatkov. Vsa ostala specifična pravila so zajeta neposredno pri opisu podatkovnega modela.

Pravila so naslednja:

1. pravilo. Ob spremembi naslova preverimo, če je bil prejšnji naslov definiran kot naslov dosegljivosti. Če je bil definiran, kot naslov dosegljivosti ustrezno spremenimo tudi naslov dosegljivosti.
2. pravilo. Ob ukinitvi naslova (naslov ni več aktiven) preverimo, če je bil ta naslov definiran kot naslov dosegljivosti. Če je bil definiran, kot naslov dosegljivosti, moramo izbrati nov naslov dosegljivosti.
3. pravilo. Vedno je aktiven natanko en stalni naslov. Za začasni naslov in kontaktni naslov velja, da je aktiven največ eden.

Izjema! Če je oseba tujec, ki ima samo začasno bivanje v Sloveniji, nima aktivnega stalnega naslova, ima pa obvezno aktiven začasni naslov.

4. pravilo. Naslov dosegljivosti je aktiven naslov.

Podatki za opis naslova osebe so združeni v entiteti (tabeli) OsebeNaslovi v naslednji preglednici.

Zap. št.	Opis podatka/Dodatno pravilo	Labela	Podatkovni tip/Format	Lastnosti	Naziv polja	Primer
1	Identifikator naslova osebe		Celo število - int	PK, TK	IDosebe Naslovi	7
	Podatek se na UV ne prikazuje.					
2	Oseba		Celo število - int	TK	IDosebe	9
	IdOsebe = Osebe.IdOsebe Pravilo vnosa/spremembe TK. Povezava je odvisna (<i>Dependent</i>) Izbiramo lahko le iz seznama obstoječih oseb. Pravilo brisanja/spremembe PK. Povezava je kaskadna (<i>Cascade</i>). V primeru brisanja/spremembe osebe iz tabele Osebe, pobrišemo tudi zapise v tabeli OsebeNaslovi. Podatek se na UV ne prikazuje.					
3	Vrsta naslova osebe	Vrsta naslova	Varchar(10)	NOT NULL	TipNaslova	Stalni
	Vrsta naslova: stalni, začasni, kontaktni PV= 'stalni'					
4	Ulica prebivališča	Ulica	Varchar(30)	NOT NULL	Ulica	Igriška ulica 3
5	Poštna številka	Pošta	Celo število - int	TK NOT NULL	IDposte	13
	IDposte=SifPoste.IDposte Prikaz na UV: šifra in naziv pošte					
6	Občina prebivališča	Občina	Celo število - smallint	PK,TK NOT NULL	IDobcine	17
	IDobcine=SifObcine.IDobcine Prikaz na UV: šifra in naziv občine					

Zap. št.	Opis podatka/Dodatno pravilo	Labela	Podatkovni tip/Format	Lastnosti	Naziv polja	Primer
7	Upravna enota prebivališča	Upravna enota	Celo Število - int	PK,TK NOT NULL	IDupenote	13
	IDupenote= SifUpravneEnote. IDupEnote Prikaz na UV: šifra in naziv upravne enote					
8	Oznaka aktivnosti naslova		bit	NOT NULL	Aktiven	
	Aktiven / Pasiven PV: Aktiven Aktiven zapis za posamezno vrsto naslova je zadnji veljaven naslov Prikaz oznake aktivnosti naslova na UV za vsak naslov.					

Referenčni podatki (šifranti)

Potrebni referenčni podatki pri administraciji naslovov so naslednji:

- Vrsta naslova
- Pošta
- Občina
- Upravna enota

Uporabniški vmesniki

Za potrebe administracije vnosa, popravljanja in spremljanja naslova osebe so predvideni naslednji UV:

1. Iskanje osebe
2. Vnos podatkov o naslovu osebe.
3. Administracija (spreminjanje) obstoječih podatkov o naslovu osebe.

Informacijski pogledi

Informacijski pogled predstavljajo vsi želeni in pričakovani pogledi v podatke o naslovu osebe na ZRSZ, katerih namen je predvsem pregled, lahko pa tudi sprememba. V nadaljevanju so naštet nekateri od potrebnih informacijskih pogledov.

1. Pregled naslovov oseb v izbrani pošti, občini ali upravni enoti, UD, OS.
2. Seznam aktivnih naslovov osebe.
3. Pregled naslovov osebe (pasivni, aktivni).
4. Pregled naslovov o dosegljivosti osebe.

Funkcije in procesi

Preglednica funkcij ali procesov podaja seznam, opis in prioriteto tistih funkcij IS, ki jih mora le-ta zadovoljiti. Implementacija funkcij je predmet načrta in implementacije programske opreme.

Naziv funkcije/procesa	Kratek opis	Prioritetni nivo*
Vnos naslova	Funkcija je namenjena vnosu natančno enega zapisa, ki nam pove naslov osebe.	9
Administracija naslova	Funkcija je namenjena administraciji sprememb podatkov o naslovu osebe.	7
Administracija aktivnosti naslova	Funkcija je namenjena administraciji sprememb podatka o aktivnosti izbranega zapisa.	9
Kontrola naslovov	Preverjanje skladnosti naslovov.	4
Ukinitev/brisanje naslovov	Kontrolirano brisanje naslova osebe zaradi kakršnega koli razloga (predvsem zaradi napake vnašalca)	6
Sprememba občine bivališča	Funkcija je namenjena administraciji podatka o občini stalnega prebivališča.	6
Sprememba upravne enote	Opomba. Funkcija bi morala biti na nivoju FS sprememb upravnih enot. (Funkcija se nanaša na problematiko prehoda določenega kraja iz ene v drugo upravno enoto.)	6

* Prioritetni nivo (1 - najnižji; 10 - najvišji)

Programiranje

Na osnovi razpisne dokumentacije in funkcijskih specifikacij je zunanji izvajalec pričel s programiranjem aplikacije. Dogovorjeno je bilo, da se v testiranje dobijo verzije po posameznih sklopih (prijava osebe, prvi intervju, zaposlitveni načrt...), ki se testirajo in javljajo napake. Prejeti popravki se ponovno testirajo do verzije, ki izpolnjuje vse zahteve. V času programiranja se z izvajalci tesno sodeluje in glede na napredovanje tudi prilagaja oz. spreminja zahteve. Predloge za izboljšanje smo velikokrat prejeli tudi od izvajalcev, ki so videli boljše rešitve, kot smo jih predlagali sami. Čas programiranja se je zaradi tega tudi podaljšal, vendar je to bilo na račun boljše kvalitete aplikacije.

Testiranje

V testiranje smo nove verzije aplikacije dobivali po sklopih. Najprej smo v testiranje dobili postopek prijave brezposelne osebe v evidenco, nato postopek izvajanja prvega intervjuja itd. S testiranjem smo preverjali, če so vse zahtevane

funkcionalnosti vključene v program in če delujejo pravilno oz. tako kot smo pričakovali. Napake so bile takoj javljene in z novimi verzijami odpravljene.

Izobraževanje

Pripravili smo tudi usposabljanja za vse uporabnike (približno 500 uporabnikov), zanje pa smo napisali tudi priročnik. Čeprav je bilo dogovorjeno, da priročnik pripravi izvajalec, smo ga vseeno pripravili sami, saj vanj niso bila vključena samo navodila za uporabo aplikacije temveč tudi vsebinska pojasnila. Poleg tega bi samo pisanje priročnika izvajalcem vzelo preveč časa.

Zaključek projekta

Aplikacija je v uporabi od 3. januarja 2005, ko se je projekt tudi uradno zaključil. Vendar pa aplikacija zaradi zahtevnosti in obsežnosti v celoti še ne zadovoljuje nas kot naročnika. Postopoma, se tudi te pomanjkljivosti še vedno odpravljajo. Sama aplikacija sicer izpolnjuje vse zahteve razpisa, vendar pa je potrebno določene funkcionalnosti dopolniti oz. optimizirati. Po zaključku projekta je vsebinsko skrbništvo nad aplikacijo prevzela služba za zaposlovanje, glavno skrbništvo pa informatika. S podpisom vzdrževalne pogodbe se aplikacija še naprej izpopolnjuje. Prišlo je tudi do spremembe zakonodaje, kar je pomenilo precejšnjo spremembo aplikacije. Zaradi dobre strukture aplikacije in dobre predpriprave pri tem ni bilo posebnih težav.

4.2 Projekt izdelave aplikacije APZ_Net

Projekt se je pričel izvajati januarja 2005 s sklepom vodstva Zavoda. V obdobju od januarja do aprila 2005 je bilo izvedeno planiranje projekta in priprava Projektnega načrta.

Projektne načrt je bil sprejet na seji Usmerjevalne skupine dne 13. 04. 2005. V aprilu 2005 se je pričel izvedbeni del projekta.

V času od aprila do junija sta delo na projektu opravljala vodja projekta Albin Kerec kot vodja in članica Sabrina Urnaut, pri določenih aktivnostih pa so sodelovali še člani projekta Irena Hribar Rajterič, Brane Kne, Branka Kužnik in Tomaž Bernik in naslednji člani podprojektne skupine za pripravo modulov: Damjana Košir, Meta Cerar, Metka Barbo Škerbinc, Ana Marija Ljubec, Vesna Cvilak, Robert Modrijan, Helena Dolinšek in Dorotea Verša.

Od 01. 07. 2005 dalje delo na projektu opravljajo vodja projekta Albin Kerec in članici: Irena Hribar Rajterič in Sabrina Urnaut.

Stroški projekta, ki so nastali v začetni fazi so: materialni stroški izvedenih dveh delavnic, stroški dela projektne skupine in stroški članov podprojektne skupine in ostalih sodelavcev, ki so sodelovali na projektu.

4.2.1 Priprava projektne dokumentacije s terminskim planom

Vodja delovne skupine je pripravil terminski plan. Plan s korekcijami, ki so se izvedle med projektom ima naslednjo vsebino in obliko:

Št.	Aktivnost	Začetek	Konec	Nosilci
1	Projekt APZnet	1.6.05	30.6.07	Albin Kerec
2	Izdelava modula Izbor izvajalca	1.6.05	31.5.06	Projektna skupina
3	Postopkovnik za modul Izbor izvajalca	1.6.05	2.6.05	Projektna skupina
4	Izvedba 1. delavnice - določitev procesa	1.6.05	2.6.05	Projektna skupina
5	Izdelava funkcijskih specifikacij (specifikacij zahtev za programsko opremo)	16.6.05	22.9.05	Projektna skupina
6	Izvedba 2. delavnice - usposabljanje in pričetek izdelava funkcijskih specifikacij	16.6.05	17.6.05	Projektna skupina
7	Dopolnjevanje funkcijskih specifikacij	1.9.05	22.9.05	Projektna skupina
8	Izdelava podatkovnega modela	4.8.05	22.9.05	Projektna skupina; Rozi Zorzut
9	Izdelava osnutka podatkovnega modela	4.8.05	22.9.05	Projektna skupina
10	Priprava vsebine šifrantov	19.9.05	22.9.05	Rozi Zorzut
11	Dokončan modul Izbor izvajalcev	23.9.05	23.9.05	Projektna skupina
12	Izobraževanje programerske skupine	4.10.05	30.11.05	Matej Nosan
13	Izobraževanje 1. sklop	4.10.05	6.10.05	Matej Nosan
14	Izobraževanje 2. sklop	11.10.05	13.10.05	Matej Nosan
15	Priprava učilnice v MB	4.10.05	13.10.05	Matej Nosan
16	Izobraževanje 3. sklop	18.10.05	20.10.05	Matej Nosan
17	Izobraževanje 4. sklop	25.10.05	27.10.05	Matej Nosan
18	Zagotovitev prostorov v MB	4.10.05	28.10.05	Matej Nosan
19	Zagotovitev opreme	4.10.05	28.10.05	Matej Nosan
20	Priprava prostorov in opreme za delo	31.10.05	7.11.05	Matej Nosan

21	Zagotovitev specifikacij učnega primera s strani ZRSZ	4.10.05	27.10.05	Projektna skupina
22	Izobraževanje 5 sklop, delo pod mentorstvom (Programiranje 1. teden)	8.11.05	10.11.05	Projektna skupina
23	Postavitev strežnika CSV	8.11.05	8.11.05	Matej Nosan
24	Programiranje dela specifikacij prvega modula, način izbora izvajalcev	8.11.05	10.11.05	Matej Nosan
25	Revizija izobraževanja s skupino in definicija morebitnih potreb za nadaljnje delo	15.11.05	17.11.05	Matej Nosan
26	Programiranje 1. teden	25.11.05	30.11.05	Matej Nosan
27	Dokončanje izobraževanja (zamujen četrtek 25.11.2005)	25.11.05	25.11.05	Matej Nosan
28	Priprava poročila o zaključku izobraževanja	28.11.05	30.11.05	Matej Nosan
29	Preverjanje funkcijskih specifikacij in podatkovnega modela pri končnih uporabnikih	24.10.05	17.11.05	Projektna skupina
30	Dopolnjevanje funkcijskih specifikacij	24.10.05	28.10.05	Projektna skupina
31	Kreiranje šifrantov	31.10.05	11.11.05	Rozi Zorzut
32	Dopolnjevanje podatkovnega modela	14.11.05	17.11.05	Projektna skupina
33	Potrjena dokumentacija za modul Izbor izvajalcev s strani uporabnikov	18.11.05	18.11.05	vodja APZ; pravna služba
34	Programiranje modula Izbor izvajalca	2.11.05	30.4.06	Matej Nosan
35	Priprava razvojnega okolja (podatkovni strežnik)	2.11.05	4.11.05	Projektna skupina
36	Prenos tabel modula Izbor Izvajalca na razvojni strežnik CSW2SQLT1 v bazo ESSRazvoj	2.11.05	3.11.05	Projektna skupina
37	Priprava kopije baze za vzpostavitev razvojnega okolja programerske skupine	4.11.05	4.11.05	Projektna skupina
38	Programiranje 2. teden	15.11.05	17.11.05	Matej Nosan
39	Programiranje dela specifikacij prvega modula, način izbora izvajalcev	15.11.05	17.11.05	Matej Nosan
40	Definiranje potreb za nadaljnje delo (nakup potrebnih	15.11.05	17.11.05	Matej Nosan

	programskih komponent)			
41	Programiranje 3. teden: programiranje dela specifikacij prvega modula, način izbora izvajalcev	22.11.05	24.11.05	Matej Nosan
42	Programiranje 4. teden: programiranje dela specifikacij prvega modula, način izbora izvajalcev	29.11.05	30.4.06	Programerska skupina
43	Programiranje dela modula Izbor izvajalca po specifikacijah Vnos vloge	1.4.06	30.4.06	Programerska skupina
44	Testiranje modula	1.5.06	31.5.06	Projektna skupina
45	Priprava testnega okolja	1.5.06	4.5.06	Projektna skupina
46	Testiranje modula	5.5.06	31.5.06	Projektna skupina
47	Usposabljanje uporabnikov	8.5.06	15.5.06	Projektna skupina
48	Usposabljanje testnih uporabnikov	8.5.06	15.5.06	Projektna skupina
49	Priprava priročnika za usposabljanje	8.5.06	15.5.06	Projektna skupina
50	Izvajanje usposabljanj za uporabnike	10.5.06	15.5.06	Projektna skupina
51	Priprava produkcijskega okolja	10.5.06	15.5.06	Matej Nosan; Viktorija Komavec
52	Operativna uporaba modula Izbor izvajalca	15.5.06	31.5.06	Usmerjevalna skupina
53	Izdelava modula Pogodbe	26.9.05	30.9.06	Projektna skupina
54	Postopkovnik za modul Pogodbe	26.9.05	4.11.05	Projektna skupina
55	Določitev nabora podatkov v pogodbah UAPA	26.9.05	30.9.05	Projektna skupina
56	Izvedba 1. delavnice - Opis procesa pogodbe	3.10.05	11.10.05	Projektna skupina
57	Priprava gradiva za 1. delavnico	3.10.05	4.10.05	Projektna skupina
58	Izvedba 1. delavnice – določitev procesa za izdajo pogodb za UAPA	5.10.05	6.10.05	Projektna skupina
59	Oblikovanje dokumentacije po 1. delavnici	7.10.05	11.10.05	Projektna skupina
60	Analiza obstoječih pogodb in aneksov (splošni členi, vsebinski členi)	11.10.05	14.10.05	Projektna skupina
61	Izvedba 2. delavnice - vnos in sledenje pogodb	17.10.05	28.10.05	Projektna skupina
62	Priprava gradiva za 2. delavnico	17.10.05	18.10.05	Projektna skupina

	(vnos in sledenje pogodb in prilog/poročil/aneksov)			
63	Izvedba 2. delavnice - model za vnos in sledenje pogodb	19.10.05	20.10.05	Projektna skupina
64	Oblikovanje dokumentacije po 2. delavnici	21.10.05	28.10.05	Projektna skupina
65	Določitev tabele stroškov in parametrov (za vnos v pogodbe)	2.11.05	4.11.05	Projektna skupina
66	Določitev izračuna vrednosti pogodb (priprava formul)	2.11.05	4.11.05	Projektna skupina
67	Priprava predlogov pogodb in aneksov za vse programe – poenotenje tekstov	2.12.05	31.12.05	Projektna skupina
68	Izdelava funkcijskih specifikacij	14.11.05	30.4.06	Projektna skupina
69	Priprava funkcijskih specifikacij	14.11.05	16.11.05	Projektna skupina
70	Dopolnjevanje funkcijskih specifikacij	23.11.05	30.4.06	Projektna skupina
71	Izdelava podatkovnega modela	13.4.06	30.4.06	Projektna skupina, Rozi Zorzut
72	Izdelava osnutka podatkovnega modela	13.4.06	30.4.06	Projektna skupina
73	Priprava vsebine šifrantov	18.4.06	30.4.06	Rozi Zorzut
74	Potrjena dokumentacija za modul Pogodbe s strani uporabnikov	30.4.06	30.4.06	vodja APZ; pravna služba
75	Programiranje modula Pogodba	1.5.06	30.6.06	Programerska skupina
76	Testiranje modula	1.7.06	31.7.06	Projektna skupina
77	Priprava testnega okolja	1.7.06	7.7.06	Projektna skupina
78	Testiranje modula	1.7.06	31.7.06	Projektna skupina
79	Usposabljanje uporabnikov	10.7.06	30.9.06	Projektna skupina
80	Priprava priročnika za usposabljanje	10.7.06	31.7.06	Projektna skupina
81	Izvajanje usposabljanj za uporabnike	15.8.06	30.9.06	Projektna skupina
82	Izdelava modula Finance	1.4.06	31.10.06	Projektna skupina
83	Postopkovnik za modul Finance	1.4.06	30.4.06	Projektna skupina; služba za finance
84	Izdelava funkcijskih specifikacij	11.4.06	30.6.06	Projektna skupina; služba za finance
85	Izdelava podatkovnega modela	1.6.06	30.6.06	Projektna skupina
86	Programiranje modula Finance	3.7.06	30.9.06	Programerska skupina; skrbnik fin. rač. aplikacije

87	Testiranje modula Finance	1.10.06	31.10.06	Projektna skupina; služba za finance
88	Usposabljanje uporabnikov modul Finance	1.10.06	31.10.06	Projektna skupina
89	Testiranje osnovnih, nujno potrebnih funkcionalnosti (vsi moduli)	1.11.06	31.12.06	Projektna skupina
90	Usposabljanje uporabnikov po posameznih modulih	1.11.06	31.12.06	Projektna skupina; vodja APZ
91	Izgradnja podatkovnega skladišča za potrebe zbirnega modula	1.3.06	31.12.06	Alenka M. Vidmar; Albin Kerec; vodja APZ; vodja analitike
92	Izdelava zbirnega modula	1.1.07	31.5.07	Albin Kerec; vodja APZ; vodja analitike
93	Postopkovnik za zbirni modul	1.1.07	31.1.07	Projektna skupina
94	Izdelava funkcijskih specifikacij	1.1.07	31.1.07	Projektna skupina
95	Programiranje zbirnega modula	1.2.07	30.4.07	Programerska skupina
96	Testiranje modula	1.5.07	31.5.07	Projektna skupina
97	Testiranje celotne aplikacije z uporabniki	1.5.07	31.5.07	Projektna skupina
98	Usposabljanje uporabnikov za celotno aplikacijo	1.5.07	31.5.07	Projektna skupina
99	Poročanje projektne skupine (1x mesečno)	1.9.05	30.6.07	Projektna skupina
100	Operativna uporaba aplikacije	1.7.06	1.1.07	Usmerjevalna skupina
101	Operativna uporaba aplikacije (osnovne, nujno potrebne funkcionalnosti)	1.1.07	1.1.07	Usmerjevalna skupina
102	Operativna uporaba celotne aplikacije	1.7.07	1.7.07	Usmerjevalna skupina

4.2.2 Analiza obstoječega stanja in izdelava postopkovnega modela

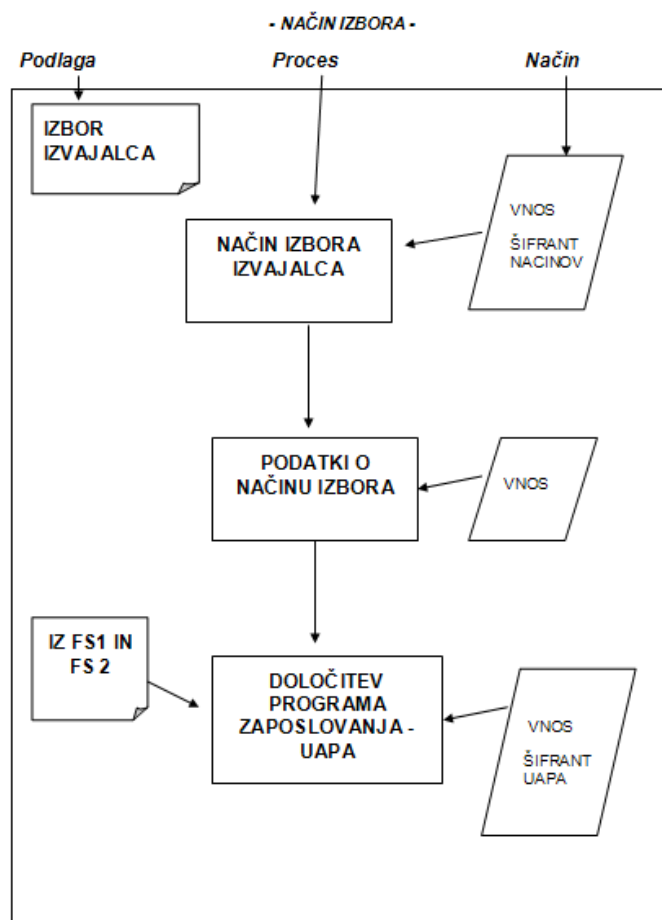
Tudi v primeru APZ-ja je programska oprema že obstajala, vendar so se člani projektne skupine odločili, da se analiza obstoječe programske opreme ne bo izvajala, temveč se bo začelo z analizo procesov in postopkov dela.

Postopki dela so se preučevali na različnih območnih službah. Pri pripravi postopkovnega modela so se upoštevali najboljše postopki.

Ukrepi, aktivnosti in podaktivnosti aktivne politike zaposlovanja (v nadaljevanju UAPA) se praviloma izvajajo tako, da je na eni strani oseba, ki se vključuje v UAPA, na drugi pa izvajalec, ki UAPA izvaja.

Zavod izvajalce za večino UAPA izbira sam in to na različne načine, ki so predpisani z zakoni, pravilniki in internimi navodili. Za UAPA so načini izbora vnaprej določeni, zato mora obstajati povezava med šifrantom UAPA in šifrantom načinov izbora izvajalcev. Potek procesa izbora izvajalca prikazuje spodnja slika.

SLIKA 20: GROBA SKICA POSTOPKOVNEGA MODELA ZA IZBOR IZVAJALCA



Za posamezen UAPA je najprej potrebno določiti način izbora. Glede na način izbora je potrebno opredeliti vse potrebne podatke o tem načinu. Gre za opredeljevanje formalnih podatkov o načinu izbora in o dokumentaciji, ki se zahteva od vlagateljev.

Izvajalci, ki se izbirajo so lahko:

- **izvajalci programov** (npr. program javnih del, klubi za iskanje zaposlitve),
- **koristniki programov** – delodajalci (npr. sofinanciranje izobraževanja in usposabljanja zaposlenih, delovni preizkus, spodbujanje zaposlovanje težje zaposljivih oseb) in
- **posamezne osebe**, ki kot vlagatelji za program postanejo koristniki sredstev (samozaposlovanje).

ZRSZ izvajalce lahko izbere na več načinov:

- z objavo javnih razpisov,
- z zbiranjem ponudb – naročila male vrednosti,
- z javnim pozivom,
- na podlagi individualne vloge ali
- sklepa pogodbe z izvajalci, ki so jih na javnem razpisu izbrale druge institucije (npr. ministrstvo za šolstvo, PCMG).

Javni razpis objavi ZRSZ v Uradnem listu in določi strokovno komisijo, ki bo vloge prejete na razpis pregledala in obravnavala. Najprej se pregleda ali je kuverta pravilno označena in pravočasna. V kolikor vloga ni pravočasna se obravnava na naslednji seji. Če ni več rokov se neodprta vrne predlagatelju. V drugi fazi se pregleda iz vidika popolnosti – ali vloga vsebuje vse predpisane elemente in priloge, če jih ne vsebuje komisija vlagatelja pozove k dopolnitvi vloge. Ko je vloga popolna jo strokovno obravnava in predlaga ali se izbere vlagatelja ali ne. O tem izda sklep, ki ga podpiše generalni direktor. Če vloga ni popolna ali dopolnjena v roku ali je nepravilno označena se izda sklep o zavržbi. Če se vlagatelj pritoži, komisija obravnava tudi pritožbo in ponovno izda sklep o predlagani odločitvi. Javni razpis se lahko izvede po:

- Zakonu o izvrševanju proračuna (ZIPRO) ali
- Zakonu o javnih naročilih (ZJN), kjer so možni različni načini in sicer:
 - omejen razpis,
 - odprt razpis in
 - razpis s pogajanji.

Pri omejenem postopku je potrebno ločiti prvo in drugo fazo postopka. V 1. fazi se za celotno obdobje izbora ugotavlja sposobnost izvajalcev, v drugi fazi pa gre za izbor tistih izvajalcev s katerimi se dejansko sklene pogodba za izvajanje programov. Ta izbor se vrši za vsako leto posebej.

Zbiranje ponudb v skladu z zakonom o javnih naročilih - za male vrednosti – delavec Zavoda pozove izvajalce da predložijo ponudbe za izvajanje

določenega programa in izmed prejetih izbere ekonomsko najustreznejšo ponudbo.

Pri **javnem pozivu** se obravnava ponudbe na enak način kot pri naročilih male vrednosti, le objava poziva je javna.

Kadar gre za **sezname že znanih izvajalcev** ni obravnave vlog, ampak se z njimi le sklepajo pogodbe.

Obstaja tudi ena izjema. To je primer **individualnih vlog**, kjer vlogo poda brezposelna oseba, način obravnave pa je enak kot pri javnih razpisih, le da ni potrebno izdajati sklepov, ampak se sklepa pogodba.

4.2.3 Definicija potrebnih podatkov

Načini izbora so:

1. **Javni razpis PIP** – postopek izbora je določen v skladu s Pravilnikom o izvrševanju proračuna. Ta vrsta načina izbora izvajalca se deli glede na različne pravne podlage za izvajanje in na to od kje se financira in sicer:

1.1. **Javni razpis PIP IP** postopek izbora je določen v skladu s Pravilnikom o izvrševanju proračuna (sredstva integralnega proračuna RS).

1.2. **Javni razpis PIP ESS** postopek izbora je določen v skladu z Uredbo o izvajanju postopkov pri porabi sredstev strukturne politike (sredstva ESS) v RS (Uredbo) in Pravilnikom o izvrševanju proračuna (PIP).

Uporabnik izbere 1.1 ali 1.2, ne more izbrati 1. Nadrejena šifra se uporablja le pri prikazu podatkov.

Opomba: Javni razpis PIP je ločen na dva dela zaradi različnih pravnih podlag in postopkov, ki bodo podrobneje opisani pri vnosu vloge. Uredba v 13. členu ureja podrobneje izbor predlogov projektov in določa npr. potrjevanje predloga liste izbranih projektov s strani programskega sveta in drugo. Iz tega razloga se loči postopke po PIP na ESS in IP. Uredba je vezana na postopke po PIP in ni predvidena njena uporaba v povezavi z ZJN.

2. **Javni razpis ZJN** - postopek izbora je določen v skladu z Zakonom o javnih naročilih. Pri tem načinu izbora je mogoče koriščenje sredstev iz integralnega proračuna, prav tako pa tudi iz ESS. Razdelitev na dva načina izbora - na IP in ESS v tem primeru ni smiselna (postopki so enaki), razlika je v obveznosti vnosa podatka referenčne številke EU ter pri vsebini izpisov.

Ta vrsta načina izbora se nadalje deli glede na načina izvajanja postopka za izbor izvajalca.

2.1. **ZJN – omejen razpis** - ima dve fazi:

2.1.1. **ZJN – omejen razpis – I. faza** - ugotavljamo sposobnost ali usposobljenost izvajalca.

2.1.2. **ZJN – omejen razpis – II. faza** – za izbrane izvajalce v I. fazi zbiramo ponudbe.

Izbor te vrste načina pri vnosu podatkov o načinu izbora je možen samo v primeru, da je predhodno vnesena prva faza in da so že izbrani izvajalci – izdane odločitve o usposobljenosti.

2.2. **ZJN – odprt razpis**

Odprt razpis oddaje javnega naročila je razpis, pri katerem lahko vsi, ki imajo interes pridobiti javno naročilo, predložijo svoje ponudbe, pripravljene skladno z vnaprej določenimi zahtevami naročnika, določenimi v razpisni dokumentaciji.

2.3. **ZJN – postopek s pogajanjem** (z ali brez predhodne objave) kadar gre za omejeno število znanih izvajalcev; na razpisu ni bilo prijavljenih izvajalcev za storitev, ki se naroča. Z njimi se pogaja za ceno in pogoje.

3. **JNMV – javna naročila male vrednosti** podlaga je Navodilo o oddaji javnih naročil male vrednosti N 9931 04 001. Zavod mora pridobiti najmanj 2 ponudbi in izbrati najugodnejšega ponudnika.
4. **Javni poziv** - poziv izvajalcem za zbiranje ponudb, ki je lahko objavljen v Ur. l. ali v medijih (tiskanih, internet), gre za izbiro ekonomsko najugodnejših ponudb oz. izvajalcev.
5. **Individualna vloga** - oseba sama poda vlogo na obrazcu v skladu z internimi navodili Zavoda.
6. **Seznam** – seznam izbranih izvajalcev pri drugih institucijah – Zavod samo vnese že potrjene izvajalce.

Podrobnejši opis potrebnih podatkov pri vnosu načina izbora:

- šifra UAPA,
- naziv načina izbora - razpisa/poziva (kako se imenuje razpis/poziv),
- delovodna številka razpisa/poziva,
- klasifikator za knjiženje vlog,
- odgovorna oseba za razpis,
- vir objave (uradni list, dnevni časopisi, tv, internet...),
- številka objave – gre za številko iz vira objave,
- referenčna številka – številka iz Uradnega lista EU – za razpise ESS, ki jih je potrebno objavljati tudi v Uradnem listu EU,
- datum objave - kdaj je bil objavljen,
- šifra objave (samo za objave v Ur. l.),

- datum odprtja - datum, od katerega dalje vlagatelji vlagajo prijave na razpis,
- datum zaprtja, datum do katerega lahko vlagatelji vlagajo prijave na razpis,
- obdobje veljavnosti izbora od do (obdobje, za katerega Zavod izbere izvajalca),
- OEN (organizacijske enote), ki lahko na tem načinu izbora delajo – s tem mislimo CS in 12 OS. Vsaki OEN se določi vrednost razpisa za to OEN (vpiše se, če je določena, ni obvezen podatek - gre za višino sredstev, ki je na razpolago za javni razpis),
- skupna vrednost razpisa (vpiše se, če je določena, ni obvezen podatek - gre za višino sredstev, ki je na razpolago za javni razpis za vse OEN skupaj),
- rok za dopolnitev vlog (potrebuje se pri izpisih, če se ne vnese je privzeta vrednost 8 dni, možno popraviti) in
- šifra zahtevane dokumentacije.

4.2.4 Izdelava podatkovnega modela

Podatkovni model APZ je nastajal v času prenove informacijske podpore za področje programov zaposlovanja z namenom njegove enotne uporabe v okviru vseh aplikacij, ki potrebujejo podatke pridobljene v procesu izvajanja programov zaposlovanja.

Dogovorjeno je bilo naslednje:

- podatkovni model sestavljajo tabele, ki podpirajo izvajanje procesa programov zaposlovanja,
- vse aplikacije ZRSZ morajo imeti vgrajeno enako logiko uporabe,
- nadaljnje spremembe se vršijo v dogovoru z vsemi, ki podatkovni model uporabljajo,
- dodajanje tabel in spreminjanje strukture tabel je v pristojnosti administratorja PM APZ, administratorja baze in strežnika in
- podatkovni model se navezuje na obstoječe tabele podatkovnih modelov Osebe, Poslovni subjekti, Zaposlovanje, IAS, referenčne podatke.

V nadaljevanju sledi opis tabel, ki gradijo PM za izbor izvajalcev UAPA. Podrobneje je predstavljena njihova struktura in namen uporabe.

Vsaka tabela je opisana z relacijsko shemo in sicer po pravilih:

- v relaciji (tabeli) so v oklepaju navedeni vsi atributi relacije (tabele)
- atributi, ki predstavljajo primarne ključe, so označeni z znakom '#', tuji ključi so podčrtani.

Splošna pravila, ki veljajo za vse tabele:

- NULL vrednost v atributih ni dovoljena, razen tam, kjer je eksplicitno navedeno.

- Privzete vrednosti za posamezne attribute so navedene pri opisu posamezne tabele.
- Atribut **Referent** se polni z vrednostjo IDuporabnika tabele ZZUporabnikiAplikacij (matična številka delavca).
- Atribut **DatSp** ima privzeto vrednost **getdate()**.
- Indeksi se dodajajo/spreminjajo izključno z dovoljenjem administratorja baze.

Pravila, ki veljajo za posamezno tabelo, so opisana pri opisu tabele v nadaljevanju dokumenta in so trenutno veljavna v okviru aplikacije APZnet.

V nadaljevanju so opisane tabele, ki sestavljajo podatkovni model APZ. Njegov dom je na SQL strežniku *CSW3SQLZPbaza ESS*, ki je tudi edino mesto za ažuriranje podatkovnega modela in ključnih podatkov. Vse spremembe se pred uvedbo predhodno obvezno preverijo na testnem SQL strežniku *CSW2SQL1*, baza *ESSTestna* in se implementirajo po potrditvi delovanja skrbnikov vseh aplikacij, ki omenjeni PM uporabljajo.

▪ **Tabela APZIzborIzvajalcaNacin**

Tabela splošnih podatkov in pogojev, ki opredeljujejo način izbora izvajalcev (upoštevajo se pri obravnavi vlog za izbor izvajalcev).

Relacijska shema:

APZIzborIzvajalcaNacin (#IDizboraIzv, #IDnacIzbora, NazivNacIzbora, VirObjave, StObjave, DatObjave, SFObjave, ReferencaEU, DatOdprtjaOd, DatOdprtjaDo, DatVeljIzboraOd, DatVeljIzboraDo, RokZaDopolnitevVlog, VrednostNacIzbora, Referent, DatSp)

Naziv atributa	Podatkovni tip	Obvezen vpis	Opombe
IDizboraIzv	int	NOT NULL	Identifikator načina izbora izvajalca
IDnacIzbora	tinyint	NOT NULL	Identifikator vrste načina izbora izvajalca
NazivNacIzbora	varchar(500)	NOT NULL	Uradni naziv izbora izvajalca (razpis, poziv, ponudba)
VirObjave	varchar(100)	NULL	Vir objave izbora izvajalca npr. Ur. I.
StObjave	varchar(50)	NULL	Številka objave dodeljena ob objavi načina izbora izvajalca (npr. iz Ur. I.)
DatObjave	smalldatetime	NULL	Datum objave načina izbora izvajalca (razpis, poziv, ponudba)
SFObjave	varchar(10)	NULL	Šifra objave načina izbora izvajalca (razpis, poziv, ponudba)
ReferencnaStEU	varchar(100)	NULL	Referenčna številka dokumenta

Naziv atributa	Podatkovni tip	Obvezen vpis	Opombe
			EU na katerega se sklicuje izbrani način izbora izvajalca (praviloma za razpise ESS, ki se objavljajo v Ur. l. EU)
DatOdprtjaOd	smalldatetime	NULL	Datum od vključno katerega dne dalje lahko vlagatelji vlagajo prijave na izbrani način izbora izvajalca (razpis, poziv, ponudba)
DatOdprtjaDo	smalldatetime	NULL	Datum do vključno katerega dne dalje lahko vlagatelji vlagajo prijave na izbrani način izbora izvajalca (razpis, poziv, ponudba)
DatVeljIzboraOd	smalldatetime	NULL	Datum veljavnosti izbora od vključno katerega lahko Zavod izbere izvajalca
DatVeljIzboraDo	smalldatetime	NULL	Datum veljavnosti izbora do vključno katerega lahko Zavod izbere izvajalca
RokZaDopolnitevVlog	tinyint	NOT NULL	Parameter, ki določa število dni v katerih mora vlagatelj dopolniti poslano vlogo (velja za vse vloge, ki prispejo na izbrani način izbora izvajalca); privzeta vrednost: 8 dni
VrednostNacIzbora	real	NULL	Skupna vrednost načina izbora izvajalca (vpiše se, če je določena)
Referent	char(4)	NOT NULL	Uporabnik, ki je zapis administriral (IDuporabnika tabele ZZUporabnikiAplikacij)
DATSP	smalldatetime	NOT NULL	Datum vnosa/spremembe zapisa

Relacije med tabelami:

SifNacinIzbora : APZizborIzvajalcaNacin 1:M

APZizborIzvajalcaNacin : APZizborIzvajalcaZaUAPA 1:M

APZizborIzvajalcaNacin : APZizborIzvajalcaDokPredvidena 1:M

Indeksi:

PK_ APZizborIzvajalcaNacin (clustered index nad IDizboraIzv)

Pravila:

- Vrsta načina izbora izvajalca nam določa nabor podatkov, ki jih je potrebno vpisati.
- Pri vrstah načina izbora izvajalca ločimo več nivojev izbora izvajalcev. Primer: Vrsta načina izbora izvajalca ZJN se deli na 3 podnivoje: omejen, odprt, razpis s pogajanjem, pri čemer ima „Omejen“ 2 fazi (3 nivo).
- Če ni 1. faze omejenega ZJN se postopek izbire izvajalca glede na 2. fazo omejenega ZJN ne sme začeti.
- ZJN - 2. faza se kot način izbora pri vnosu podatkov o načinu izbora ne sme izbrati (le izbor izmed izvajalcev, ki so dobili pozitivni sklep o sposobnosti in usposobljenosti izvajalca).

▪ **Tabela APZIZborIzvajalcaDokPredvidena**

Tabela dokumentov, ki tvorijo predvideno dokumentacijo za predhodno opredeljen način izbora izvajalca npr. razpis za izbor izvajalcev za javna dela (upoštevajo se pri obravnavi vlog za izbor izvajalcev).

Relacijska shema:

APZIZborIzvajalcaDokPredvidena (# *IDizboraIzv*, # *IDdokumenta*, Referent, DatSp)

Naziv atributa	Podatkovni tip	Obvezen vpis	Opombe
IDizboraIzv	int	NOT NULL	Identifikator načina izbora izvajalca
IDdokumenta	smallint	NOT NULL	Identifikator zahtevane dokumentacije (vrste dokumentov, ki jih je potrebno zahtevati pri izbranem načinu izbora izvajalca)
Referent	char(4)	NOT NULL	Uporabnik, ki je zapis administriral (IDuporabnika tabele ZZUporabnikiAplikacij)
DATSP	smalldatetime	NOT NULL	Datum vnosa/spreembe zapisa

Relacije med tabelami:

SifZahtevanaDokumentacija : APZIZborIzvajalcaDokPredvidena 1:M

APZIZborIzvajalcaNacin : APZIZborIzvajalcaDokPredvidena 1:M

Indeksi:

PK_ APZIZborIzvajalcaDokPredvidena (clustered index nad IDizboraIzv, IDdokumenta)

Pravila:

- Šifrant zahtevane dokumentacije se dopolnjuje ob vnosu načina izbora izvajalca (v aplikaciji); dopolnjevanje izvaja ena ali več oseb za celotni Zavod (*pravice so omejene*).

▪ **Tabela APZIZborIzvajalcaZaUAPA**

Tabela **U**krepev, **A**ktivnosti in **PodA**ktivnosti za katere je definiran način izbora izvajalca. UAPA je lahko definiran na različnih nivojih; nivojska struktura je razvidna iz šifranta UAPA.

Relacijska shema:

APZIZborIzvajalcaZaUAPA (# IDizboraIzv, # IDUAPA, Referent, DatSp)

Naziv atributa	Podatkovni tip	Obvezen vpis	Opombe
IDizboraIzv	int	NOT NULL	Identifikator načina izbora izvajalca
IDUAPA	smallint	NOT NULL	Identifikator nivojev in podnivojev UAPA
Referent	char(4)	NOT NULL	Uporabnik, ki je zapis administriral (IDuporabnika tabele ZZUporabnikiAplikacij)
DATSP	smalldatetime	NOT NULL	Datum vnosa/spremembe zapisa

Relacije med tabelami:

SifUAPA : APZIZborIzvajalcaZaUAPA 1:M

APZIZborIzvajalcaNacin : APZIZborIzvajalcaZaUAPA 1:M

Indeksi:

PK_ APZIZborIzvajalcaZaUAPA (clustered index nad IDizboraIzv, IDUAPA)

Pravila:

- Dopusča se možnost vnosa različnih nivojev UAPA.

▪ **Tabela APZIZborIzvajalcaZaPodUAPA**

Relacijska shema:

APZIZborIzvajalcaZaPodUAPA (# IDizboraIzv, # IDUAPA, # IDpodUAPA, Referent, DatSp)

Naziv atributa	Podatkovni tip	Obvezen vpis	Opombe
IDizboraIzv	int	NOT NULL	Identifikator načina izbora izvajalca
IDUAPA	smallint	NOT NULL	Identifikator nivojev in podnivojev UAPA
IDpodUAPA	smallint	NOT NULL	Identifikator podaktivnosti UAPA
Referent	char(4)	NOT NULL	Uporabnik, ki je zapis administriral (IDuporabnika tabele ZZUporabnikiAplikacij)
DATSP	smalldatetime	NOT NULL	Datum vnosa/spremembe zapisa

Relacije med tabelami:

APZ_IzborIzvajalcaZaUAPA : APZ_IzborIzvajalcaZaPodUAPA 1:M
 SifPodUAPA : APZ_IzborIzvajalcaZaPodUAPA 1:M

Indeksi:

PK_ APZ_IzborIzvajalcaZaPodUAPA (clustered index nad IDizboraIzv, IDUAPA, IDpodUAPA)

Pravila: /

▪ **Tabela APZRegulacijaVnosaIzborIzvajalca**

Tabela za regulacijo vnosnih polj na uporabniškem vmesniku v primeru vnosa podatkov o načinu izbora izvajalca, ki se razlikujejo glede na UAPA in glede na vrsto načina izbora izvajalca.

Relacijska shema:

APZRegulacijaVnosaIzborIzvajalca (#IDnacIzbora, #IDUAPA, #IDpodatka)

Naziv atributa	Podatkovni tip	Obvezen vpis	Opombe
IDnacIzbora	tinyint	NOT NULL	Identifikator vrste načina izbora izvajalca
IDUAPA	smallint	NOT NULL	Identifikator nivojev in podnivojev programov zaposlovanja (Ukrep, Aktivnost, PodAktivnosti)
IDpodatka	smallint	NOT NULL	Identifikator podatka za prikaz podatka na vnosni maski za UAPA in vrsto načina izbora izvajalca.

Relacije med tabelami:

SifNacinIzbora : APZRegulacijaVnosaIzborIzvajalca
 SifUAPA : APZRegulacijaVnosaIzborIzvajalca
 APZRegulacijaVnosaPodatki : APZRegulacijaVnosaIzborIzvajalca

Indeksi:

PK_ APZRegulacijaVnosaIzborIzvajalca (clustered index nad IDnacIzbora, IDUAPA, IDpodatka)

Pravila:

- Izbor izvajalca za izvajanje UAPA je možen na eno ali več vrst načina izbora izvajalca. Primer: Izvajalca za izvajanje UAPA 3.1 Institucionalno usposabljanje se lahko izbere po postopku ZJN (Zakon o javnih naročilih) ali ZJN – MV (Naročilo male vrednosti).
- Vrsta načina izbora izvajalca določa kateri podatki so ob vnosu podatkov o izboru izvajalca obvezni.

▪ Tabela APZRegulacijaVnosaPodatki

Pomožna tabela, ki vsebuje nabor možnih podatkov za katere je potrebno regulirati prikaz na uporabniškem vmesniku.

Relacijska shema:

APZRegulacijaVnosaPodatki (#IDpodatka, ImeTabele, ImeAtributa, OpisUporabe, DatVeljaDo, DatumVnosa)

Naziv atributa	Podatkovni tip	Obvezen vpis	Opombe
IDpodatka	smallint	NOT NULL	Identifikator podatka za prikaz UAPA in vrsto načina izbora izvajalca
ImeTabele	varchar(100)	NOT NULL	Ime tabele v kateri je shranjen podatek
ImeAtributa	varchar(50)	NOT NULL	Ime atributa v katerem je shranjen podatek
OpisUporabe	varchar(150)	NULL	Dodatni opis
DatVeljaDo	smalldatetime	NOT NULL	Datum veljavnosti uporabe podatka
DatumVnosa	smalldatetime	NOT NULL	Datum definiranja podatka

Relacije med tabelami:

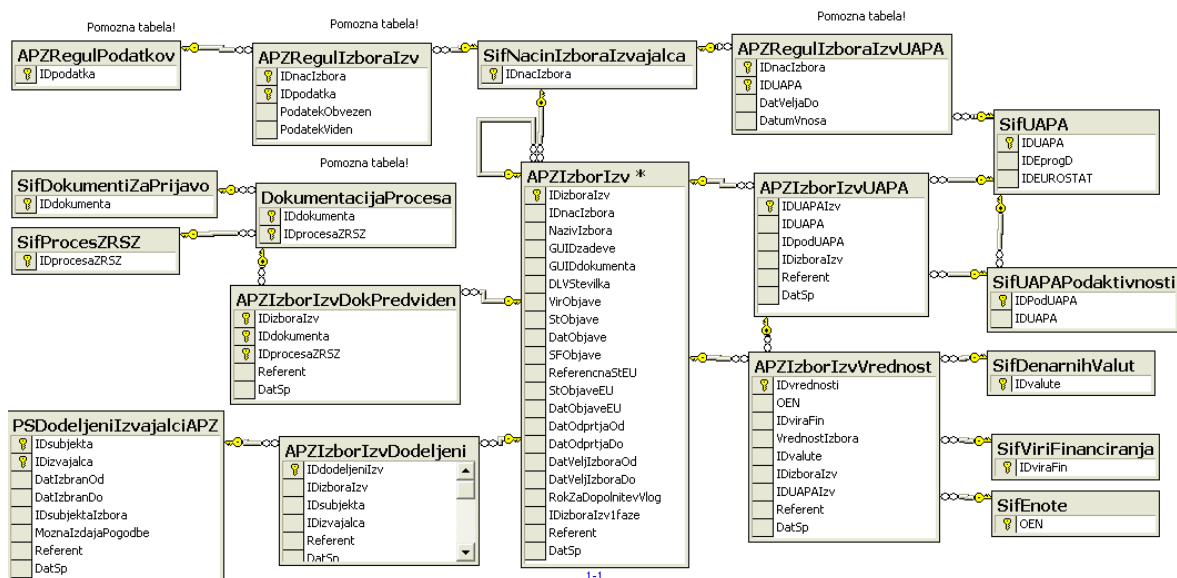
APZRegulacijaVnosaPodatki : APZRegulacijaVnosaIzborIzvajalca 1:M

Indeksi:

PK_ APZRegulacijaVnosaPodatki (clustered index nad IDpodatka)

Pravila: /

SLIKA 21: RELACIJSKI MODEL – IZBOR IZVAJALCA



4.2.5 Funkcionalnost aplikacije

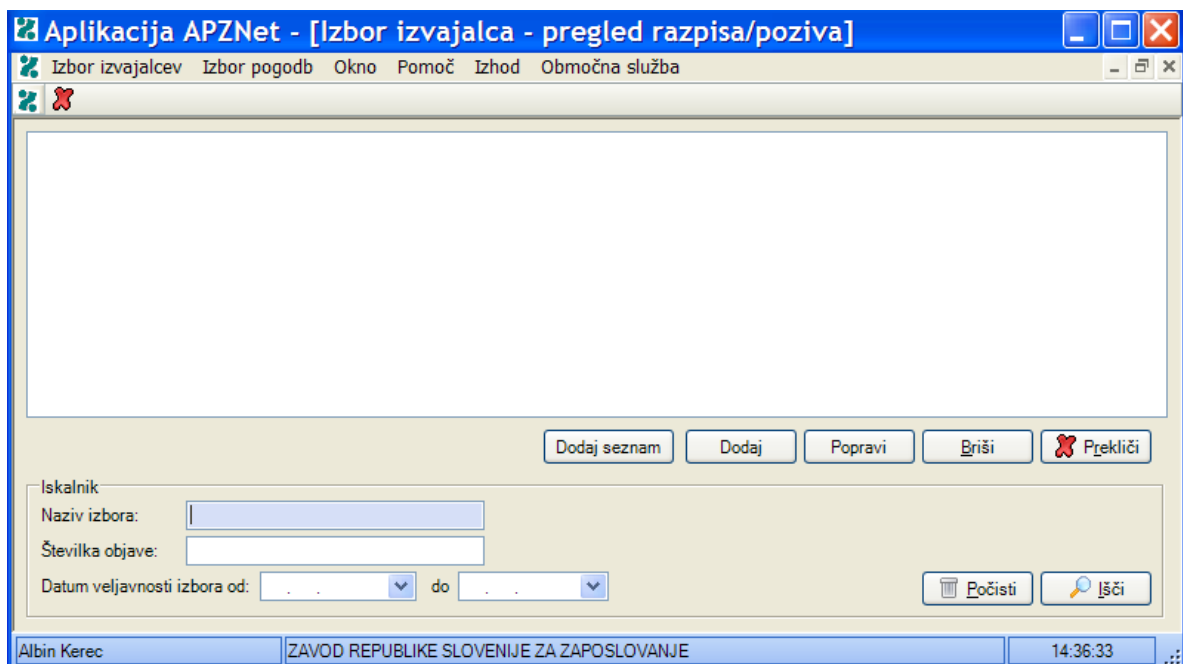
V aplikacijo ne bodo vgrajene nobene posebne funkcionalnosti, temveč se bodo uporabljale privzete Windows funkcionalnosti ter izbire preko menija in potrditvenih gumbov.

4.2.6 Ekranski vmesniki

Ekranski vmesnik je namenjen samo prikazu ter približni razvrstitvi podatkov, ki jih želimo prikazati. Izvedba samega vmesnika je prepuščena programerjem, možna pa so tudi večja odstopanja od same zasnove.

Na spodnjih slikah prikazujemo primer izvedbe ekranskega vmesnika za izbor izvajalca.

SLIKA 22: IZVEDEN EKRANSKI VMESNIK ZA IZBOR IZVAJALCA



4.2.7 Prenos podatkov

Prenos podatkov se je obstoječih aplikacij izvajal samo za ukrepe, ki se jim veljavnost 31. 12. 2006 ni iztekla. Vsi podatki iz starih aplikacij so se arhivirali.

Pri izvajanju aktivne politike zaposlovanje se namreč večino ukrepov prične z novim koledarskim letom. Zaradi tega se je aplikacijo tudi začelo uporabljati s 01.

01. 2007, kar pa je sovpadalo tudi z uvedbo €. Vse pogodbe, ki so veljale še iz preteklih let in se jih avtomatično ni dalo prenesti, so se vnesle ročno.

4.2.8 Programiranje, testiranje, izobraževanje in zaključek projekta

Programiranje izvaja programerska ekipa Zavoda v sestavi: Matej Nosan, Rihard Škorjanec, Aleksander Atelšek, Marko Dorbež, Brane Kne in Matej Kristan.

Poleg opisov vseh postopkov, definicij podatkov in izdelanega podatkovnega modela se je, za pomoč pri programiranju, za vsak proces napisala še specifikacija zahtev, ki je na krajši način programerju predstavila, kaj se od aplikacije pričakuje. V nadaljevanju je prikazan skrajšan primer specifikacije zahtev za izbor izvajalca:

Definicije

Način izbora pomeni pravno podlago in pot po kateri se izberejo izvajalci za izvajanje UAPA. V enem Načinu izbora vnašamo splošne podatke za različne nazive izbora npr. v okviru načina izbora PIP – lahko vnesemo Javni razpis za izbor javnih del in vse podatke o tem razpisu in pa še npr. javni razpis za starejše in vse podatke o tem razpisu.

Izvajalci so lahko:

1. izvajalci ukrepov, aktivnosti in pod aktivnosti aktivne politike zaposlovanja (npr. javnih del, klubov, institucionalna usposabljanja, 10000.....)
2. koristniki programov (delodajalci)
3. posamezne osebe, ki kot vlagatelji postanejo koristniki sredstev (samozaposlovanje)

Opis procesa

Ukrepi, aktivnosti in pod aktivnosti aktivne politike zaposlovanja (v nadaljevanju UAPA) se praviloma izvajajo tako, da imamo na eni strani osebo, ki jo vključujemo v UAPA, na drugi pa izvajalca, ki UAPA izvaja.

Zavod izvajalce za večino UAPA izbira sam in to na različne načine, ki so predpisani z zakoni, pravilniki in internimi navodili. Za UAPA so načini izbora vnaprej določeni, zato mora obstajati povezava med šifrantom UAPA in šifrantom načinov izbora izvajalcev.

V aplikaciji se izbere način izbora in vnese vse potrebne podatke o tem načinu. Gre za vnos formalnih podatkov o načinu izbora in o dokumentaciji, ki se zahteva od vlagateljev. Za vse vrste načinov izbora izvajalcev (razpisa, javnega poziva, naročila male vrednosti) se obvezno vnese naziv razpisa/poziva. Ostali podatki se vnesejo v odvisnosti od vrste načina izbora.

Ta proces opisuje pripravo pogojev za vnašanje posameznih vlog ali ponudb.

Splošna pravila

En UAPA ima lahko več načinov izbora, vendar se vnaša le enega naenkrat.

Znotraj posameznega načina izbora so različni nazivi aktivnosti za ta način izbora (npr. znotraj PIP IP: Javni razpis za javna dela, Javni razpis za izobraževanje).

Šifrant zahtevane dokumentacije se oblikuje ob vsakokratnem novem vnosu podatkov o načinu izbora za vsak način izbora (v katerem je to predvideno) posebej.

Šifrant UAPA se oblikuje ob vsakokratnem novem vnosu podatkov o načinu izbora, za vsak način izbora (v katerem je to predvideno) posebej.

Podatkovni model

Podatkovni model sestavljata definicija potrebnih podatkov, ki je bila opisana v poglavju 4.2.3. in podatkovni model opisan v poglavju 4.2.4.

Uporabniški vmesnik

Uporabniški vmesnik naj zajema spodaj prikazane elemente.

Na ekranu v okencu Način izbora se naj bodo prikazani načini izbora iz šifranta.

1. Izberi način izbora

1. Javni razpis PIP

1.1. Javni razpis PIP IP

1.2 Javni razpis PIP ESS

2. Javni razpis ZJN

2.1. ZJN –omejen razpis

2.1.1. ZJN – omejen razpis – I. faza Izbereš tega

2.1.2 ZJN – omejen razpis – II. faza kada izbereš to pri vnosu splošnih podatkov aplikacija vpraša na kateri ZJN 1. faza se naj aplikacija poveže. Pokaže vse razpise ZJN 1. faza, ki so znotraj obdobja veljavnosti.

2.2. ZJN – odprt razpis

2.3. ZJN – postopek s pogajanjem

3. JNMV - naročila male vrednosti

4. Javni poziv

5. Individualna vloga

6. Seznam

ZJN – omejen razpis – I. faza se Izpiše in ostane na ekranu.

2. Pokaže nabor UAPA, za katere je izbor izvajalcev predviden preko tega načina izbora izvajalcev.

Izberi UAPA: enega ali več – praviloma enega

Način izbora:	ZJN
Šifre UAPA:	1.2, 1.3, 2.1, 2.2, 3.1, 3.3.2

Izpiše se naziv UAPA in ostane na ekranu.

POMOČ PRI NAČRTOVANJU POKLICNE POTI IN ISKANJU ZAPOSLOTITVE

INSTITUCIONALNO USPOSABLJANJE

3. Pokažejo se tabele z nazivi podatkov, ki jih je potrebno vnesti za ta način izbora

Vnesi podatke o načinu izbora

NAZIV RAZPISA/POZIVA	X	Javni razpis za izbiro izvajalcev za izvedbo izobraževanj in usposabljanj za Pomoč pri načrtovanju poklicne poti in iskanju zaposlitve in za institucionalno usposabljanje
DELOVODNIŠKA ŠTEVILKA ZADEVE (splošno o javnem razpisu, naročilu ali drugem načinu izbora)		NPR. 961 ZA RAZPISE 962 ZA JN MV
KLASIFIKATOR ZA KNJIZENJE VLOG		Npr. 561
ODGOVORNA OSEBA ZA RAZPIS		
VIR OBJAVE	X	Uradni list
ŠTEVILKA OBJAVE	X	72/05 – možnost vnosa več števil
DATUM OBJAVE	X	15. 11. 2005
ŠIFRA OBJAVE	X	7239
REFERENČNA ŠTEVILKA – URADNI LIST EU	**	-
ŠTEVILKA OBJAVE V URADNEM LISTU EU	**	70/05
DATUM OBJAVE URADNEM LISTU EU	**	17. 12. 2005
DATUM ODPRTJA	X	15. 12. 2005
DATUM ZAPRTJA	X	15. 12. 2005
OBDOBJE VELJAVNOSTI IZBORA OD	X	01. 01. 2006
OBDOBJE VELJAVNOSTI IZBORA DO	X	31. 12. 2007
VREDNOST RAZPISA ZA VSE OEN SKUPAJ		-
VREDNOST RAZPISA PO VIRIH***		
ROK ZA DOPOLNITEV VLOGE	X	Pokaže se privzeto 8 dni

Potrdi – DA NE POPRAVI: Vnos: 15 dni

Podatki v zgornji tabeli, ki imajo x so pri tem načinu izbora izvajalca obvezni za vnos.

**Vse kar se tiče Uradnega lista EU se vpiše samo pod pogojem da javni razpis presega določeno vrednost. To pomeni, da se izpiše tudi kot tekst na izpisih če je tukaj vneseno.

*** V primeru, da izberemo pri enem vnosu dve šifri iz šifrantu UAPA, nam mora aplikacija omogočiti vnos vrednosti po virih in OEN za vsako šifro posebej in pokazati potem seštevek-skupno vrednost vnesenega.

4. Določi OEN, ki lahko na tem razpisu delajo:

CS LJ MB KP NG

Vpiši vrednost razpisa za izbrane OEN

MB 1000 SIT

KP 2000 SIT

NG 3000 SIT

5. Ali želiš pripraviti šifrant zahtevane dokumentacije za ta način izbora: DA NE

Pokaže se šifrant zahtevane dokumentacije – splošni. Označi zelene dokumente:

Št.	Opis
1.	Potrdilo DURS o poravnanih davkih in prispevkih
2.	Izpisek iz sodnega registra
3.	Nekaznovanost
4.	Izjava o sprejemu razpisnih pogojev
5.	Parafiran predlog pogodbe

Želiš Šifrant dopolniti: DA NE

Vnesi naziv dokumenta:

BILANCA STANJA ZA LETO 2004

Želiš Šifrant ZAKLJUČITI: DA NE

Pokaže pripravljen šifrant:

POTRDI POPRAVI

6. Ali želiš pripraviti Šifrant Pod UAPA za ta način izbora: DA NE

Izberi UAPA:

POMOČ PRI NAČRTOVANJU POKLICNE POTI IN ISKANJU ZAPOSLOTITVE
INSTITUCIONALNO USPOSABLJANJE

Vnesi šifro in naziv PodUAPA:

ŠIFRA	NAZIV
1a	Delavnica PDZ
1b	Delavnica Hitreje...
1c	Delavnica spoznajmo....

Želiš Šifrant ZAKLJUČITI: DA NE

7. Ali želiš pripraviti Šifrant Pod UAPA še za naslednji UAPA :

INSTITUCIONALNO USPOSABLJANJE

DA NE

8. Želiš zaključiti vnos načina izbora:

DA NE

9. Prikaz ključnih podatkov o načinu izbora:

NAZIV RAZPISA	Javni razpis za izbiro izvajalcev za izvedbo izobraževanj in usposabljanj za Pomoč pri načrtovanju poklicne poti in iskanju zaposlitve in za institucionalno usposabljanje
OBDOBJE VELJAVNOSTI IZBORA OD	01. 01. 2006
OBDOBJE VELJAVNOSTI IZBORA DO	31. 12. 2007
ŠIFRANT ZAHTEVANE DOKUMENTACIJE	POKAŽE ŠIFRANT
ŠIFRANT PODUAPA	POKAŽE ŠIFRANT

POTRDI POPRAVI

IZJEMA: V Prilogi 3 je opisana izjema ki jo je potrebno zagotoviti v aplikaciji in sicer seznam že izbranih izvajalcev. Poleg splošnih podatkov o načinu izbora je potrebno omogočiti še vnos izbranih izvajalcev v skladu s postopkom opisanem v Prilogi 3.

Informacijski pregledi

1. prikaz na ekranu:
 - pregled podatkov o izbranem načinu izbora
 - pregled podatkov o vseh načinih izbora za posamezen UAPA v določenem obdobju
 - pregled aktivnih načinov izbora
2. prenos v Excel

Programiranje

Na osnovi specifikacije zahtev je programerska skupina pričela s programiranjem aplikacije. Tudi tu je bilo dogovorjeno, da se dobiva v testiranje verzije po posameznih sklopih (izbor izvajalca, vnos UAPA in kvot, vnos vloge...), ki se testirajo, napake pa se sproti javljajo. Prejeti popravki se ponovno testirajo do verzije, ki izpolnjuje vse zahteve (želje) testerjev (uporabnikov).

Testiranje

S testiranjem se je preverjalo, če so vse zahtevane funkcionalnosti vključene v program in če delujejo pravilno oz. tako kot se je pričakovalo. Napake so bile takoj javljene in z novimi verzijami odpravljene.

Izobraževanje

Pripravljeno je bilo usposabljanja za vse uporabnike. Napisan je bil tudi priročnik za uporabo aplikacije, ki se stalno dopolnjuje.

Zaključek projekta

Po terminskem planu se bo projekt zaključil 01. 07. 2007, ko bo v uporabi celotna aplikacija. Sam razvoj se s tem seveda ne bo zaključil, saj bodo potrebne še številne spremembe v aplikaciji. Gre predvsem za izboljšanje funkcionalnosti in odpravo nekritičnih napak.

5 SKLEP

Pri izdelavi programske opreme in zagotavljanju njene kakovosti je nujno treba upoštevati katero od metod nabave programske opreme. Metode nabave predpisujejo natančne postopke dela, vendar nam kljub temu, da se teh postopkov držimo, ne zagotavljajo končnega uspeha. Postavljajo nam le smernice in če jih upoštevamo, je možnost uspeha bistveno večja.

Na Zavodu za zaposlovanje teh metod nismo poznali. Že na začetku smo si sami načrtali način dela, po katerem smo nabavili programsko opremo in se ga tudi držali. Glede na to, da smo metodo postavili sami in smo vanjo tudi verjeli, je bila verjetnost, da bo končni izdelek kvaliteten, zelo velika. Ugotavljam, da smo kljub nepoznavanju metod, v našem primeru, upoštevali veliko smernic iz različnih metod, ki smo jih prilagodili našemu znanju in sposobnostim. Pri tem smo upoštevali tudi življenjski cikel programske opreme, vendar smo vrstni red postopkov prilagodili našemu načinu dela.

Za uspeh pri vseh opisanih projektih je bila bistvena prva faza projekta, kjer so se procesi zelo natančno popisali. S tako natančnim opisom kasneje ni bilo težko pripraviti ostale dokumentacije. Bistveno je bilo tudi, da smo, kot uporabniki oz. naročniki, natančno vedeli, kaj želimo. Tudi odločitev, da želimo imeti aplikacijo narejeno z najsodobnejšimi orodji (.Net in MS SQL) je pripomogla h kvalitetnejšemu končnemu izdelku in s tem zadovoljstvu uporabnikov.

Zavod se je na začetku prenavljanja programske opreme po CMMI metodi nahajal v prvem zmožnostnem nivoju. Po opisani metodi se sedaj nahaja v tretjem zmožnostnem nivoju, nekateri postopki, ki se izvajajo, pa že nakazujejo možnost prehoda v četrti in peti nivo. Določene aktivnosti, ki se že izvajajo, so namreč že del četrtega in petega zmožnostnega nivoja. Za uspešnost projekta se mora organizacija nahajati najmanj v tretjem zmožnostnem nivoju.

Glede na zrelostno raven se je Zavod nahajal na drugem nivoju, z opredeljeno metodo pa se sedaj nahaja na tretjem nivoju. V spodnji tabeli so s „✓“ označena procesna področja, ki jih Zavod izpolnjuje, z „x“ pa področja, ki jih še ne izpolnjuje v celoti.

Procesno področje	Kategorija	Zrelostni nivo	Zavod
1. Vzorčna analiza in zaključki	Podpora	5	x
2. Konfiguracijski management	Podpora	2	✓
3. Odločitvena analiza in zaključki	Podpora	3	✓
4. Integrirano vodenje projektov	Upravljanje projektov	3	✓
5. Merjenje in analiza	Podpora	2	✓
6. Organizacijska prenova in razvoj	Upravljanje procesov	5	✓
7. Organizacijska definicija procesov	Upravljanje procesov	3	✓
8. Izboljšanje procesov v organizaciji	Upravljanje procesov	3	✓
9. Izvedba organizacijskega procesa	Upravljanje procesov	4	✓
10. Izobraževanje v organizaciji	Upravljanje procesov	3	✓
11. Produktna integracija	Inženiring	3	✓
12. Projektno spremljanje in nadzor	Upravljanje projektov	2	✓
13. Planiranje projektov	Upravljanje projektov	2	✓
14. Zagotavljanje kakovosti procesov in proizvodov	Podpora	2	✓
15. Kvantitativno projektno vodenje	Upravljanje projektov	4	x
16. Razvojne zahteve	Inženiring	3	✓
17. Vodstvene zahteve	Inženiring	2	✓
18. Rizično vodenje	Upravljanje projektov	3	✓
19. Upravljanje s pogodbami dobaviteljev	Upravljanje projektov	2	✓
20. Tehnične rešitve	Inženiring	3	✓
21. Validacija	Inženiring	3	✓
22. Verifikacija	Inženiring	3	✓

Na osnovi prikazanega ugotavljam, da Zavod izpolnjuje vse zahteve prvega, drugega in tretjega nivoja ter nekatere zahteve četrtega in petega nivoja. Za uspešnost projekta pa mora organizacija izpolnjevati vse zahteve tretjega nivoja.

Projekt izdelave aplikacije ZP_Net je bil že zaključen. Po zaključku projekta in odpravi vseh večjih pomanjkljivosti sledi optimizacija (po CMMI), kar se sedaj dogaja.

V predstavljeni metodi so upoštevana tudi vsa štiri področja Cobit modela, vendar ne vsi procesi, ki jih Cobit opredeljuje. Na podlagi pridobljenih informacij iz analize procesov, metoda določa načrtovanje in planiranje razvoja programske opreme, določa nabavo (naročanje ali nakup) in uvajanje programske opreme, določa pomoč in podporo uporabnikom in na koncu tudi kontrolo in ocenjevanje programske opreme. S tem je v metodi zajet velik del Cobit procesa, izločeni pa so procesi, ki bi pomenili dodatno obremenitev virov (ljudi in sredstev), ne bi pa prinesli bistvenih pridobitev v končanem programskem produktu.

Ena od bistvenih težav, večine podjetij, pri projektih je, kdaj s projektom zaključiti. Na Zavodu smo ugotovili, da je projekt izdelave programskega produkta potrebno zaključiti, ko aplikacija izpolnjuje vse predvidene osnove funkcionalnosti in ima odpravljene vse bistvene napake. Razvoj aplikacije namreč in nikoli dokončan in bi se projekt zaradi tega lahko vlekel v nedogled. To je tudi točka, ko se aplikacija lahko v celoti začne uporabljati v produkciji. Od tega trenutka dalje prevzame skrb za popravljanje, spreminjanje in razvoj aplikacije služba, ki aplikacijo uporablja.

Oblikovana metoda je bila preverjena pri naročanju in izdelavi programske opreme. Dokazano je, da je uporabna v obeh primerih. Njena uporabnost se je izkazala se je tudi pri manjših projektih prenove programskih produktov, kot je bil na primer projekt obveščanja brezposelnih oseb z SMS sporočili in po elektronski pošti.

Pričakujem, da se bo opisana metoda uporabljala pri vseh projektih razvoja programske opreme, saj se je izkazala za dovolj uporabno. Če se bo izkazalo, da v kakšnem primeru ni popolnoma uporabna, jo bo potrebno še dopolniti in ustrezno prilagoditi. Trenutno se je še v vseh primerih izkaza kot primerna.

SEZNAM LITERATURE IN VIROV

Literatura

1. Armstrong, Deborah J. 2006. The quarks of object-oriented development. New York: ACM Press.
2. Black Rex. 1999. Managing the Testing Process. Redmond: Microsoft Press.
3. Bregar, Andrej. (2004a). Specifikacija zahtev za programsko opremo (Povzeto po standardu IEEE Std 830-1984) [online]. UM FERi – Univerza v Mariboru Fakulteta za elektrotehniko, računalništvo in informatiko – Inštitut za informatiko. Dostopno na:
[<http://lisa.uni-mb.si/~bregar/Vaje/IEEE-STD-830-1984.pdf>]. [26. 02. 2007].
4. Bregar, Andrej. (2004b). Načrt testiranja programske opreme (Povzeto po standardu IEEE Std 829-1998) [online]. UM FERi – Univerza v Mariboru Fakulteta za elektrotehniko, računalništvo in informatiko – Inštitut za informatiko. Dostopno na:
[<http://lisa.uni-mb.si/~bregar/Vaje/Nacrt%20testiranja%20PO.pdf>]. [26. 02. 2007].
5. CobiT 4.0. 2005. Rolling Meadows: IT Governance Institute.
6. CVI – Vlada Republike Slovenije. Center vlade RS za informatiko. 2001. Metodologija vodenja projektov v državni upravi: projekti informacijske tehnologije: priročnik. Ljubljana: Center Vlade RS za informatiko.
7. Dogša, Tomaž. 1993. Verifikacija in validacija programske opreme. Maribor: Založniško tiskarska dejavnost Tehniške fakultete.
8. Gradišar, M., J. Jaklič, T. Damij in P. Baloh. 2005. Osnove poslovne informatike. Ljubljana: Ekonomska fakulteta.
9. Hauc, Anton. 1995. Projektni management. Maribor: EPF.
10. Hauc, Anton. 2002. Projektni management. Ljubljana: GV založba.
11. IEEE - The Institute of Electrical and Electronics Engineers. 1998a. IEEE Recommended Practice for Software Requirements Specifications (IEEE Std 830-1998). New York: The Institute of Electrical and Electronics Engineers, Inc.
12. IEEE - The Institute of Electrical and Electronics Engineers. 1998b. IEEE Standard for Software Test Documentation (IEEE Std 829-1998). New York: The Institute of Electrical and Electronics Engineers, Inc.
13. ISO 9000:2000.
14. Kerzner, Harold. 1992. Project Management A Systems Approach To Planning, Scheduling, and Controlling. New York: Van Nostrand Reinhold.
15. Kovačič, Andrej, in Mirko Vintar. 1994. Načrtovanje in gradnja informacijskih sistemov. Ljubljana: DZS.
16. Object Management Group (OMG). 2005b. Unified Modeling Language Specification. [online]. Dostopno na:
[<http://www.omg.org/cgi-bin/doc?formal/05-04-01>]. [04. 02. 2007].

17. Object Management Group (OMG). 2006a. Unified Modeling Language: Infrastructure. [online]. Dostopno na: [\[http://www.omg.org/docs/ptc/06-04-03.pdf\]](http://www.omg.org/docs/ptc/06-04-03.pdf). [04. 02. 2007].
18. Object Management Group (OMG). 2006b. Diagram Interchange Specification. [online]. Dostopno na: [\[http://www.omg.org/cgi-bin/apps/doc?formal/06-04-04.pdf\]](http://www.omg.org/cgi-bin/apps/doc?formal/06-04-04.pdf). [16. 02. 2007].
19. Pivka, Marjan. 1996. Kakovost v programskem inženirstvu. Izola: Desk d. o. o.
20. Potočnik E., T. Babnik, F. Černe, U. Gunčar, M. Kiauta, R. Novak, M. Pivka, J. Potočnik, J. Čuk in M. Mihelčič. 1996. ISO 9001 - iz teorije v prakso. Ljubljana: Taxus.
21. Prabhu C. S. R.: Object-oriented Database Systems: Approaches and Architectures. 2005. New Delhi: Prentice-Hall of India.
22. Rainer, Rex Kelly, Efraim Turban in Richard Potter. 2007. Introduction to information systems. Hoboken: John Wiley & Sons, inc.
23. Rupnik, Rok. (2005). Osnove informacijskih sistemov. Univerza v Ljubljani, Fakulteta za računalništvo in informatiko. Dostopno na:
24. Ruthberg Zella G., Fisher Bonnie T., Lainhart John W. IV: System Development Auditor. 1991. Oxford: Elsevier Advanced Technology.
25. SEI – Software Engineering Institute. 2002. Capability Maturity Model Integration (CMMISM) Version 1.1. [online]. Dostopno na: [\[http://www.sei.cmu.edu/cmimi/models/cont-to-v1.1.doc\]](http://www.sei.cmu.edu/cmimi/models/cont-to-v1.1.doc). [8.1.2006].
26. SEI – Software Engineering Institute. 2006. CMMI for Development, Version 1.2. [online]. Dostopno na: [\[http://www.sei.cmu.edu/pub/documents/06.reports/pdf/06tr008.pdf\]](http://www.sei.cmu.edu/pub/documents/06.reports/pdf/06tr008.pdf) [09. 02. 2007]
27. Solina, Franc. 1998. Projektno vodenje razvoja programske opreme. Ljubljana: Fakulteta za računalništvo in informatiko.
28. Srića, Velimir, Sonja Treven in Mile Pavlič. 1995. Informacijski sistemi. Ljubljana: Gospodarski vestnik.
29. SSK – Slovar slovenskega knjižnega jezika. Elektronska izdaja verzija 1.0. DZS, Ljubljana
30. Vallabhaneni S. Rao. 1996. CISA Examination Textbooks, Volume 1: Theory. Second Edition. SRV Professional Publications.
31. Warren, J. Donald. Edelson Lynn W., Parker Xenia Ley. 1998. Handbook of IT Auditing. New York: Warren, Gorham & Lamont.
32. ISO 9000 – Napotki za procesni pristop v sistemu vodenja kakovosti. 2006. [online]. Dostopno na: [\[http://www.sist.si/slo/g3/napotki_pristop.pdf\]](http://www.sist.si/slo/g3/napotki_pristop.pdf). [15. 01. 2006].
33. CAA – Civil Aviation Authority. 2007. [online]. Dostopno na: [\[http://www.caa.co.uk/docs/5/ergdocs/nats_triggerscmmi_may05.pdf\]](http://www.caa.co.uk/docs/5/ergdocs/nats_triggerscmmi_may05.pdf). [06. 01. 2007].
34. SIST – Slovenski inštitut za standardizacijo. 2003. Napotki za procesni pristop v sistemu vodenja kakovosti. [online]. Dostopno na: [\[http://www.sist.si/slo/g3/napotki_pristop.pdf\]](http://www.sist.si/slo/g3/napotki_pristop.pdf). [15.1.2006].

Viri

1. CIO - Research & Analysis – Business Technology Leadership. 2007. [online]. Dostopno na: [\[http://www2.cio.com/analyst/report2669.html\]](http://www2.cio.com/analyst/report2669.html). [06. 01. 2007].
2. Micro Process. 2007. [online]. Dostopno na: [\[http://www.microprocess.si/rev.htm\]](http://www.microprocess.si/rev.htm). [06. 01. 2007].
3. Object Management Group (OMG). 2007. UML® Resource Page. [online]. Dostopno na: [\[http://www.omg.org/technology/uml/index.htm\]](http://www.omg.org/technology/uml/index.htm). [04. 02. 2007].
4. Object Management Group (OMG). 2007a. Introduction to OMG's Unified Modeling Language™ (UML®) [online]. [17. 01. 2007] Dostopno na: [\[http://www.omg.org/gettingstarted/what_is_uml.htm\]](http://www.omg.org/gettingstarted/what_is_uml.htm). [05. 02. 2007].
5. Wikipedia. 2007. [online]. Dostopno na: [\[http://en.wikipedia.org/wiki/Object-oriented_programming\]](http://en.wikipedia.org/wiki/Object-oriented_programming). [09. 02. 2007].
6. CVI – Center vlade za informatiko. 2006. [online]. Dostopno na: [\[http://www.gov.si/cvi/slo/projb/cvi-mvp.htm\]](http://www.gov.si/cvi/slo/projb/cvi-mvp.htm). [24. 04. 2006].

SEZNAM UPORABLJENIH KRATIC

AK	- Alternativni ključ
APZ	- Aktivna politika zaposlovanja
BO	- Brezposelna oseba
CS	- Centralna služba
CVI	- Center vlade za informatiko
DM	- Delovno mesto
DN	- Denarno nadomestilo
DP	- Denarna pomoč
DZ	- Drugi zakoni
EMŠO	- Enotna matična številka osebe
ESS	- Evropski strukturni sklad
EU	- Evropska unija
FS	- Funkcijska specifikacija (specifikacija zahtev za programsko opremo)
HW	- Strojna oprema (hard ware)
IAS	- Identifikacija, avtorizacija in spremljanje uporabnikov aplikacij ZRSZ
IS	- Informacijski sistem
JNMV	- Javno naročilo male vrednosti
M1	- Obrazec za prijavo delavca na Zavod za pokojninsko in invalidsko zavarovanje
MV	- Mala vrednost
OEN	- Organizacijska enota – šifra urada za delo, območne službe ali centralne službe
OS	- Območna služba
PCMG	- Pospeševalni center za malo gospodarstvo
PIP	- Pravilnik o izvrševanju proračuna
PIP ESS	- Pravilnik o izvrševanju proračuna – sredstva strukturne politike
PIP IP	- Pravilnik o izvrševanju proračuna – integralni proračun
PK	- Primarni ključ
PM	- Podatkovni model
PV	- Privzeta vrednost
RIZ	- Služba za izvajanje zavarovanja oseb za primer brezposelnosti
RS	- Republika Slovenija
SKP	- Standardna klasifikacija poklicev
SQL	- strukturirani povpraševalni jezik za delo s podatkovnimi bazami (Structured Query Language)
SW	- Programska oprema (soft ware)
SZPO	- Specifikacija zahtev za programsko opremo
TK	- Tuji ključ
UAPA	- Ukrepi, aktivnosti in pod aktivnosti aktivne politike zaposlovanja
UD	- Urad za delo
UE	- Upravna enota
Ur. l.	- Uradni list

UV	- Uporabniški vmesnik (ekranski vmesnik)
ZIPRO	- Zakonu o izvrševanju proračuna
ZJN	- Zakon o javnih naročilih
ZN	- Zaposlitveni načrt
ZP	- Zaposlovanje
ZPIZ	- Zavod za pokojninsko in invalidsko zavarovanje
ZRSZ	- Zavod Republike Slovenije za zaposlovanje
ZUP	- Zakon o upravnem postopku

SEZNAM SLIK

SLIKA 1: KRIZA PROGRAMSKE OPREME.....	9
Slika 2: Razvoj CMMI modela.....	17
SLIKA 3: KOMPONENTE CMMI MODELA.....	18
Slika 4: Prikaz Cobit procesa.....	29
SLIKA 5: PLANIRANJE IT PROCESOV NA IT VODSTVENIH PODROČJIH, COSO PODROČJIH, COBIT IT SREDSTVIH IN COBIT INFORMACIJSKIH KRITERIJIH.....	32
Slika 6: Model sistema vodenja kakovosti, ki temelji na procesu (vzet iz ISO 9000:2000).....	37
SLIKA 7: POLNA ORGANIZACIJSKA STRUKTURA PROJEKTA	49
Slika 8: Groba skica postopkovnega modela za primer prijave brezposelne osebe v evidenco.....	54
SLIKA 9: DTP DIAGRAM PRIJAVE.....	57
SLIKA 10: E-R DIAGRAM.....	60
SLIKA 11: PREDLAGAN EKRANSKI VMESNIK ZAVIHKA OSEBE.....	61
SLIKA 12: GANTOV DIAGRAM PRIKAZA PROJEKTA.....	68
Slika 13: Groba skica postopkovnega modela za primer PRIJAVE IN OBRAVNAVE brezposelne osebe.....	73
SLIKA 14: DIAGRAM PROCESA ZAPOSLOVANJA.....	79
SLIKA 15: DTP DIAGRAM ZA EVIDENTIRANJE IN SPREMLJANJE BO.....	80
SLIKA 16: DTP DIAGRAM ZA PRIJAVO OSEB V EVIDENCO.....	81
Slika 17: Entitetno-Relacijski model – Izobrazba osebe.....	87
SLIKA 18: PREDLAGAN EKRANSKI VMESNIK OSNOVNIH PODATKOV O OSEBI....	89
SLIKA 19: IZVEDEN EKRANSKI VMESNIK OSNOVNIH PODATKOV O OSEBI.....	89
SLIKA 20: GROBA SKICA POSTOPKOVNEGA MODELA ZA IZBOR IZVAJALCA.....	103
Slika 21: Relacijski model – IZBOR IZVAJALCA.....	113
SLIKA 22: IZVEDEN EKRANSKI VMESNIK ZA IZBOR IZVAJALCA.....	114

DELOVNI ŽIVLJENJEPIŠ

Osební podatki: Drago Perc
Rojen 9. 12. 1967 v Celju
Državljanstvo: slovensko

Šolanje:

1974-1982 Osnovna šola Savo Kladnik Sevnica
1982-1986 Srednja šola za ekonomsko usmeritev Tone Grčar Celje

Študij: Univerza v Mariboru, Ekonomsko-poslovna fakulteta

1987-1992 Višješolski študij na smeri ekonomska informatika (ekonomist)
1991-1995 Visokošolski študij na smeri ekonomska informatika (univerzitetni diplomirani ekonomist)
Diploma: Analiza baze podatkov Republiškega zavoda za zaposlovanje in izdelava konceptualnega modela
1997- Magistrski študij na smeri poslovna informatika

Izobraževanja:

1995 – Administracija NetWare 3.1
1996 – Uporaba urejevalnika Word, preglednice Excel in interneta
2001 – Udeležba na delavnici „Zastavljanje ciljev, spremljanje in evalvacija“
2001 – Študijski obisk na avstrijskem zavodu za zaposlovanje (AMS)
2002 – Delavnica „Intervju“
2003 – Računalniški tečaj – Vrtilne tabele v Excelu
2004 – Izobraževanje „Proclarity“
2004 – Delavnica „Vodenje skupin“

Zaposlitve:

1989-1989 Ljubljanska banka Krško – delo v obračunskem centru
1994-1996 Zavod Republike Slovenije za zaposlovanje območna služba Sevnica – informatik
1997-1999 Zavod Republike Slovenije za zaposlovanje območna služba Sevnica – vodja informatike
1999- Zavod Republike Slovenije za zaposlovanje centralna služba Ljubljana – koordinator zaposlovanja

Pomembnejše delovne naloge:

- 1995- Izobraževanje sodelavcev na različnih področjih (izobraževanja za uporabo različnih računalniških aplikacij (Microsoft Office, lastnih zavodskih aplikacij), vsebinska izobraževanja (zakonske in druge spremembe), pisanje navodil in priročnikov

- 1996-1996 Sodelovanje v delovni skupini za vzpostavitev elektronske pošte na Zavodu

- 1997-1998 Vodenje delovne skupine za pripravo kataloga oseb in kataloga podjetij na Zavodu Republike Slovenije za zaposlovanje

- 1999-2000 Sodelovanje v delovni skupini za vzpostavitev skupne baze podatkov na SQL strežniku

- 2001-2003 Vodenje delovne skupine za pripravo postopkovnega in podatkovnega modela za prenovo računalniške aplikacije za podporo dela svetovalcev zaposlitve

- 2002-2002 Član strokovne komisije za ocenitev ponudb javnega razpisa

- 2002-2005 Član projektne skupine za pripravo informacijske podpore za področje zaposlovanja

- 2004- Član delovne skupine za pripravo enotnega sistema evidenc na področju socialnega zavarovanja na ministrstvu za delo, družino in socialne zadeve

- 2006-2007 Član delovne skupine za uvedbo evra

Ostala dela:

- 1995-1999 – Računalniško opismenjevanje oseb (delo za različna računalniška podjetja in Ljudsko univerzo Krško)

Bibliografija:

- 1995 Diploma: Analiza baze podatkov Republiškega zavoda za zaposlovanje in izdelava konceptualnega modela

- 2000 Priročnik za nadaljevalni tečaj Microsoft Word-a

- 2000 Navodila za vnos podatkov v aplikacijo ZP

- 2000 Tehnična dokumentacija za aplikacijo ZP
- 2004 Priročnik za uporabo aplikacije ZP_Net
- 2004 Priročnik za uporabo aplikacije ZP_Net – modul Posredovanje
- 2005 Navodilo za prijavo podatkov o pokojninskem in invalidskem ter zdravstvenem zavarovanju, zavarovanju za starševsko varstvo, zavarovanju za primer brezposelnosti in o sklenitvi delovnega razmerja (M-1, M-2, M-3, M-3a, M-DC) - soavtor
- 2006 Priročnik za uporabo aplikacije ZP_Net – modul Invalidi