

UNIVERZA V LJUBLJANI

EKONOMSKA FAKULTETA

MAGISTRSKO DELO

**RAZVOJ INFORMACIJSKEGA SISTEMA
LOGISTIKE Z UPORABO METODOLOGIJE
TABELARNI RAZVOJ APLIKACIJ (TAD)**

Ljubljana, junij 2002

MILOŠ BLANC

IZJAVA

Študent Miloš Blanc izjavljam, da sem avtor tega magistrskega dela, ki sem ga napisal pod mentorstvom prof. dr. Damij Taliba in skladno s 1. odstavkom 21. člena Zakona o avtorskih in sorodnih pravicah dovolim objavo magistrskega dela na fakultetnih spletnih straneh.

V Ljubljani, 27.6.2002

Podpis:_____

KAZALO VSEBINE

1. UVOD.....	1
1.1. NAMEN DELA	1
1.2. CILJI DELA	2
1.3. METODE DELA	2
1.4. STRUKTURA DELA	2
2. O RAZVOJU INFORMACIJSKIH SISTEMOV	4
2.1. TRADICIONALNI ŽIVLJENJSKI CIKEL RAZVOJA SISTEMA	5
2.2. UPORABA PROTOTIPOV	9
2.3. RAD	11
2.4. INFORMACIJSKO INŽENIRSTVO	14
2.4.1. Načrtovanje informacijske strategije	15
2.4.2. Analiza poslovnega področja.....	16
2.4.3. Načrtovanje sistema	17
2.4.4. Konstrukcija in prehod.....	17
2.5. POSTOPNI NARAŠČAJOČI MODEL RAZVOJA IS	18
3. PRENOVA POSLOVNIH PROCESOV	20
3.1. IZHODIŠČA ZA PRENOVO POSLOVNIH PROCESOV	20
3.1.1. Organizacija podjetja in poslovni procesi.....	21
3.2. PRENOVA PROCESOV	22
3.2.1. Informacijska tehnologija omogoča prenovo procesov	23
3.2.2. Pristop k preurejanju procesov.....	23
3.2.3. Kadri, potrebni za prenovo.....	24
3.2.4. Koraki prenove poslovnih procesov	25
3.2.5. Posledice prenove poslovnih procesov.....	29
3.2.6. Tveganja pri prenovi poslovnih procesov	31
4. OBJEKTNO USMERJENE METODOLOGIJE RAZVOJA INFORMACIJSKIH SISTEMOV	32
4.1. OSNOVNI KONCEPTI OBJEKTNEGA PRISTOPA.....	32
4.1.1. Objekt	32
4.1.2. Ograjevanje	32
4.1.3. Razred.....	33
4.1.4. Asociacije.....	34
4.1.5. Sporočilo.....	35
4.1.6. Mnogoličnost.....	35
4.1.7. Dedovanje	35
4.2. POENOTEN JEZIK ZA MODELIRANJE – UML	37
4.2.1 Osnovni gradniki UML.....	38
4.2.1.1. Elementi	38
4.2.1.2. Povezave	39
4.2.1.3. Diagrami	40
4.2.2. Poenoteni proces razvoja programske opreme.....	47
4.2.2.1. Ključne aktivnosti procesa.....	48
4.2.2.2. Glavne faze procesa	52

5. METODOLOGIJA »TABELARNI RAZVOJ APLIKACIJ« (TAD).....	54
5.1. OSNOVNE ZNAČILNOSTI METODOLOGIJE.....	54
5.2. DEFINIRANJE PROBLEMA	54
5.3. ANALIZA DELOVANJA SISTEMA.....	56
5.3.1. <i>Oprelitev aktivnosti</i>	56
5.3.2. <i>Oprelitev nalog</i>	57
5.3.3. <i>Oprelitev delovnih in poslovnih procesov</i>	58
5.4. PRENOVA POSLOVNIH PROCESOV	59
5.4.1. <i>Analiza in prenova tabel</i>	59
5.4.2. <i>Procesni model</i>	60
5.5. OBJEKTNI MODEL SISTEMA	62
5.5.1. <i>Razvoj začetnega objektnega modela</i>	62
5.5.2. <i>Končni objektni model</i>	63
5.6. NAČRTOVANJE.....	63
5.6.1. <i>Oprelitev operacij</i>	63
5.6.2. <i>Aplikacijski model</i>	64
5.7. IMPLEMENTACIJA SISTEMA	64
6. RAZVOJ INFORMACIJSKEGA SISTEMA LOGISTIKE	65
6.1. PREDSTAVITEV PODJETJA IN NAMENA DELOVANJA DISTRIBUCIJSKEGA CENTRA	65
6.2. UMETITEV OBSTOJEČEGA IS LOGISTIKE V ŠIRŠI KONTEKST IS PODJETJA	68
6.3. DEFINICIJA PROBLEMA	71
6.4. ANALIZA DELOVANJA SISTEMA.....	74
6.4.1. <i>Tabela aktivnosti</i>	75
6.4.2. <i>Tabela nalog</i>	76
6.4.3. <i>Oprelitev delovnih in poslovnih procesov</i>	80
6.5. PRENOVA POSLOVNIH PROCESOV	82
6.5.1. <i>Analiza in prenova tabel</i>	82
6.5.2. <i>Procesni model</i>	84
6.6. OBJEKTNI MODEL SISTEMA	89
6.6.1. <i>Začetni objektni model</i>	89
6.6.2. <i>Končni objektni model</i>	93
6.7. NAČRTOVANJE.....	95
6.7.1. <i>Definiranje operacij</i>	95
6.7.2. <i>Aplikacijski model</i>	95
6.8. IMPLEMENTACIJA	98
7. ZAKLJUČEK	99
8. LITERATURA IN VIRI	100
8.1. LITERATURA:	100
8.2. VIRI:.....	102

PRILOGA

KAZALO SLIK

Slika 1:	Tradicionalni model življenjskega cikla sistema.....	6
Slika 2:	Problem konvencionalnega načina razvoja programske opreme	8
Slika 3:	Piramida informacijskega inženirstva	14
Slika 4:	Postopni naraščajoči model razvoja IS	18
Slika 5:	Funkcijski »silosi«	21
Slika 6:	Vpliv temeljnih ciljev na uspešnost prenove poslovanja	28
Slika 7:	Stopnje izboljšav poslovnega procesa	30
Slika 8:	Objekt vsebuje podatke in operacije.....	32
Slika 9:	Primerki razreda	33
Slika 10:	Primer asociacij med razredi	34
Slika 11:	Primer dedovanja	36
Slika 12:	Razvoj jezika UML.....	37
Slika 13:	Primer razrednega diagrama	41
Slika 14:	Primer objektnega diagrama	41
Slika 15:	Diagram komponent	42
Slika 16:	Diagram porazdelitve	42
Slika 17:	Diagram primerov uporabe	44
Slika 18:	Diagram zaporedja	45
Slika 19:	Diagram sodelovanja.....	46
Slika 20:	Diagram stanj objekta :kupec	46
Slika 21:	Diagram aktivnosti	47
Slika 22:	Faze in aktivnosti poenotenega procesa	48
Slika 23:	Modeli poenotenega procesa	49
Slika 24:	Del poslovnega procesa prodajanje	61
Slika 25:	Sestava Skupine Mercator.....	65
Slika 26:	Organizacijska shema.....	67
Slika 27:	Povezave IS logistike z drugimi informacijskimi sistemi	69
Slika 28:	Prikaz procesa nabavljanja.....	80
Slika 29:	Prikaz procesa prodajanja	81
Slika 30:	Procesni model del 1/3.....	88
Slika 31:	Funkcionalne odvisnosti med atributi.....	89
Slika 32:	Povezave med identifikacijskimi atributi	90
Slika 33:	Integracija analize.....	90
Slika 34:	Razredi in njihove medsebojne povezave.....	91
Slika 35:	Začetni objektni model.....	92
Slika 36:	Končni objektni model	94
Slika 37:	Aplikacijski model.....	97

KAZALO TABEL

<i>Tabela 1:</i>	<i>Tabela entitet.....</i>	<i>55</i>
<i>Tabela 2:</i>	<i>Del tabele aktivnosti logistike</i>	<i>57</i>
<i>Tabela 3:</i>	<i>Del tabele nalog logistike</i>	<i>58</i>
<i>Tabela 4:</i>	<i>Tabela entitet.....</i>	<i>74</i>
<i>Tabela 5:</i>	<i>Tabela aktivnosti - obstoječe stanje 1/2.....</i>	<i>77</i>
<i>Tabela 6:</i>	<i>Tabela aktivnosti – obstoječe stanje 2 / 2</i>	<i>78</i>
<i>Tabela 7:</i>	<i>Tabela nalog - obstoječe stanje 1/4.....</i>	<i>79</i>
<i>Tabela 8:</i>	<i>Prenovljena tabela aktivnosti 1 / 2.....</i>	<i>85</i>
<i>Tabela 9:</i>	<i>Prenovljena tabela aktivnosti 2/2.....</i>	<i>86</i>
<i>Tabela 10:</i>	<i>Prenovljena tabela nalog 1 / 4</i>	<i>87</i>
<i>Tabela 11:</i>	<i>Tabela operacij 1 / 4.....</i>	<i>96</i>

1. UVOD

V današnjem svetu so podjetja življenjsko odvisna od svojih informacijskih sistemov, ki morajo zagotavljati pravočasne in kakovostne informacije za ustrezno poslovanje organizacije. Tako je zagotavljanje informacijskih sistemov, ki so se sposobni prilagajati vedno novim zahtevam, ki jih postavlja poslovno okolje in uporabniki sistema, bistvenega pomena pri ohranjanju uspešnosti organizacije.

Za razvoj informacijskih sistemov, ki postajajo vedno bolj kompleksni, so v zadnjih desetletjih nastale številne metodologije, katerih namen je vodenje razvijalcev skozi predpisane faze, opredelitev potrebnih tehnik, ki se uporabljajo v posameznih fazah razvoja in zagotavljanje nadzora nad projektom izgradnje informacijskega sistema tako v časovnem kot tudi vsebinskem vidiku. Metodologije so do pojava objektno orientiranih metodologij ločevale podatkovni in procesni vidik. S pojavom objektno orientiranih metodologij se podatkom priključijo operacije za manipulacijo z njimi tako, da je kontrola nad spreminjanji podatkovnih struktur zagotovljena na enem mestu – znotraj objekta. V zadnjem desetletju so različne objektno orientirane metodologije skozi procese usklajevanj in standardizacije prerasle v UML (poenoten jezik za modeliranje, angleško unified modelling language), ki skupaj s priporočenim procesom izgradnje informacijskih sistemov predstavlja trenutno najbolj razširjeno objektno usmerjeno metodologijo. Alternativni pristop ponuja metodologija »Tabelarni razvoj aplikacij – TAD«, ki namesto množice različnih diagramov predstavlja obravnavani sistem v okviru posebnih tabel, ki omogočajo bolj zgoščen pogled na obravnavano problematiko. Zaradi tega in dejstva, da metodologija TAD zelo dobro vodi analitika skozi faze izgradnje informacijskega sistema, vključuje pa tudi posebno fazo prenovе poslovnih procesov, sem metodologijo TAD izbral za osrednji predmet obravnave v magistrskem delu.

Metodologijo nameravam preizkusiti na praktičnem primeru informacijskega sistema logistike v distribucijskem centru podjetja Poslovni sistem Mercator, d.d., ki ga je treba temeljito prenoviti zaradi dodajanja novih funkcionalnosti, ki bistveno posegajo v delovanje sistema, tako da je potrebna podrobna analiza potrebnih sprememb v poslovnem procesu ter delovanju informacijskega sistema.

1.1. Namen dela

Namen magistrskega dela je, s pomočjo domače in tuje strokovne literature, preučiti metodologije za objektno načrtovanje informacijskega sistema. Poseben poudarek bo na proučevanju in uporabi TAD metodologije. V okviru naloge bom prikazal problematiko prenovе poslovnih procesov, saj bo v okviru praktičnega primera prišlo do sprememb delovnih procesov.

Glavni namen naloge je popisati vse aktivnosti in delovne procese, ki nastopajo v distribucijskem centru v okviru logističnega procesa. Procese bom opisal s pomočjo metodologije TAD. V nadaljevanju nameravam z uporabo omenjene metodologije opredeliti prenovljene delovne procese znotraj poslovnih procesov, ki potekajo v okviru distribucijskega centra. Tako prenovljeni procesi bodo predstavljali osnovo za izdelavo prenovljenega informacijskega sistema.

1.2. Cilji dela

Cilji, ki jih želim doseči v okviru naloge, izhajajo iz namena naloge, in so:

- Predstaviti osnovne faze razvoja informacijskega sistema.
- Predstaviti nekatere metodologije izgradnje informacijskega sistema in UML kot najbolj razširjeno objektno usmerjeno metodologijo.
- Podrobno predstaviti metodologijo TAD.
- Z uporabo metodologije TAD analizirati obstoječi proces logistike, identificirati nujne spremembe, prenoviti proces kot celoto in pripraviti načrt informacijskega sistema.
- V okviru delne implementacije ugotoviti, kako lahko z uporabo objektno orientirane metodologije načrtujemo prenovo obstoječega informacijskega sistema.

1.3. Metode dela

Magistrsko delo je zasnovano tako, da na praktičnem primeru razvoja informacijskega sistema logistike uporabimo predstavljena teoretična spoznanja. Pri analizi in načrtovanju sistema bom uporabil objektno usmerjeno metodologijo TAD.

Delo v prvih petih poglavjih obsega predstavitev teoretičnih spoznanj s področja izgradnje informacijskih sistemov in prenove poslovnih procesov. Podrobno bo predstavljena metodologija TAD.

Za pripravo teoretičnih izhodišč je bila preučena domača in tuja strokovna literatura iz navedenih področij in interni viri podjetja, zbrani v obliki organizacijskih predpisov in delovnih navodil, v okviru sistema kakovosti ISO 9001. Poleg tega so bili opravljeni pogovori z odgovornimi ljudmi v podjetju in uporabljene izkušnje avtorja, ki že dalj časa aktivno sodeluje pri informacijski podpori obravnavanega procesa.

V šestem poglavju je prikazan praktični primer uporabe metodologije TAD pri razvoju informacijskega sistema logistike. V okviru implementacije se delo omejuje na načrtovanje potrebnih sprememb obstoječega informacijskega sistema.

1.4. Struktura dela

Delo je poleg uvoda razdeljeno v osem poglavij in prilogo.

Prvo poglavje podaja kratek uvod.

V drugem poglavju so opredeljeni osnovni pojmi in opisan tradicionalni pristop k izgradnji informacijskih sistemov, njegove slabosti in nekatere metodologije, ki te slabosti odpravljajo.

Tretje poglavje je namenjeno predstavitvi področja prenove poslovnih procesov, opredelitvi potrebe po korenitem spreminjanju poslovnih procesov in načinu, na katerega se prenova izvaja.

V četrtem poglavju so predstavljeni osnovni koncepti objektnega pristopa izgradnje informacijskih sistemov in UML, skupaj s priporočenim procesom razvoja informacijskega

sistema, kot najbolj razširjena metodologija za objektni način izgradnje informacijskih sistemov.

V petem poglavju je podrobno predstavljena metodologija »Tabelarni razvoj aplikacij«, ki s pomočjo uporabe posebnih tabel predstavlja bistveno drugačen pristop k izgradnji informacijskega sistema.

V šestem poglavju je metodologija TAD praktično uporabljena na primeru prenove informacijskega sistema logistike v podjetju Poslovni sistem Mercator, d.d.

V sedmem poglavju so navedene sklepne ugotovitve in predlogi za nadaljnje delo.

Na koncu dela sledita seznama uporabljene literature in virov ter priloga. V prilogi so v celoti podane tabele nalog in aktivnosti ter procesni model, ki so deloma prikazani v okviru šestega poglavja.

Osnovna hipoteza dela je, da je mogoče z uporabo metodologije TAD pregledno popisati obstoječe poslovne procese, jih spremeniti v skladu z novo ugotovljenimi cilji organizacije in na osnovi spremenjenih procesov načrtovati izgradnjo informacijskega sistema tako, da bo možno na osnovi načrta zgraditi nov informacijski sistem kot tudi opredeliti spremembe, ki so potrebne v obstoječem informacijskem sistemu logistike.

2. O RAZVOJU INFORMACIJSKIH SISTEMOV

V začetni fazi uporabe računalnikov v poslovne namene so imeli računalniki glede na današnji čas majhno zmogljivost, zato je bilo težišče raziskovanja v iskanju učinkovitih programskih algoritmov, ki so optimalno izkoriščali razpoložljive zmogljivosti v okviru proizvedenih programskih rešitev (angleško: applications) omejenega obsega. Take programske rešitve so bile po vsebini omejene, izdelovali pa so jih programerji, katerih znanja so bila predvsem tehnične narave, manj pa so poznali poslovne probleme, ki jih je bilo treba rešiti. Poleg tega so jim pogosto manjkale tudi komunikacijske sposobnosti. Uporabniki so tako imeli težave pri prenašanju svojih potreb programerjem (Avison, Fitzgerald, 1996, str.18). Posledično so se pojavili problemi:

- Rešitve pogosto niso ustrezno zadovoljevale uporabniških zahtev.
- Četudi je bila rešitev ustrezna, ker je programer zelo dobro poznal problem in strukturo rešitve, je bilo izvajanje sprememb v kolikor toliko obsežni rešitvi dolgotrajno in zamudno zaradi posledic, ki so se lahko pojavile v drugem delu sistema.
- Rešitve so bile pogosto slabo dokumentirane.

Iz navedenih razlogov so podjetja kmalu postala odvisna od posameznikov, ki so podrobno poznali strukturo rešitev, saj so bile le-te za druge programerje nepregledne zaradi pomanjkanja dokumentacije in uporabe različnih tehnik programiranja.

Kot odgovor na navedene probleme se je pojavila nadaljnja delitev dela pri izgradnji informacijskih sistemov¹:

- analiza poslovnih zahtev – opravi jo sistemski analitik,
- načrtovanje sistema, ki naj izpolni definirane poslovne zahteve,
- programiranje.

Tako se poleg same implementacije pojavita še dve pomembni fazi v razvoju informacijskih sistemov; to sta analiza in načrtovanje (angleško: design). Naslednja reakcija na opisane probleme je pojav formaliziranih metodologij za razvoj računalniških aplikacij (Avison, Fitzgerald, 1996, str. 20).

Metodologija je po definiciji skupek postopkov, tehnik, metod, ki jih uporabljamo pri reševanju nekega problema. Pod pojmom »metodologija gradnje informacijskih sistemov« si vsaj v praksi največkrat predstavljamo organizacijsko-tehnično znanje, ki ga uporabljamo pri zasnovi in izdelavi računalniških rešitev. (Kovačič, Vintar, 1994, str. 29).

Metodologija za razvoj informacijskih sistemov je zbirka postopkov, tehnik, orodij in dokumentacijskih pripomočkov, ki bo pomagala razvijalcem sistema v njihovih naporih za implementacijo novega informacijskega sistema. Metodologija se sestoji iz faz, ki se nadalje sestojijo iz podfaz, ki vodijo razvijalce sistema v njihovi izbiri tehnik, ki bi bile lahko

¹ Informacijski sistem lahko definiramo kot sistem, ki zbira, hrani, obdeluje in dostavlja informacije, relevantne za organizacijo tako, da je informacija dostopna in uporabna tistim, ki jo želijo uporabiti, vključujoč managerje, osebje, stranke in državljane. Informacijski sistem je sistem človeških aktivnosti, ki lahko vsebuje, ali pa tudi ne, uporabo računalniških sistemov. (Buckingham, 1987, povzeto iz Avison, Fitzgerald, 1996, str. 13).

primerne v vsaki fazi projekta in jim pomagajo načrtovati, upravljati, nadzirati in ocenjevati projekte izgradnje informacijskih sistemov (Avison, Fitzgerald, 1996, str. 10). Metodologija tipično vsebuje predpisane korake razvoja, tehnike za pripravo izdelkov, orodja, integrirana z metodologijo, in vzorce, ponovno uporabne dokumente in sezname za preverjanje² (Murch, 2001, str. 139).

Namen uporabe metodologije pri izgradnji informacijskih sistemov je doseganje različnih ciljev, ki naj jih uporaba metodologije omogoča (Avison, Fitzgerald, 1996, str. 11):

- Natančno beleženje informacijskih zahtev. Metodologija naj pomaga pri opredeljevanju in analizi uporabniških zahtev.
- Podana mora biti sistematična metoda razvoja tako, da lahko napredek učinkovito nadzorujemo. Nadzor je posebnega pomena pri projektih velikega obsega, zato so nujne kontrolne točke in dobro definirane faze, ki omogočajo uporabo tehnik načrtovanja projektov.
- Metodologija naj omogoča razvoj informacijskega sistema v primernem časovnem okviru in ob sprejemljivih stroških.
- Izdelan informacijski sistem naj bo dobro dokumentiran in enostaven za vzdrževanje. Potreba po prihodnjih spremembah sistema je neizbežna zaradi sprememb v organizaciji in okolju. Spremembe naj bodo izvedljive s čim manjšim vplivom na preostali del sistema.
- Zagotovljena mora biti indikacija potrebnih sprememb v čim zgodnejši fazi razvojnega procesa. Strošek spremembe se povečuje, ko izgradnja sistema napreduje od začetne faze analize proti implementaciji sistema.
- Izdelan informacijski sistem naj bo všečen ljudem, na katere vpliva. Tako je zagotovljena uspešnost uporabe takega informacijskega sistema.

Kot tipičen primer metodologije, ki naj izpolnjuje navedene zahteve, je v nadaljevanju opisana metodologija tradicionalnega življenjskega cikla razvoja informacijskega sistema.

2.1. Tradicionalni življenjski cikel razvoja sistema³

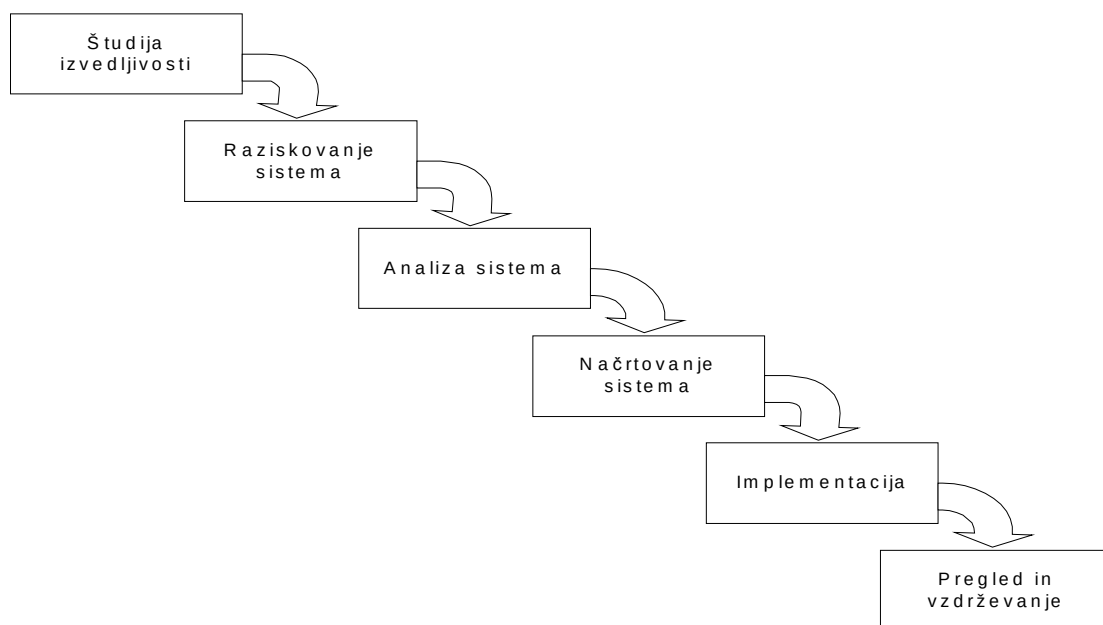
Metodologija je nastala konec šestdesetih let v Veliki Britaniji in predstavlja osnovno izhodišče mnogih kasnejših metodologij. Že sama uporaba besedne zveze »življenjski cikel« nas opozarja na iterativno naravo procesa razvoja informacijskih sistemov. Zaradi sprememb v organizaciji in okolju se dopolnjujejo uporabniške zahteve, kar sproži nov cikel razvoja – predelave informacijskega sistema.

² Checklists

³ V literaturi srečamo sopomenke: tradicionalna sistemska analiza, SDLC-systems development life cycle, waterfall modell, kaskadni model, stopničasti model, linearni pristop.

Metodologija predvideva naslednje faze:

Slika 1: Tradicionalni model življenjskega cikla sistema



Vir: Jacobson et al., 1994, str.71

V fazi študije izvedljivosti (angleško: feasibility study) je treba analizirati obstoječi informacijski sistem, nove informacijske zahteve, probleme obstoječega sistema in nove tehnološke možnosti. Pripravijo se alternativne rešitve, ki se ocenijo po pravno formalni, organizacijski, tehnični in ekonomski plati. Opredelijo se meje sistema, potrebni viri in opredelijo stroški in koristi posameznih alternativ. Izbere se najboljša rešitev in se predlaga vodstvu v obliki formalnega poročila. Odobritev poročila s strani vodstva in odločitev o nadaljevanju projekta predstavljata prvo kontrolno točko metodologije.

Ko je nadaljevanje projekta odobreno s strani vodstva, je treba v fazi raziskovanja sistema podrobno analizirati zahteve uporabnikov in pripraviti funkcionalno specifikacijo sistema. Potrebno je podrobno opredeliti tipe in količino podatkov, omejitve in probleme obstoječega sistema. Informacije pridobimo s pomočjo intervjujev vodilnega in operativnega kadra, z uporabo vprašalnikov, analizo obstoječih dokumentov in opazovanjem načinov dela. Orodja za prikazovanje rezultatov so diagrami poteka, hierarhični diagram strukture poročanja v organizaciji, specifikacije obstoječih dokumentov, zapiski intervjujev.

Na podlagi zbranih dejstev preidemo v fazo analize sistema, v kateri poskušamo odkriti vzroke obstoječih problemov, razloge za uporabo določenih metod, obstoj alternativnih metod, predviden porast količine podatkov (Avison, Fitzgerald, 1996, str. 23). Poskušamo razumeti razloge za trenutno stanje sistema in nakazati načine za izboljšanje.

Fazi analize sledi faza načrtovanja sistema, v kateri je treba narediti načrt bodočega sistema. Potrebno je opredeliti podatkovne strukture – zapise, načine za zajemanje podatkov, procese za transformacijo vhodnih podatkov v izhodne, procedure za varovanje in restavriranje

podatkov. Opravi se odločitev o izboru konkretnega okolja implementacije, v smislu izbora strojne in sistemske programske opreme ter sistema za upravljanje z bazami podatkov, če ga uporabimo. Postopkovni del se dokumentira v obliki diagramov poteka (angleško: flowcharts), mrežni diagram se uporabi za prikaz povezav med procesi in podatki – kateri programi uporabljajo določene podatke.

Sledi faza implementacije, ki vsebuje nakup in instalacijo opreme, kodiranje in testiranje programov ter kreiranje dokumentacije. Nadalje je treba izvesti integralni test sistema tudi s strani uporabnikov ob uporabi večje količine (realnih) podatkov, da preizkusimo tako funkcionalno kot tudi performančno ustreznost sistema. Izvesti je treba še šolanje uporabnikov sistema in izvesti prehod iz starega v novi sistem⁴.

Ko sistem nekaj časa operativno deluje, izvedemo pregled sistema (angleško: review), v katerem preverimo skladnost delovanja sistema z opredeljenimi zahtevami in primerjamo dejanske stroške s stroški, predvidenimi v študiji izvedljivosti. Pripraviti je treba tudi formalno poročilo. V okviru vzdrževanja se odpravljajo napake, ki so bile vgrajene v sistem, ter dograjujejo funkcionalnosti, ki so potrebne zaradi novih zahtev uporabnikov, ali spremenjenega okolja. V kolikor so potrebne spremembe prevelike, da bi jih lahko izvedli brez večjih posegov v sistem, sprožimo nov cikel, ki ga začnemo s fazo študije izvedljivosti.

Kot še mnogo drugih metodologij tudi tradicionalni model življenjskega cikla sistema temelji na nekaterih predpostavkah, ki v realnosti niso vedno izpolnjene. Navajam dve skupini predpostavk (Kovačič, Vintar, 1994, str. 41)

- Predpostavke glede IS
 - o IS je samostojen, zaključen sistem, katerega funkcije lahko vnaprej določimo na osnovi analize konkretnega sistema.
 - o Tudi če je sistem zelo obsežen, morda decentraliziran, se predpostavlja, da so vsi deli sistema v skladu z neko globalno shemo in konsistentni med seboj. To pomeni, da lahko predvidimo pravila obnašanja vseh delov sistema v vsakem trenutku – deterministična koncepcija sistema.
 - o Sedanje in bodoče informacijske potrebe organizacije so znane in vnaprej določljive.
- Predpostavke glede procesov v okviru IS
 - o Pomembno je analizirati obstoječe in predvidene aktivnosti organizacije, da dobimo zaključeno množico procesov, ki nastopajo v okviru IS.
 - o V okviru IS prevladujejo rutinski, strukturirani procesi, ki jih je možno definirati vnaprej.
 - o Proces v okviru IS so razmeroma stabilni in se v toku življenjske dobe IS le malo spreminjajo.

⁴ Angleški izraz: cutover

Glede na opisane predpostavke in cilje, ki naj jih izpolnjuje metodologija, opredeljene v začetku poglavja, ima tradicionalni model življenjskega cikla sistema naslednje lastnosti:

- Razvoj poteka zaporedno po fazah, naslednja faza se prične po formalni odobritvi rezultata predhodne faze.
- Vsaka razvojna faza je natančno opredeljena in dokumentirana, definirane so diagramске tehnike.
- Uporabljene so tehnike za nadzor izvajanja projekta, tako po časovni kot stroškovni plati.

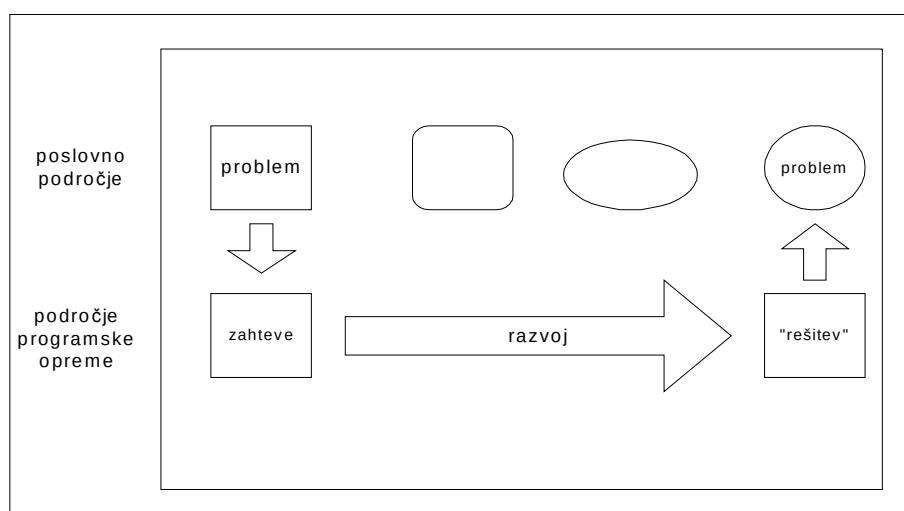
V kolikor bi predpostavke metodologije bile vedno izpolnjene, bi njena uporaba zagotavljala doseganje zahtevanih ciljev.

V realnosti se srečujemo z vedno večjo hitrostjo sprememb poslovnega okolja, ki narekuje prožen odziv podjetij in posledično kontinuirano spreminjanje informacijskih sistemov. Tako lahko ugotovimo naslednje:

- Bodočih informacijskih potreb ni mogoče opredeliti vnaprej.
- Procesi se v okviru življenjskega cikla sistema lahko bistveno spremenijo.
- Zahteve, opredeljene na začetni fazi razvoja informacijskega sistema, ne ustrezajo več dejanskim potrebam poslovnega procesa.
- Zaradi zaporedne izvedbe faz razvojnega cikla in velike stopnje formalizacije je hitrost razvoja novega informacijskega sistema premajhna glede na hitrost sprememb poslovnega okolja in posledično pričakovanih funkcionalnosti sistema.

Grafično lahko prikažemo problem na naslednji način:

Slika 2: Problem konvencionalnega načina razvoja programske opreme



Vir: Taylor, 1995, str. 8

Čas razvoja kolikor toliko kompleksnega informacijskega sistema se pri uporabi tradicionalne metodologije in z dosledno zaporednostjo posameznih faz lahko meri v letih, kar ob današnji hitrosti sprememb poslovnega okolja povzroča neustreznost izdelanih rešitev.

Posledice navedenih dejstev se kažejo v naslednjih simptomih:

- Nezadovoljstvo uporabnikov, saj sistem ne rešuje današnjega problema.
- Pomanjkljivosti sistema se pokažejo šele ob predaji sistema uporabnikom, saj ti ne sodelujejo v fazah analize, načrtovanja in implementacije, njihova vloga je omejena na podajanje zahtev na začetku in sprejem rešitve na koncu procesa razvoja IS. Odpravljanje napak na koncu razvojnega cikla pa je najdražje.
- Zaostanek pri razvoju potrebnih uporabniških rešitev (angleško: application backlog). Ker se zahteve poslovnega okolja spreminjajo hitreje, kot je z uporabo tradicionalne metodologije mogoče razvijati rešitve, je zaostanek neizbežen.
- Velika količina potrebnega vzdrževanja obstoječih sistemov. Ker je razvoj novih sistemov prepočasen, se najbolj nujne zahteve poskušajo vgraditi v obstoječe sisteme »na hitro«. Tako se v želji po hitrosti odziva žrtvuje kakovost opravljenega načrtovanja. Končni rezultat takega načina dela je informacijski sistem, ki postane nemogoč za vzdrževanje. Tako se za vzdrževanje obstoječih sistemov porabi tudi do 80% časa in na ta način skoraj ni razvoja novih rešitev.

Poleg navedenih problemov informacijski sistemi, razviti na ta način, večinoma ne zadovoljujejo potreb vrhnjih vodilnih struktur podjetja. V fazi opredelitev potreb se podrobno analizirajo zahtevani izhodi sistema, obstoječi sistem in funkcionalnost, kot jo vidijo uporabniki. Rešitev naj pokrije funkcionalnost posameznega področja, pogosto se avtomatizirajo obstoječi procesi. Tako je za zadovoljitev informacijskih potreb vodstva potrebno poseganje v celoten spekter rešitev. Zaradi neambicioznega informatiziranja obstoječih procesov brez njihove prenove, ki jo lahko omogoča uporaba informacijske tehnologije, je končni rezultat neoptimalen.

Če želimo izboljšati kakovost programskih rešitev, je treba:

- povečati udeležbo uporabnikov v celotnem procesu razvoja sistema,
- skrajšati razvojni cikel,
- omogočiti sprotno odpravljanje ugotovljenih napak z možnostjo vračanja v predhodne faze razvojnega cikla.

Kakovost programske rešitve definiramo kot čimbolj učinkovito zadovoljevanje poslovnih oziroma uporabniških potreb v času uvedbe sistema (Martin, 1991, str.4).

Za reševanje navedenih problemov so se pojavili različni pristopi in metodologije ter drugačni modeli življenjskega cikla sistema.

2.2. Uporaba prototipov

Prototip je približek sistema, ki ga želimo zgraditi. Vsebuje glavne funkcionalnosti zelenega sistema. Izdelava prototipov na področju razvoja programske opreme je omogočena s pojavom programskih orodij in generatorjev rešitev (angleško: application generators).

Uporaba prototipov naslavlja nekatere probleme tradicionalnega pristopa in še posebej povečuje udeležbo uporabnikov v procesu nastajanja rešitve. Izdelani prototipi niso le shematski prikazi, ampak dejanski izpisi na papirju in ekranske slike.

Uporaba prototipov tako izboljšuje faze raziskave, analize in načrtovanja sistema. Še posebej je uporabna v primerih, ko (Avison, Fitzgerald, 1996, str. 77):

- Področje rešitve ni dovolj dobro definirano.
- Nova tehnologija, ki je potrebna za izdelavo rešitve, še ni dovolj dobro poznana v organizaciji.
- Komunikacija med uporabniki in analitiki ni dovolj dobra.
- Strošek zavrnitve rešitve s strani uporabnikov bi bil visok in je zagotovitev pokritja uporabniških potreb bistvenega pomena.
- Potrebna je ocena vpliva bodočega informacijskega sistema.

Prototip mora vsebovati naslednje lastnosti oziroma mora biti (Kovačič, Vintar, 1994, str. 238) :

- dovolj majhen za izvedbo,
- dovolj vpliven in prepričljiv,
- dovolj poceni in dolgoročen,
- dovolj nujen za izvedbo.

Prototipe se navadno gradi s pomočjo posebnih orodij, kot so generatorji ekranov in poročil, ki omogočajo hitro izdelavo ekranov in poročil. Po potrebi se ti izdelki tudi hitro spreminjajo, dokler ni dosežena ustrezna stopnja zadovoljstva uporabnikov. Namen prototipa je praktična predstava izgleda in funkcionalnosti bodoče rešitve. Rezultati, pridobljeni pri obravnavi prototipa se lahko uporabijo v (Avison, Fitzgerald, 1996, str. 77):

- fazi analize za razumevanje obstoječega sistema in predlog funkcionalnih zahtev alternativnega sistema,
- fazi izgradnje prototipa za ocenjevanje s strani uporabnikov,
- več ponavljajočih se fazah spreminjanja in ocenjevanja prototipa,
- fazah načrtovanja in razvoja novega sistema, kjer je prototip uporabljen kot del specifikacije.

Običajno prototip ni namenjen nadaljnji dodelavi in se po dokončanju novega sistema zavrže. Alternativni pristop je uporaba prototipa kot osnove bodočega operativnega sistema. V okviru prototipnega pristopa⁵ se prototipu v več iteracijah dodajajo in spreminjajo funkcionalnosti, dokler ne ustreza potrebam uporabnika. V tej fazi se prototip preda v operativno rabo. Pri

⁵ Prototipni pristop ali pristop prototipa predstavlja postopno razvijanje računalniške programske rešitve z neposrednim sodelovanjem njenega razvijalca in uporabnika; postopno izdelovanje in preizkušanje modela rešitve po načelih pristopa od grobega k podrobnemu; priporočljiv pri razvoju računalniških rešitev za podporo procesa upravljanja; predpostavlja uporabo določene tehnologije (programskih jezikov, baz podatkov) in določeno stopnjo razvitosti uporabe računalnika v organizaciji (Kovačič, Vintar, 1994, str. 237).

izdelavi takega prototipa je treba posebej paziti na dobro načrtovanje. Prototip mora biti sposoben obvladovati količine podatkov, ki so predvidene v nadaljnji operativni rabi. Potrebno je tudi ustrezno dokumentiranje sistema, tako zaradi prihodnjih sprememb kot tudi za pravilno uporabo s strani uporabnikov. Kadar je prototip namenjen kasnejši operativni rabi, je izjemnega pomena kontrola nad procesom razvoja, da se izognemo iskanju bližnjic, ki neizbežno povzročijo težave pri nadaljnji uporabi tako nastale rešitve.

2.3. RAD

Metodologija, ki učinkovito uporablja vgradnjo prototipov v končno rešitev se imenuje RAD⁶. RAD je oblika razvojnega cikla, načrtovanega za mnogo hitrejši razvoj in izdelke večje kakovosti, kot tradicionalni življenjski cikel (Martin, 1991, str.2). Za doseg tako opredeljenih ciljev RAD upošteva naslednje dejavnike, ki izboljšujejo kakovost izdelanega sistema:

- Močno udeležbo uporabnika pri načrtovanju sistema.
- Uporabo prototipov, ki uporabnikom omogočajo predstavo in popravljanje sistema.
- Uporabo integriranih CASE⁷ orodij, ki zagotavljajo tehnično integriteto pri modeliranju in načrtovanju sistema.
- CASE repozitorij, ki omogoča ponovno uporabo preizkušenih vzorcev in komponent.
- Integrirano CASE orodje, ki omogoča generiranje kode iz preizkušenega načrta.
- Vključevanje uporabnikov v fazi konstrukcije, ki po potrebi omogoča korekcijo podrobnosti.

Celoten koncept temelji na majhnih skupinah visoko usposobljenih razvijalcev, ki z uporabo integriranega CASE orodja ob sodelovanju uporabnikov hitro izdelajo ustrezno programsko rešitev. Skupina razvijalcev mora biti majhna, saj se v večjih skupinah veliko časa porabi za usklajevanje in komunikacijo med člani skupine. V primeru večjih projektov je predvidena delitev projekta na manjše dele, ki jih lahko obvladuje posamezna skupina in se izvajajo vzporedno, v več skupinah. Pri tem je ključnega pomena kakovost CASE orodja, saj se integriteta projekta zagotavlja skozi uporabo skupnega podatkovnega in procesnega modela. RAD lahko nastopa kot samostojna metodologija, ali pa v okviru infrastrukture metodologije informacijskega inženirstva, ki bo opisana v nadaljevanju.

Metodologija RAD ima naslednje faze:

- načrtovanje potreb,
- uporabniška zasnova sistema,
- konstrukcija sistema,
- uvajanje sistema.

Kritični element v vseh fazah je omejen čas trajanja projekta. RAD projekti naj ne bi trajali dlje od 90 dni. Dosledno spoštovanje tega načela omogoča orientacijo na najbolj bistvene

⁶ Rapid Application Development; hitri razvoj aplikacij

⁷ Computer Aided Software Engineering

funkcionalnosti, ki jih mora novi sistem imeti, in že v fazi zbiranja zahtev izloči funkcije, ki niso nujno potrebne.

V fazi načrtovanja potreb je predvidena uporaba tehnike skupnega planiranja potreb JRP⁸, ki predvideva udeležbo ključnih ljudi v organizaciji, ki poznajo obravnavani poslovni proces in so pooblaščen za sprejemanje odločitev glede zahtev bodočega sistema. Predstavnik vodstva (angleško: executive owner), zavezan izdelavi sistema, je nujni pogoj za uspešnost JRP delavnice. Opredeljevanje zahtev poteka v okviru delavnice, kjer moderator zgolj usmerja udeležence proti nedvoumnemu opredeljevanju zahtev. Zapisovalec sproti zapisuje vse, kar je mogoče, v CASE orodje. Tipični izdelki JRP delavnice so (Martin, 1991, str.149):

- spisek oddelkov in lokacij, ki jim je novi sistem namenjen,
- spisek ciljev sistema,
- podrobnosti posameznih funkcij sistema,
- diagram dekompozicije procesa,
- diagram toka procesa,
- diagram vmesnikov z drugimi sistemi,
- spisek nerešenih zadev,
- ciljni datum implementacije,
- naslednji korak.

V primeru nasprotujočih si pogledov so udeleženci pooblaščen, da se o rešitvi dogovorijo v okviru delavnice. Opredelitev zahtev sistema se lahko izvede v le nekaj ponovitvah delavnice.

Ko je odobren prehod v fazo uporabniške zasnove sistema, se organizira delavnica JAD⁹. V delavnici sodelujejo uporabniki bodočega sistema, informatiki, gostujoči svetovalci. Običajno traja delavnica pet delovnih dni, brez motenj iz okolja. Odločilna je vloga uporabnikov, informatiki prevedejo njihove zahteve v ustrezne diagrame in po potrebi prototipne izdelke. V okviru delavnice se podrobno opredelijo procesi, načrtujejo podatkovne strukture in podrobno opredelijo poslovna pravila. Kot orodja za modeliranje se uporabijo entitetno-relacijski diagram, diagram funkcionalne dekompozicije, diagram toka podatkov in akcijski diagram. Vsi rezultati se beležijo v CASE orodju, ki omogoča preverjanje konsistentnosti različnih diagramov. Udeleženci podrobno pregledajo povezave med različnimi modeli.

Delavnica JAD učinkovito zamenjuje tradicionalni način opredeljevanja zahtev, kjer analitik opravlja intervjuje s posameznimi nosilci funkcij in uporabniki ter naknadno usklajuje nerešene zadeve, kar lahko traja zelo dolgo. Prisotnost pooblaščenih ljudi omogoča odločanje o projektu znotraj delavnice, tako je tudi dosežena večja preglednost odločanja.

Predstavnik vodstva, ki je zavezan projektu, učinkovito odstranjuje birokratske in politične ovire, ki bi lahko zadržale izvajanje projekta. Tehniki JRP in JAD odločno skrajšata čas, ki je

⁸ Joint Requirements Planning, skupno planiranje potreb

⁹ Joint Application Design, skupno načrtovanje rešitev

potreben za definiranje zahtev in podrobno opredeljevanje sistema, kar omogoča kratek cikel razvoja informacijskega sistema.

V fazi konstrukcije sistema se načrt iz predhodne faze dopolni do podrobnosti, nato se s pomočjo orodja generira programska koda. Tu je ključnega pomena dobro CASE orodje, ki vključuje generator kode. Vnosne maske in izgled transakcij se izvede v obliki prototipa, ki ga potrdijo uporabniki. V kolikor gre za večja odstopanja, se vrnemo v predhodno fazo, sicer pa se rešitev izpopolni v seriji iteracij do končne rešitve. S pomočjo orodja za generiranje rešitev se izvede tudi dokumentiranje. V tej fazi se izvedeta tudi testiranje celotne rešitve in optimizacija. Pred prehodom v produkcijo lahko tudi preizkusimo prototipno rešitev, ki obsega polno funkcionalnost, uporabimo pa jo v omejenem obsegu, npr. za manjše število uporabnikov.

V fazi prehoda je treba zagotoviti podatke na ročni ali elektronski način. V primeru prepisa podatkov iz starega sistema je treba pripraviti ustrezne postopke za konverzijo in polnjenje podatkov. Sledi priprava in izvedba končnega testiranja, ki obsega (Martin, 1991, str. 333) :

- testiranje programov in transakcij glede pravilnosti izvedbe transakcij in ravnanja v primeru odpovedi¹⁰,
- integralni test sistema preverja delovanje sistema kot celote,
- test povezljivosti z ostalimi sistemi¹¹,
- obremenilni test,
- test sprejemljivosti za uporabnike.

Po uspešnem testiranju izvedemo šolanje uporabnikov, instaliramo sistem v produkcijo in napolnimo podatke preko ustreznih postopkov konverzije. Po začetku delovanja analiziramo delovanje sistema, kjer opredelimo potrebo po ugaševanju sistema, dodatnem šolanju uporabnikov in potrebnih spremembah rešitve.

Pri oceni metodologije RAD je potrebno poudariti kritično odvisnost metodologije od odličnega CASE orodja, ki omogoča tovrsten način dela. Poleg tega je problematična izdelava večjih sistemov, saj je kljub skupnemu repozitoriju modelov v praksi težko uskladiti delovanje neodvisnih skupin razvijalcev, ki delujejo pod časovnim pritiskom. Tako RAD svoje prednosti pokaže na projektih, kjer (Yeates, Cadle, 1996, str. 52) :

- rešitev ni preveč kompleksna in je obseg dobro opredeljen,
- je okolje organizacije zrelo in stabilno,
- se na projektu ne uporablja nova, nepoznana tehnologija,
- se rešitev dobro vklaplja v obstoječo infrastrukturo.

Tako je možno ob uporabi obstoječe infrastrukture, ki olajšuje integracijo, ponovno uporabo komponent in v dobrem poznavanju organizacije in tehnologije, s katero delamo resnično dosegati veliko hitrost razvoja novih rešitev.

¹⁰ angleško: transaction crash test

¹¹ angleško: intersystem integration test

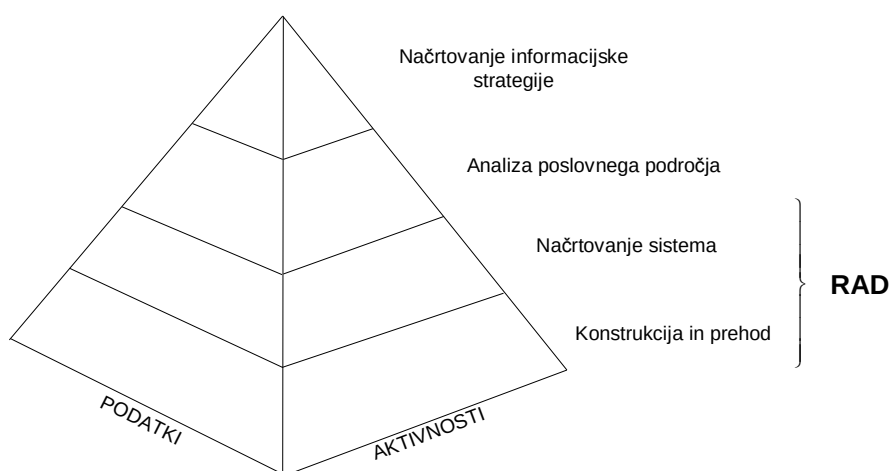
2.4. Informacijsko inženirstvo

Informacijsko inženirstvo¹² je metodologija, ki celovito obravnava problematiko razvoja informacijskih sistemov v podjetju, razdeljeno na štiri ravni:

- načrtovanje informacijske strategije¹³,
- analizo poslovnega področja¹⁴,
- načrtovanje sistema¹⁵,
- konstrukcija in prehod¹⁶.

Predstavlja uporabo trdno spetih formalnih tehnik za planiranje, analizo, načrtovanje in konstrukcijo informacijskih sistemov na ravni celotnega podjetja, ali pa preko glavnih področij podjetja (Martin, 1989, str. 1). Zaradi celovite obravnave in kompleksnosti problematike, ki jo v podjetjih pokrivajo informacijski sistemi, učinkovito planiranje, analiza, načrtovanje in konstrukcija informacijskih sistemov na ravni celotnega podjetja, niso mogoči brez uporabe avtomatiziranih orodij. Tako lahko informacijsko inženirstvo definiramo kot niz trdno spetih avtomatiziranih tehnik, s katerimi so modeli podjetja, podatkovni in procesni modeli predstavljeni v enovitem repozitoriju in uporabljeni za kreiranje in vzdrževanje sistemov za procesiranje podatkov (Martin, 1991, str. 348). Ravni metodologije lahko grafično predstavimo v obliki piramide.

Slika 3: Piramida informacijskega inženirstva



Vir: Martin, 1991, str. 351

Na prikazani sliki so vidne vse štiri ravni metodologije in avtorjev predlog, da se faze načrtovanja in konstrukcije sistema ter prehoda izvedejo v okviru metodologije RAD. Osnovni gradniki metodologije so podatkovni in procesni model (podatki in aktivnosti) ter povezava med njima, navadno v obliki matrike, ki prikazuje interakcijo posameznih procesov

¹² angleško: information engineering

¹³ angleško: information strategy planning

¹⁴ angleško: business area analysis

¹⁵ angleško: system planning and design

¹⁶ angleško: construction and cutover

in entitet. Osnovna ideja metodologije temelji na principu delitve problema na manjše obvladljive dele, ki jih lahko v okviru enovitega modela, ki ga razvijamo v repozitoriju integriranega CASE orodja, neodvisno razvijamo in implementiramo. Najprej je treba opredeliti informacijsko strategijo.

2.4.1. Načrtovanje informacijske strategije

Namen načrtovanja informacijske strategije je priprava informacijske arhitekture in strategije, ki podpirata strateške cilje in informacijske potrebe organizacije (Avison, Fitzgerald, 1996, str. 284). V okviru načrtovanja opredelimo poslovna področja in različne informacijske sisteme ter način njihove integracije v okviru enotnega modela podjetja. Pri načrtovanju informacijske strategije je nujno sodelovanje najožjega vodstva podjetja, saj je treba uskladiti poslovne cilje podjetja s cilji, ki jih opredeljuje informacijska strategija. Pri načrtovanju informacijske strategije je potrebno izdelati dve vrsti študij (Martin, 1991, str. 357) :

- Opredelitev strateških priložnosti, ciljev, kritičnih dejavnikov uspeha¹⁷ in informacijskih potreb različnih delov podjetja.
- Opredelitev načina, s katerim lahko uporabimo novo tehnologijo za izboljšanje delovanja organizacije in ustvarjanje novih poslovnih priložnosti.

Metodologija predpisuje naslednje korake za opredelitev informacijske strategije s poslovnega in informacijsko-tehničnega pogleda (Martin, 1991, str. 359):

- Poslovni pogled vsebuje:
 - o izdelavo organizacijske sheme podjetja v okviru izbranega CASE orodja,
 - o opredelitev ciljev in strategij organizacije,
 - o analizo tehnoloških trendov in možnosti uporabe novih tehnologij za kreiranje novih poslovnih priložnosti ali konkurenčnih prednosti,
 - o opredelitev kritičnih dejavnikov uspeha podjetja in njihovo delitev navzdol po organizacijski shemi,
 - o intervju s ključnimi managerji v podjetju, z namenom opredelitve problemov, priložnosti in informacijskih potreb na najvišjem nivoju,
 - o beleženje vseh pridobljenih informacij v repozitorij znotraj integriranega CASE orodja.
- Informacijsko-tehnični pogled vsebuje:
 - o razvoj procesnega modela za celotno podjetje, ki prikazuje osnovne funkcije podjetja,
 - o razvoj grobega entitetno-relacijskega modela,
 - o analizo funkcij in entitet s pomočjo matrike in določitev poslovnih področij,
 - o opredelitev prioritet razvoja informacijskih sistemov za posamezna poslovna področja.

¹⁷ angleško: critical success factors (CSF), so maloštevilni dejavniki, ki so kritični za nadaljnji uspeh poslovanja. (Avison, Fitzgerald, 1996, str. 49)

Tako opredeljen načrt informacijske strategije dokumentira poslovne potrebe in določa prioritete, ki so osnova nadaljnega razvoja informacijskih sistemov. Načrt omogoča stalen pogled na informacijske potrebe najvišjega vodstva pri razvoju posameznih projektov (Avison, Fitzgerald, 1996, str. 284).

2.4.2. Analiza poslovnega področja

V okviru faze načrtovanja informacijske strategije so opredeljena poslovna področja in prioriteta izgradnje informacijskih sistemov za posamezna področja. V fazi analize poslovnega področja obravnavamo izbrano področje in podrobno analiziramo ter opredelimo procese in podatkovne strukture. Učinkoviti tehniki v fazi analize sta JRP in JAD. V tej fazi je analiza neodvisna od tehnologije, ki jo nameravamo uporabiti, in od obstoječih sistemov, ki že pokrivajo izbrano področje. Pomembno je opredeliti, kaj je potrebno zgraditi, način implementacije je prepuščen naslednjima fazama. V tej fazi izvedemo (Avison, Fitzgerald, 1996, str. 286-287):

- Analizo entitet in funkcij. Treba je podrobno analizirati procese in njihove medsebojne odvisnosti ter dopolniti diagram funkcionalne dekompozicije in diagram odvisnosti med procesi. Nadalje je treba podrobno opredeliti entitete in povezave med njimi. Entitete opredelimo do nivoja atributov.
- Analizo interakcij, kjer v obliki matrike prikažemo povezavo med posameznimi procesi in entitetami. Opredelimo način manipuliranja s podatki (kreiranje, branje, popravljanje, brisanje¹⁸).
- Analizo obstoječih sistemov, kjer obstoječi sistem analiziramo na enak način, in prav tako pripravimo entitetno-relacijski ter procesni model, z namenom primerjave obstoječega stanja in novo načrtovanega sistema.
- Potrditev pravilnosti opravljenih analiz glede pravilnosti, popolnosti in stabilnosti. Model preizkusimo tudi z možnimi scenariji novo nastalih potreb in sprememb.
- Pripravo za naslednjo fazo, ki obsega določitev vrstnega reda nadaljnega načrtovanja in konstrukcije po posameznih funkcionalnostih in analizo možnosti ponovne uporabe že obstoječih modelov ali kode.

Podrobni procesni in podatkovni modeli predstavljajo osnovo za fazi načrtovanja in konstrukcije sistema, ki jih lahko izvedemo po posameznih podsistemih z metodologijo RAD, ali z uporabo nadaljnjih korakov tradicionalnega življenjskega cikla razvoja informacijskih sistemov. V primeru uporabe RAD je treba izbrati ustrezno velikost podsistema, da ga je mogoče razviti v omejenem časovnem obdobju v okviru majhne velikosti teama. V nadaljevanju sledi kratek opis faz načrtovanja ter konstrukcije in prehoda.

¹⁸ angleško: create, read, update, delete, od tod ime CRUD matrika

2.4.3. Načrtovanje sistema

Fazo načrtovanja sistema razdelimo na logično in fizično načrtovanje, kjer v logičnem načrtovanju še niso upoštevane posebnosti okolja implementacije. Logično načrtovanje vsebuje (Avison, Fitzgerald, 1996, str. 289-291):

- Preliminarni načrt podatkovne strukture, ki ga izvedemo za vse načrtovane sisteme v poslovnem področju, da zagotovimo integracijo in povezljivost sistemov. V okviru tega koraka konvertiramo entitetno-relacijski model v strukturo izbranega sistema za upravljanje s podatkovno bazo.
- Načrt strukture sistema, v katerem povežemo posamezne procese s procedurami, ki jih opredelimo v naslednjem koraku.
- Načrt procedur vsebuje opredelitev algoritmov posameznih procedur in načinov prehoda iz ene procedure v drugo. Načine prehoda opišemo z diagrami strukture dialoga¹⁹.
- Potrditev pravilnosti, popolnosti in uporabnosti pripravljenih načrtov. Uporabnost pogosto preizkušamo z uporabo prototipov.
- Pripravo na fizično načrtovanje.

V okviru logičnega načrta opredelimo tudi vse uporabniške vmesnike in poročila. V okviru fizičnega načrtovanja je treba:

- pripraviti načrt polnjenja podatkov in dokončno opredeliti podatkovno strukturo v izbranem sistemu za upravljanje s podatkovno bazo,
- pripraviti načrt vseh programov, modulov in ponovno uporabnih vzorcev,
- pripraviti načrt programov, ki bodo omogočali konverzijo podatkov iz starega v nov sistem,
- načrtovati varnostne procedure ter procedure za merjenje hitrosti delovanja sistema,
- izdelati načrt systemskega testa in
- načrt implementacije.

Na koncu faze načrtovanja pripravimo tudi seznam potrebnih virov za fazo konstrukcije in prehoda.

2.4.4. Konstrukcija in prehod

V tej fazi dokončno implementiramo sistem in ga uvedemo v obratovanje. V okviru konstrukcije generiramo programske module, kreiramo ustrezne tabele, pripravimo podatke za testiranje in izvedemo test posameznih modulov. Nadalje izvedemo integracijski test in test sprejemljivosti sistema za uporabnike ter generiramo spremljajočo dokumentacijo. Ko sistem uspešno izpolni kriterije sprejemljivosti, smo pripravljeni na prehod.

Prehod je kontroliran premik iz uporabe obstoječih sistemov v uporabo novo razvitega sistema. V okviru prehoda izvedemo šolanje uporabnikov, namestitve potrebne strojne in

¹⁹ angleško: dialogue structure diagram

ново razvite programske opreme ter preizkus delovanja sistema. Sledi končni test sprejemljivosti glede na opredeljene kriterije in morebitna namestitve sistema na več lokacij.

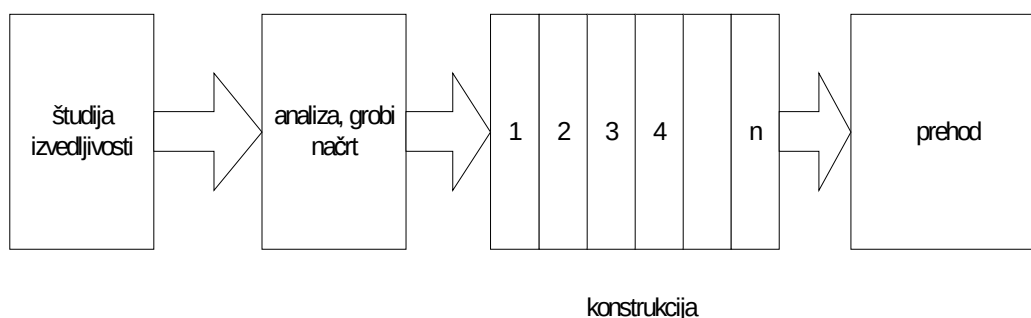
Po prehodu v operativno delovanje sledi pregled in ocenjevanje sistema, optimizacija delovanja in nadaljnje vzdrževanje.

Uporaba metodologije informacijskega inženirstva je, prav tako kot RAD, kritično odvisna od uporabljenega integriranega CASE orodja, saj le-to omogoča izgradnjo podatkovnih in procesnih modelov na nivoju celotnega podjetja ter generiranje ustreznih programskih modulov iz podrobno opredeljenih modelov. Na ta način je možno ločeno graditi posamezne sisteme in podsisteme vzporedno, tudi po RAD metodologiji, a še vedno ohraniti povezljivost tako nastalih sistemov preko enotnega podatkovnega modela.

2.5. Postopni naraščajoči model razvoja IS

Postopni naraščajoči model razvoja IS²⁰ upošteva dejstvo, da je za kompleksne sisteme nemogoče podati vse zahteve na začetku razvoja. Ideja pristopa je v tem, da se najprej opravi analiza in načrt na višjem nivoju celotnega želenega sistema, nadaljnje podrobno načrtovanje in implementacija pa potekata v več iteracijah.

Slika 4: Postopni naraščajoči model razvoja IS



Vir: Fowler, Scott, 2000, str. 14

Faza konstrukcije se sestoji iz več iteracij. Rezultat vsake iteracije je programska rešitev produkcijske kakovosti, preizkušena in integrirana, rešuje pa del celotnih zahtev projekta (Fowler, Scott, 2000, str. 15). Izdelek je lahko namenjen zgolj za notranjo uporabo ali pa za predajo zgodnjim uporabnikom. Vsaka iteracija vsebuje običajne faze analize, načrtovanja, implementacije in testiranja. V fazi prehoda je treba izvesti uglasovanje sistema in izobraževanje uporabnikov.

Tovrsten način razvoja informacijskega sistema omogoča postopno izgradnjo posameznih podsistemov in prenos izkušenj pridobljenih v izgradnji predhodnih podsistemov, v nove. Zaradi postopnosti razvoja je lažje vključevati spremenjene zahteve, ki se lahko pojavijo v času razvoja, saj se podrobna analiza in načrtovanje opravita znotraj vsake iteracije pri izdelavi posameznega dela sistema. Razdelitev projekta v več iteracij skrajšuje cikel, ki je

²⁰ angleško: incremental model

potreben za izdelavo posameznega dela rešitve in tako omogoča kontrolo nad hitrostjo in kakovostjo razvojnega procesa.

Glede na dejstvo, da se posamezni deli sistema razvijajo v različnih zaključenih iteracijah, je zaradi izogibanja problemom pri integraciji posameznih podsistemov treba načrtovati vsebino vsake iteracije ter neprestano vzdrževati usklajen model analize in načrtovanja. Pristop je še posebej primeren za objektno orientirani razvoj informacijskih sistemov, saj združevanje podatkov in operacij nad njimi v objekte ter jasno definirani vmesniki zaradi lažje integracij novo razvitih funkcionalnosti omogočajo postopno dograjevanje funkcionalnosti v že izdelane rešitve.

3. PRENOVA POSLOVNIH PROCESOV

3.1. Izhodišča za prenovo poslovnih procesov

V zadnjih dveh desetletjih smo v poslovnem svetu priča procesom, ki bistveno menjajo način, na katerega morajo organizacije poslovati, da bi si zagotovile napredek, ali pa samo goli obstoj podjetja. Hiter tehnološki razvoj omogoča prodornim podjetjem, da proizvajajo kvalitetne izdelke v vedno krajših razvojnih ciklih. Procesi globalizacije in odpiranja meja so podjetja, ki so prej delovala v relativno zaščitenih okoljih, izpostavili močni konkurenci. Zaradi močne konkurence in velike ponudbe blaga so kupci v močnejšem položaju od ponudnikov, ki se morajo prilagajati njihovim vse bolj različnim potrebam tako s pestrostjo svojega proizvodnega programa kot z odlično kakovostjo izdelkov in vedno nižjimi cenami. V današnjem svetu tako delujejo trije dejavniki, ki posamič ali v medsebojni kombinaciji povzročajo potrebo po novem načinu poslovanja podjetij (Hammer, Champy, 2001, str. 20) :

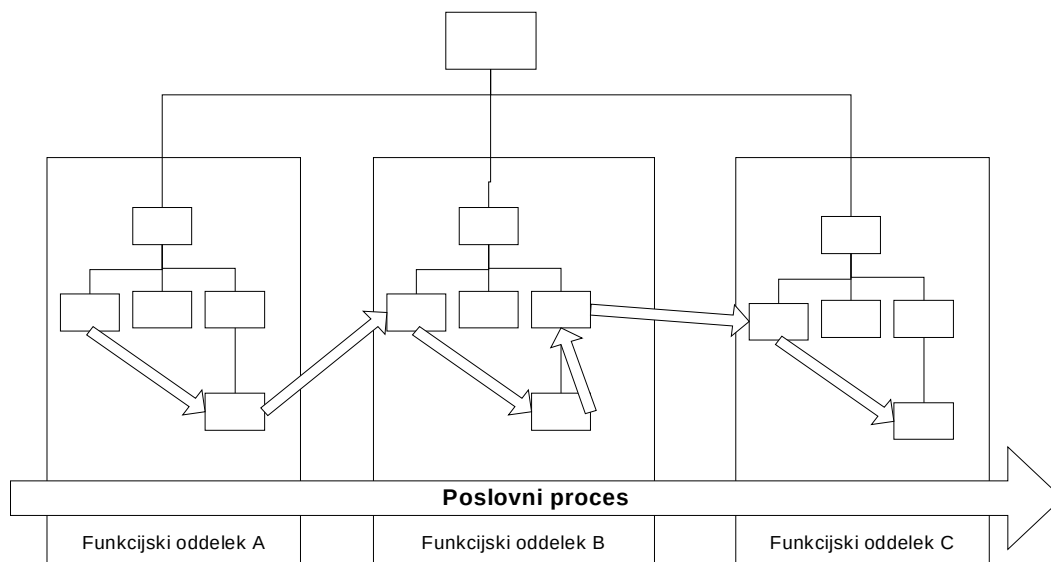
- **Kupci prevzemajo pobudo.** Zaradi dovolj velike ponudbe med seboj različnih izdelkov v današnjem svetu kupci prevzemajo pobudo in niso več pripravljeni kupovati standardnih izdelkov, ki bi jih podjetja lahko množično proizvajala, kot so to delala v preteklosti. Tako potrošniki kot podjetja, ki kupujejo izdelke za potrebe lastnega procesa, zahtevajo izdelke, ki ustrezajo njihovim specifičnim potrebam. Zato morajo podjetja razviti načine poslovanja, ki zagotavljajo fleksibilno proizvodnjo in visoko raven storitev, ki zadovoljujejo kupčeve potrebe.
- **Konkurenca je vedno močnejša.** Imeti izdelek, ki funkcionalno ustreza in je cenovno sprejemljiv, ni več dejavnik, ki bi zagotavljal prodajo. Zaradi velikega števila variant izdelkov podjetij, ki izkoriščajo tržne niše, in različnega načina konkuriranja na različnih trgih, je potrebno imeti kakovostne izdelke, širok spekter ponudbe in dodane storitve tako pred kot tudi po prodaji. Novo nastala podjetja, ki ne nosijo bremen preteklosti, pišejo pravila na novo. Ne poskušajo posnemati uveljavljenih podjetij, ampak poskušajo izkoriščati njihove slabosti. Kreativna uporaba sodobne tehnologije lahko spremeni način, na katerega podjetja med seboj tekmujejo na nepričakovane načine.
- **Sprememba postane edina stalnica.** Spremenjene zahteve kupcev in močna konkurenca vzpodbujata podjetja k uvedbi vedno novih izdelkov v vedno krajših časih. V današnjem svetu ne zadošča več kratek čas, ki je potreben za izdelavo izdelka, pomemben je kratek čas, v katerem je podjetje sposobno razviti nov izdelek. Ne zadošča le imeti odličen izdelek po ustrezni ceni, ki bi podjetju zagotavljal uspešno poslovanje. Zaradi napredka tehnologije in ostre konkurence še tako dober izdelek zelo hitro zastara in ga je potrebno zamenjati z novim. Kratki razvojni cikli uvajanja novih izdelkov so predpogoj za uspešno poslovanje. Podjetja, ki organizirajo svoje procese tako, da so sposobna slediti veliki hitrosti sprememb, postanejo zmagovalna podjetja. Ne izdelki, temveč procesi za izdelavo izdelkov, prinašajo podjetju dolgoročni uspeh (Hammer, Champy, 2001, str. 27).

3.1.1. Organizacija podjetja in poslovni procesi

Kljub zahtevam poslovnega okolja, ki narekujejo obliko organizacije podjetja, ki omogoča večjo prilagodljivost, večina današnjih podjetij v svoji organizacijski strukturi in načinu poslovanja nosi bremena preteklosti. V časih, ko je bila množična proizvodnja prevladujoč način poslovanja in so bili kupci zadovoljni, če so le uspeli kupiti izdelek, ki so ga potrebovali, so podjetja svojo učinkovitost dosegala z uporabo znanega mehanizma delitve dela na najenostavnejša opravila in uporabo ostro določene hierarhične strukture. Hierarhična struktura ima namen zbiranja in prenosa informacij od spodaj navzgor ter posredovanja navodil za delo v obratni smeri. Podjetja so razdeljena na različne funkcije, ki so dokaj avtonomne in zaposleni različnih funkcij v nižjih hierarhičnih ravneh praviloma ne sodelujejo med seboj, ampak se problematika, povezana z delovanjem več funkcij, posreduje na ustrezne višje nivoje odločanja in nazaj. Tako uporaba hierarhične strukture in delitve funkcij povzroča probleme, ki onemogočajo dovolj hiter odziv na zahteve poslovnega okolja (Martin, 1994, str. 630) :

- Hierarhije gradijo meje med funkcijskimi področji, tako da se med funkcijska komunikacija usmerja preko višjih hierarhičnih ravni.
- Informacija se na svoji poti filtrira in spreminja.
- Hierarhija preprečuje sprotno odzivanje in podaljšuje čas, potreben za sprejem odločitev.
- Managerji posvečajo svojo pozornost političnim motivom, boju za prevlado in status namesto, da bi se ukvarjali s potrebami kupca.
- Zaposleni, ki so v stiku s kupcem vedo, kaj kupci pričakujejo, vendar informacija ne pride do vrhnjega managementa, ki lahko brez tovrstnih informacij sprejema neustrezne odločitve.

Slika 5: Funkcijski »silosi«



Poslovni proces²¹, ki je namenjen izpolnjevanju potreb kupca, tako potuje skozi več funkcijskih oddelkov. Zaradi kontrol, ki so prisotne ob prehodu izločkov iz ene funkcije v drugo, so prisotne številne zamude in napake, ki podaljšujejo čas, ki je potreben za izpolnitev kupčevega naročila. Poleg tega kupec nima vpogleda v stanje njegovega naročila, saj zaradi funkcijske razdrobljenosti v t.i. »funkcijske silose« nihče od sodelujočih v posameznih **delovnih procesih**²² nima celotnega pregleda nad procesom. Tako organizirano podjetje ni pripravljeno na današnje zahteve, kjer sta potrebna sproten odziv na kupčeve zahteve in visoka kakovost spremljajoče storitve.

Organizacija, ki naj bo sposobna za tekmovanje v današnjem poslovnem okolju, mora biti organizirana okrog svojih ključnih poslovnih procesov. Poslovni proces je potrebno obravnavati celovito in imeti vzpostavljen sistem nadzora nad izvajanjem procesa. Poslovne procese v organizaciji lahko delimo na ključne, podporne in upravljalvske (Črv, 2000, str. 59).

- Ključni poslovni procesi ustvarjajo vrednost z vidika kupcev in so tesno povezani s strateškimi usmeritvami poslovnega sistema. Rezultati teh procesov morajo imeti za odjemalce jasno razpoznavno vrednost.
- Podporni procesi neposredno ne ustvarjajo vrednosti kupca, vendar ustvarjajo pogoje za delovanje ključnih poslovnih procesov. Njihov namen je predvsem zadovoljevanje notranjih potreb odjemalcev v ključnih procesih, z vidika kupcev pa lahko dodajajo vrednost le s podpiranjem ključnih procesov.
- Tretja skupina so upravljalvske procesi, katerih namen je predvsem načrtovanje in nadziranje izvajanja preostalih dveh skupin. Njihova pomembna naloga je tudi razvijanje in uveljavljanje ustreznega sistema meril za vrednotenje učinkovitosti poslovnih procesov in njihovih izvajalcev, posredno pa prispevajo k ustvarjanju in vzdrževanju ustrezne organizacijske kulture²³.

Za izboljšanje uspešnosti poslovanja je treba podrobno analizirati in po potrebi prenoviti procese iz vseh treh skupin. Ključni procesi morajo biti urejeni tako, da so usklajeni s potrebami kupcev. Posledično je zaradi spremembe ključnih procesov treba spremeniti tudi podporne procese. Upravljalvske procesi morajo vzpostaviti nov sistem meril za ocenjevanje novo oblikovanih procesov.

3.2. Prenova procesov

Prenovo poslovnih procesov²⁴ lahko opredelimo kot temeljito preverjanje procesov in njihovo radikalno spremembo za doseganje dramatičnih izboljšav kazalcev uspešnosti, kot so stroški, kakovost, storitev in hitrost (Hammer, Champy, 2001, str. 50). Predstavlja več disciplinarni in več funkcijski pristop k prenovi, da lahko naslovi potrebe celotne

²¹ Poslovni proces (angleško: business process) je zbirka aktivnosti, ki pretvorijo eno ali več vrst vhodov v izhod, ki ima vrednost za kupca (Hammer, Champy, 2001, str. 38).

²² Delovni proces je niz aktivnosti, nalog in korakov, v katerih je opravljeno dejansko delo organizacije za proizvodnjo ekonomskega izločka, ki generira donos na vloženi kapital (Watson, 1994, str. 36)

²³ Splošno lahko kulturo opredelimo kot način razmišljanja in življenja, ki si ga je skozi čas izoblikovala skupina ljudi, ki si delijo skupne in enake vrednote. V tem smislu organizacijsko podjetniško (poslovno) kulturo vidimo kot vedenje posameznega člana skupine in način, kako posamezniki sodelujejo v delovnih skupinah, oddelkih in celotnem podjetju. (Kovačič, 1998, str. 87)

²⁴ V literaturi zasledimo angleške izraze business process reengineering, business engineering;

organizacije. Vsebuje uskladitev kupčevih potreb s poslovnimi procesi za izločitev izmeta²⁵ in stroškov, za skrajšanje proizvodnega cikla ter povečanje produktivnosti, ki ustvarja priložnost za doseganje poslovnih ciljev, tako rasti dobička kot tudi povečanja tržnega deleža (Watson, 1994, str. xi).

Prenove procesov se lahko lotimo na bolj ali manj radikalen način. Poznamo (Kovačič, 1998, str. 99):

- celovito ali strateško prenovo poslovanja, ki je usmerjena v vsa ključna strateška vprašanja poslovanja podjetja in zajema v celoti vse poslovne procese podjetja in njihovo informatizacijo in
- preureditev ali prenovo in informatizacijo posameznih poslovnih procesov ali njihovih delov. Pri tem gre največkrat poudarek možnostim, ki jih nudi sodobna informacijska tehnologija. Zato slednji obliki pravimo tudi informacijske prenova.

3.2.1. Informacijska tehnologija omogoča prenovo procesov

Pri uporabi informacijske tehnologije kot sredstva za spreminjanje procesov je potrebno biti pozoren predvsem na možnosti, ki jih tehnologija ponuja za organiziranje dela na drugačen način in ne le za avtomatizacijo obstoječih (neoptimalnih) procesov. Avtomatizacija obstoječih procesov lahko blokira nadaljnjo prenovo, saj podpira star način razmišljanja in stare procese.

Ključno vprašanje pri uporabi tehnologije je (Hammer, Champy, 2001, str. 89) : Kako lahko uporabimo tehnologijo, da bi delali stvari, ki jih sedaj ne delamo? Bistvo prenove je v inovaciji, v izrabi tehnologije za doseganje novih ciljev. Sposobnost prepoznavanja novih načinov uporabe tehnologije je ključnega pomena za uspešno prenovo procesov.

Informacijska tehnologija omogoča decentralizacijo poslovanja ob ohranjanju prednosti centralizacije, sprotno spremljanje rezultatov poslovanja in njihovo primerjavo s planiranimi rezultati, izločanje nepotrebnih postopkov preverjanja vsebine dokumentov, avtomatizacijo kreiranja naročil in plačevanja, uporabo ekspertnih sistemov, pomoč pri odločanju in druge prednosti, s katerimi lahko bistveno spremenimo obstoječe poslovne procese.

3.2.2. Pristop k preurejanju procesov

Preden preidemo k preurejanju procesov, je najprej treba ugotoviti, da je preurejanje potrebno. Posledice neustreznega delovanja ključnih procesov so navadno hitro opazne skozi pritožbe kupcev in izgubo poslov, dolge proizvodne cikle, velik izmet, ipd. Identifikacija vzrokov navadno ni tako preprosta.

Prvi in osnovni korak prenove je, da vodstvo podjetja spozna potrebo po koreniti prenovi in sprejme trdno odločitev, da bo prenovilo ključne poslovne procese podjetja. Odločitev ni lahka, saj od teh procesov podjetje živi in ideja o koreniti spremembi sistema med delovanjem ni najbolj prijetna. Potrebno je identificirati ključne procese in cilje prenove.

²⁵ angleško: waste

V ta namen je potrebno odgovoriti na vprašanja:

- Kdo so naši kupci?
- Kako zadovoljujemo njihove potrebe?
- Kateri so njihovi faktorji zadovoljstva?

Glede na ugotovljeno opredelimo ključne procese in cilje prenove poslovnih procesov.

3.2.3. Kadri, potrebni za prenovu

Pred nadaljevanjem je treba izbrati ustrezne kadre, ki bodo vodili in izvajali prenovu. V različnih vlogah pri procesu prenove nastopajo (Hammer, Champy, 2001, str. 106-121) :

- **Vodja prenove** je navadno predstavnik najožjega vodstva podjetja, ki avtorizira in motivira v celotnem procesu prenove. Njegova vloga je v prepričevanju ljudi in njihovi pripravi na spremembe. S pojasnjevanjem vizije prenovljenega podjetja utira pot ostalim sodelujočim v projektu, da lažje premagujejo morebitne ovire, ki izvirajo iz strahu pred spremembami. Poleg motiviranja sodelujočih tudi pooseblja podporo vodstva organizacije in zagotavlja sodelovanje vseh ravni managementa.
- **Lastnik procesa** je manager, ki je odgovoren za obravnavani poslovni proces in njegovo prenovu. Njegova naloga je podobna nalogi vodje prenove, le da je omejena na konkretno obravnavani poslovni proces. Lastnik procesa sestavi team strokovnjakov, ki bo načrtoval in izvedel prenovu. Nadalje zagotavlja potrebne vire in sodelovanje ostalih managerjev v funkcijskih enotah, skozi katere teče obravnavani proces.
- **Prenovitveni team**²⁶ je sestavljen iz strokovnjakov, ki trenutno delujejo znotraj obravnavanega procesa²⁷, in strokovnjakov, ki niso vezani na obravnavani proces, ali pa so celo iz zunanjih svetovalnih podjetij²⁸. Vloga zunanjih strokovnjakov je v nepristranskem objektivnem pristopu k obravnavani problematiki in v postavljanju vprašanj, ki predstavljajo preboj izven ustaljenega miselnega vzorca ljudi, ki že dlje časa sodelujejo v obravnavanem procesu. Naloga teama je izdelava načrtov novih procesov, običajno pa člani sodelujejo tudi pri implementaciji novih procesov vsakdanje delovanje. V kolikor se hkrati prenavlja več poslovnih procesov, je treba oblikovati ustrezno število teamov.
- **Usmerjevalni odbor** se običajno uporablja, ko gre za projekte večjih razsežnosti. V odboru so zastopani lastniki procesov in managerji iz najvišjih položajev v organizacijski strukturi. Naloga odbora je usklajevanje prenove, ko gre za različne poslovne procese ter opredelitev strategije prenove. Pri manjših projektih se uporaba odbora navadno opušča.
- **Strokovnjak za prenovu**²⁹ je izkušen posameznik, ki je v preteklosti že sodeloval pri prenovi poslovanja. Odlično pozna metode in tehnike za prenovu procesov in je zadolžen za metodološko podporo lastnikom procesov in prenovitvenim teamom. Njegova naloga

²⁶ Team je maloštevilna skupina ljudi s komplementarnimi veščinami, ki so enako zavezani skupnemu namenu, ciljem in delovnemu pristopu, za katerega se počutijo vzajemno odgovorne (Katzenbach, Smith, 1999, str.92)

²⁷ imenujemo jih »notranji«

²⁸ imenujemo jih »zunanji«

²⁹ v literaturi navaden kot »reengineering czar« (Hammer, Champy, 2001, str. 119)

je tudi usklajevanje teamov, ki delujejo na različnih poslovnih procesih, svetovanje pri izboru notranjih članov teama in zagotavljanje potrebne infrastrukture za prenovu.

3.2.4. Koraki preнове poslovnih procesov

Če hočemo prenoviti poslovne procese, jih moramo najprej poznati in razumeti. Treba je izbrati tiste procese, ki so primerni za prenovu in jih nato na ustrezen način prenoviti. Navedeno storimo v naslednjem vrstnem redu (Hammer, Champy, 2001, str. 122 – 152) :

- identifikacija poslovnih procesov,
- izbira procesov za prenovu,
- podrobno razumevanje procesa,
- prenova procesa.

Skozi vse korake preнове postopno izgrajujemo model poslovnih procesov, ki nam prikazuje delovanje poslovnih procesov iz vidika dodajanja vrednosti za stranko. Poslovni procesi ustrezajo naravnemu toku aktivnosti, vendar so pogosto drobljeni in nejasni zaradi ostrih meja med organizacijskimi strukturami.

Identifikacija poslovnih procesov

V fazi identifikacije poslovnih procesov identificiramo ključne poslovne procese in podporne procese, ki omogočajo njihovo delovanje. Opredelimo njihov namen in odvisnosti med procesi. Pri identifikaciji procesov je treba opredeliti proces v njegovem naravnem toku in ne razmišljati o organizacijski strukturi, skozi katero proces poteka. Procesni model prikazuje delo, ki ga je treba opraviti, da pretvorimo vhode v izhode, ki bodo predstavljali vrednost za kupca in ne organizacijske strukture podjetja.

Izbira procesov za prenovu

Ko smo identificirali poslovne procese, je treba izbrati procese, ki jih je potrebno prenoviti, in vrstni red njihove preнове. Zaradi omejenosti virov, ki so potrebni za prenovu, in tveganja, ki se mu izpostavimo, če poskušamo prenoviti celo podjetje v enem samem zamahu, je običajno, da prenavljamo posamezne poslovne procese, ali pa le nekaj procesov naenkrat.

Razlogi, ki nas usmerjajo na to, da je proces potreben preнове, so lahko (Martin, 1994, str. 215) :

- proces ne deluje ustrezno³⁰,
- pričakujemo veliko izboljšanje, prenova se bo izplačala³¹,
- proces je strateško pomemben,
- proces predstavlja našo ključno sposobnost³², ki mora biti odlična, da nam omogoča konkurenčno prednost,
- izgleda, da procesa ne bo težko prenoviti (najprej poberi sadje, ki visi nizko),
- pričakovano je veliko zmanjšanje stroškov,

³⁰ v literaturi se tak proces imenuje »broken process«

³¹ angleško: payoff

³² angleško: core capability

- kupci zahtevajo boljše rezultate.

Vse našteje razloge in še mnogo drugih lahko opredelimo v okviru treh kriterijev, ki jih uporabimo za določanje vrstnega reda procesov v prenovi (Hammer, Champy, 2001, str. 127-133):

- Nefunkcionalnost procesa. Takšne procese običajno odkrivamo s pomočjo identifikacije zunanjih simptomov, ki kažejo na neoptimalno delovanje. Simptomi in njihove posledice so npr.:
 - Intenzivna izmenjava podatkov med oddelki, podvajanje podatkov in ponovni vnos podatkov kažejo na preveliko razdrobljenost naravnega procesa.
 - Velike rezervne zaloge ali odvečna sredstva kažejo na to, da podjetje nima dobrega načina za reševanje problema negotovosti poslovanja.
 - Veliko število kontrol, ki porabijo preveč časa, glede na količino časa, ki je porabljen za dodajanje vrednosti kaže na razdrobljenost procesa in pogoste prehode odgovornosti za proces med organizacijskimi enotami.
 - Potreba po ponovnem opravljanju že opravljenega dela kaže na pomanjkanje ustreznih povratnih povezav med fazami procesa, ki bi pravočasno odkrila potrebo za popravek neustrezno opravljenega dela.
 - Kompleksnost postopkov, ki so potrebni za izvedbo najenostavnejših nalog, kaže na to, da so v scenarij procesa vgrajeni mehanizmi za obravnavanje vseh mogočih izjem, ki popolnoma zabrišejo osnovni tok procesa.
- Pomembnost procesa. Pomembnost procesa je odvisna od vpliva procesa na zadovoljstvo kupca. Treba je opredeliti dejavnike, ki so za kupca posebej pomembni, kot npr. nizki stroški, stalna kakovost, predvidljivost dobavnih časov, odzivnost na kupčeve zahteve, posebne lastnosti izdelka, ipd. Osnovno načelo pri uporabi kriterija pomembnosti je, da je treba prenoviti tiste poslovne procese, ki bodo maksimirali vrednost poslovanja s podjetjem, kot jo vidi kupec iz svoje perspektive.
- Izvedljivost³³ predstavlja verjetnost, da bo prenova konkretno obravnavanega procesa uspela in prinesla želene rezultate. Večjo verjetnost za uspeh predstavljajo procesi, katerih prenova ni obsežna in ne zahteva veliko vloženih sredstev. Prenova procesov, ki vplivajo na spremembe v večjem delu organizacije, zahteva veliko sredstev in zasuk v prepričanju ljudi, je bolj tvegana, a prinaša ob uspešni prenovi tudi mnogo večje rezultate kot prenova procesov, ki so zaokroženi v majhnem delu organizacije in pomenijo izboljšanje obstoječega načina delovanja.

³³ angleško: feasibility

Podrobno razumevanje procesa

V koraku razumevanja procesa dopolnjujemo model poslovnih procesov z razgradnjo procesa na nižje ravni ter opredeljevanjem delovnih procesov in njihovih elementov. Elementi, ki opredeljujejo delovni proces so : vhodi, lastnik procesa, prejemniki izhodov procesa, izhod procesa, meje, aktivnosti, strošek, potrebni čas izvedbe, dodana vrednost in kritični dejavniki uspeha procesa (Watson, 1994, str. 67-68). Oblikovati je treba merila za ugotavljanje uspešnosti izvajanja procesov in merjenje izboljšav, ki jih nameravamo doseči s prenovo procesa.

Poleg opredelitve obstoječih procesov je treba pogledati na proces s perspektive kupca. Bistveno je razumevanje dejstva, kaj izhod iz procesa dejansko pomeni za kupca, v kakšnem procesu ga torej kupec uporablja v okviru svojega poslovnega procesa. Treba je razumeti kupčeve resnične potrebe in probleme, ne le mehanizme, ki povezujejo obe organizaciji (Hammer, Champy, 2001, str. 135).

Ko smo opredelili resnične zahteve, ki jih mora naš prenovljeni proces izpolnjevati, je treba primerjati rezultat in način dela obstoječega procesa z želenim rezultatom. Namen primerjave je ugotoviti, kaj je potrebno izboljšati, in zakaj obstoječi proces ne more izpolnjevati opredeljenih zahtev.

Eden od možnih načinov za ocenjevanje ustreznosti procesa je tudi primerjava z načinom, na katerega obravnavani proces opravljajo podjetja, ki se odlikujejo pri izvajanju tovrstnega procesa. Primerjava poslovnih procesov³⁴ predvideva iskanje podjetij, ki se odlikujejo pri izvajanju obravnavanega procesa in proučitev načina, na katerega izvajajo ta proces, s ciljem, da se naučimo, kako spremeniti lasten proces in izboljšati njegove rezultate. Na ta način lahko veliko izvemo o potrebnih spremembah lastnih procesov, obstaja pa tudi nevarnost, da zaradi nekritičnega kopiranja zanemarimo morebitne lastne prednosti, ki nam omogočajo izgradnjo še boljših procesov.

Z vprašanjem, kako bomo spremenili proces, da bo ustrezal opredeljenim zahtevam, se ukvarjamo v koraku prenove procesa, v koraku razumevanja procesa pa je potrebno nedvoumno opredeliti dejavnike, ki so kritični za uspešno izvajanje procesa.

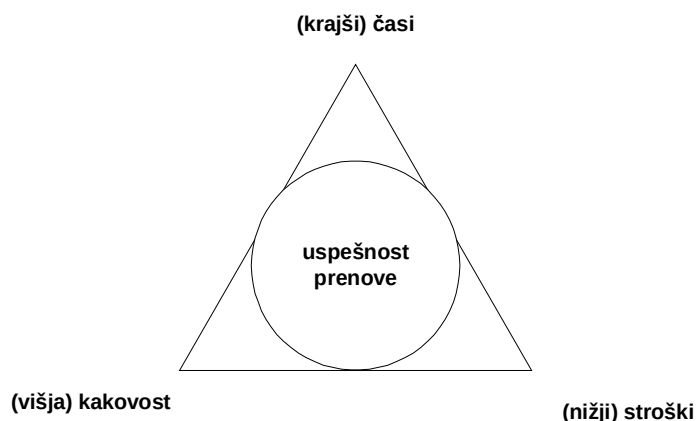
Prenova procesa

V koraku načrtovanja in uvedbe spremenjenega procesa poskušamo proces preoblikovati tako, da odpravljamo slabosti, ki smo jih ugotovili v predhodnih fazah. Pri tem je pomembno ugotoviti, ali si lahko z uporabo tehnologije pomagamo tako, da bistveno spremenimo in poenostavimo proces. Končni cilj je nastanek procesa, ki bo v čim manj korakih v čim krajšem času s čim nižjimi stroški zagotavljal ustrezno kakovost izdelka in storitev, ki so izhod iz procesa in ustrezajo dejanskim potrebam kupca.

³⁴ angleško: benchmarking

Shematično lahko prenovo prikažemo na naslednji način (Kovačič, 1998, str. 86) :

Slika 6: Vpliv temeljnih ciljev na uspešnost prenove poslovanja



Vir: Kovačič, 1998, str. 86

Pri doseganju želenih ciljev prenove uporabljamo različne tehnike, ki vplivajo na spreminjanje posameznih rezultatov procesa, ali njihovo kombinacijo. Pogosta je uporaba naslednjih načinov:

- Združevanje več nalog v eno, bolj kompleksno nalogo, pri čemer izločimo nepotrebne čakalne čase, ki so prej obstajali med posameznimi fazami procesa. Seveda takšno združevanje zahteva bolj vsestransko usposobljene sodelavce, ki so sposobni opravljati bolj kompleksno nalogo.
- Pooblaščenje izvajalcev za sprejemanje odločitev, ki se nanašajo na način izvajanja naloge, brez nepotrebne posredovanja nadzornih in vodilnih kadrov. Na ta način izvajalec ni zadolžen le za izvajanje naloge, temveč je odgovoren tudi za njen rezultat. Tak način dela omogoča bolj celovito obravnavanje kupčevih zahtev na enem mestu.
- Korake procesa je treba izvajati v njihovem naravnem zaporedju. Zaporedno opravljanje faz ni več nujna predpostavka procesa, kjer je le mogoče iščemo načine za vzporedno opravljanje različnih nalog, kar omogoča krajšanje časa izvajanja celotnega procesa.
- Procesi imajo več variant. Namesto enega zapletenega scenarija izvajanja procesa, ki vključuje vse možne zaplete, oblikujemo glavni proces, ki je kratek in učinkovit, ter alternativne procese, v primeru zapletov. V toku dejanskega izvajanja procesa uvrstimo konkretno zahtevo kupca v ustrezen način obravnavanja³⁵. Za vsak primer uporabimo le tiste postopke, ki so nujno potrebni. Tako opredeljeni procesi so tudi bolj pregledni, saj vsaka varianta obravnava le primere, za katere je predvidena.
- Zmanjševanje števila kontrol in ukinjanje nalog, ki ne dodajajo vrednosti. Prenovljeni proces naj vsebuje le tak obseg kontrole, ki je še ekonomsko upravičen. Namesto izvajanja sprotne kontrole se bolj poslužujemo statističnih vzorcev, ki naj pokažejo odstopanja od predvidenih načinov izvajanja procesa.

³⁵ tak način opredelimo z izrazom »triaža«, ki je v medicini običajen pri razvrščanju poškodovancev, glede na nujnost obravnavanja primera in zahtevnost poškodbe

- Zmanjšanje števila izvajalcev, ki so glede obravnavanega procesa v stiku s stranko, saj tako izločimo možnost različnih neusklajenih dogovorov, ki bi jih sklenili različni izvajalci, vsak za svoj del procesa. V kolikor je proces kompleksen, lahko imenujemo posebnega izvajalca, ki skrbi za usklajevanje in je v kontaktu s stranko³⁶.
- Uporaba hibridnih decentralizirano-centraliziranih načinov organiziranja, ki po eni strani omogoča, da se posamezni oddelki specializirajo na posameznem produktu, ki ga stranki ponujajo (npr. različni tipi zavarovanj), hkrati pa z uporabo informacijske tehnologije zagotovimo celovit nadzor nad poslovanjem konkretne stranke na vseh področjih, kjer se stranka pojavlja v naši organizaciji. Tovrsten način obravnavanja je mogoč izključno le ob izgradnji ustreznega informacijskega sistema.

3.2.5. Posledice prenove poslovnih procesov

Korenita prenova poslovnih procesov bistveno spremeni način dela v organizaciji. Spremenita se distribucija moči v podjetju in organizacijska struktura. Poleg tega se spremenijo tudi:

- naloge postanejo bolj kompleksne in zahtevajo bolj vsestransko izobražene izvajalce,
- s tem se spremeni način priprave izvajalcev za opravljanje dela iz priučitve na posamezne naloge v celovito izobraževanje,
- enote, v katerih se opravlja delo, se spremenijo iz funkcionalnih oddelkov v procesne teame,
- spremeni se vloga izvajalcev, ki dobijo pooblastila in odgovornost za uspeh izvedene naloge³⁷,
- način merjenja rezultatov se spremeni iz merjenja aktivnosti v merjenje njihovih rezultatov.
- Spremeni se sistem vrednot v podjetju v smeri direktne povezave zadovoljstva kupcev s percepcijo dobro opravljenega dela. Ustrezno je treba spremeniti tudi sistem nagrajevanja, kjer je variabilni del predvsem odvisen od končnega rezultata in meritev zadovoljstva kupcev, manj pa od rezultatov posameznih oddelkov. Tako se prepreči zapiranje oddelkov v svoje meje in ozka notranja optimizacija procesov, brez širšega pogleda na celoten poslovni proces.
- Vloga managerja se spremeni iz vloge nadzornika v vlogo usmerjevalca in »trenerja«.
- Organizacijska struktura se spremeni iz hierarhične v bolj plosko obliko.

Zaradi vseh navedenih sprememb je pri prenovi poslovnih procesov vedno treba računati na odpor znotraj podjetja, v organizacijskih enotah, ki se jih prenova dotika. Predvsem je močan odpor managerjev na srednjem in nižjih nivojih, ki v pričakovanih spremembah vidijo nevarnost za svoj položaj in se ne čutijo sposobne za nov način opravljanja dela.

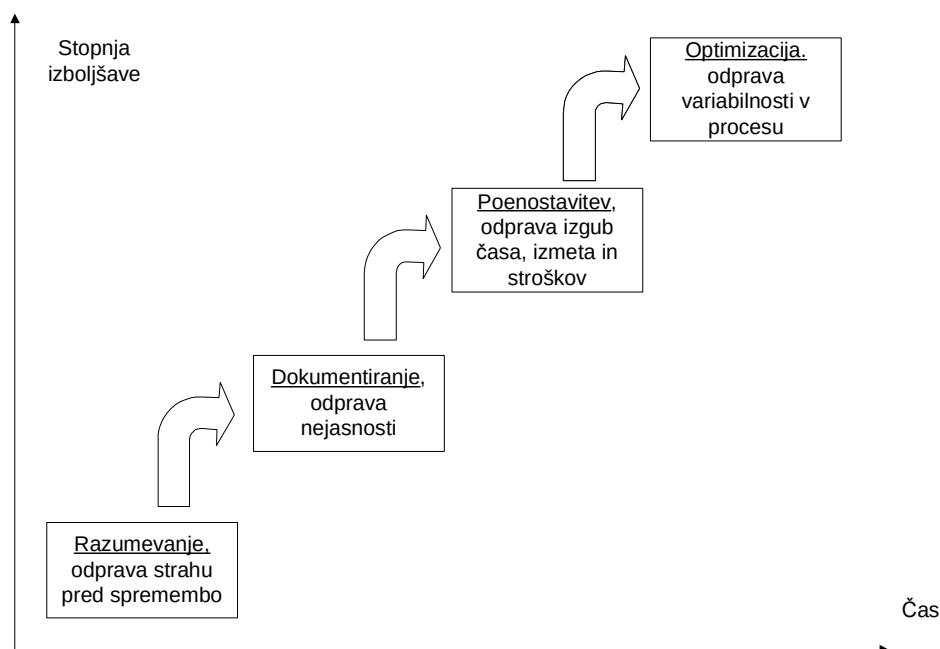
Zato se je spreminjanja poslovnih procesov treba lotiti na način, ki bo pojasnil potrebnost predvidenih sprememb.

³⁶ V literaturi ga imenujejo »case manager«

³⁷ V literaturi imenujejo ta pojav »employee empowerment«

Eden od načinov je uporaba naslednjega postopka za spreminjanje procesov, ki je v uporabi tudi v okviru programov postopnih izboljšav, kot je celovito obvladovanje kakovosti³⁸.

Slika 7: Stopnje izboljšav poslovnega procesa³⁹



Vir: Watson, 1994, str. 69

V prvem koraku je treba povečati pripravljenost ljudi, da sprejmejo spremembe. To izvedemo tako, da preučimo obstoječ način dela in razumemo kritične dejavnike uspeha, zahtevane zmožnosti in primanjkljaj potrebnih sposobnosti. Treba je jasno opredeliti, kakšen naj bi bil bodoči proces. Opredelitev prihodnosti mora biti jasna in razumljena s strani vseh udeležencev, ki se jih sprememba dotika. V procesu vzpostavljanja skupnega razumevanja problema izločimo večino strahu pred spremembami.

V drugem koraku je treba dokumentirati nove poslovne procese. Ob dokumentiranju se navadno pokažejo prej skrite nedoslednosti in prekrivanja vlog izvajalcev. Nejasnosti je treba podrobno opredeliti in proces dokumentirati na urejen način, na primer v okviru sistema ISO 9000. V tem koraku je ključno vprašanje: »Ali delamo prave stvari?«

V tretjem koraku poskušamo izboljšati produktivnost z zmanjševanjem stroškov, časa in izmeta, ki jih lahko enostavno identificiramo. Izločamo tiste aktivnosti, ki ne dodajajo vrednosti, in tako poskušamo čimbolj poenostaviti proces.

V četrtem koraku izvedemo nadaljnjo optimizacijo z zmanjševanjem variabilnosti. Z uporabo poskusnega načrtovanja ter statističnega spremljanja rezultatov procesa poskušamo oblikovati proces, ki konsistentno izdeluje izdelke, ki med seboj ne odstopajo. Šele ko smo končali četrti korak izločevanja nepotrebnih aktivnosti, lahko proces učinkovito avtomatiziramo.

³⁸ angleško: total quality management, TQM

³⁹ »UDSO« model, understand-document-simplify-optimize (Watson, 1994, str. 69)

3.2.6. Tveganja pri prenovi poslovnih procesov

Pri prenovi poslovnih procesov je tveganje tem večje, čim bolj je proces pomemben in čim večji del organizacije je treba prenoviti. Ob tem lahko verjetnost neuspeha še povečamo, če se prenove ne lotimo dovolj celovito in ne zagotovimo potrebnih predpostavk. Najpomembnejša predpostavka je trdna odločenost vodstva podjetja, da se prenova opravi, in dosledna vztrajnost na izbrani poti. Treba je zagotoviti vse potrebne vire, še posebej se to nanaša na ustrezno informacijsko tehnologijo in zagotavljanje potrebnih kadrovskih okrepitev. Pri prenovi je treba upoštevati obstoječo kulturo podjetja z razumevanjem, kako jo je potrebno spremeniti, da lahko prenova uspe. V kolikor ne spremenimo kulture in načina definiranja dobrega rezultata, je malo verjetno, da bodo prenovljeni procesi zaživel, kot je predvideno v načrtu prenove. Pri samem postopku preoblikovanja procesov se ne smemo zadovoljiti z malimi popravki, če je možno z korenito spremembo doseči bistveno boljše rezultate.

Končni namen prenove poslovnih procesov podjetja je v oblikovanju takega podjetja, ki se bo zaradi svoje fleksibilnosti v prihodnosti lahko prilagajalo spremembam, ki prihajajo, ter s kombinacijo večjih prenov posameznih procesov in stalne postopne izboljšave vzdrževalo svojo konkurenčno prednost skozi daljše časovno obdobje.

4. OBJEKTNO USMERJENE METODOLOGIJE RAZVOJA INFORMACIJSKIH SISTEMOV

Metodologije razvoja informacijskih sistemov pred pojavom objektno orientiranih metodologij dosledno ločujejo podatkovni in procesni pogled. Tako se podatki in procesi modelirajo v različnih modelih, v fazi implementacije pa se podatki nahajajo v podatkovni bazi, proces pa se kodira v programske procedure in rešitve, ki manipulirajo s temi podatki. Problematična je sprememba podatkovne strukture, ki obvezno povzroči iskanje programskih modulov, ki uporabljajo spremenjeno strukturo.

4.1. Osnovni koncepti objektnega pristopa

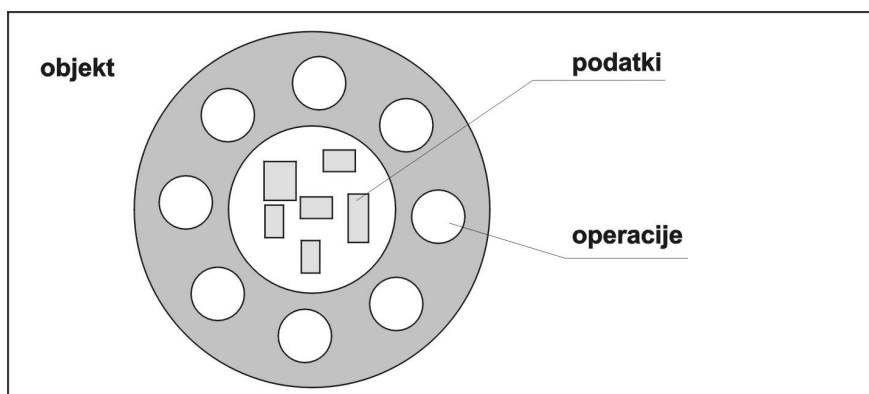
Osnovna ideja objektnega pristopa razvoja informacijskih sistemov je v čimbolj neposredni preslikavi objektov realnega sveta (npr. stranka, artikel) v model⁴⁰ informacijskega sistema, ki obravnava dotično poslovno področje. Osnovni koncepti takega modela: objekt, ograjevanje, razred, sporočilo, mnogoličnost, dedovanje, bodo predstavljeni v nadaljevanju.

4.1.1. Objekt

Objekt je entiteta⁴¹, ki je sposobna shranjevati stanje (informacijo) in ponuja operacije za pregledovanje ali spreminjanje tega stanja (Jacobson et al., 1994, str. 44). Objekt je katerakoli stvar (entiteta) realnega sveta, o kateri hranimo podatke in obnašanje (Grad, Jaklič, 1996, str. 117). Objekti vsebujejo podatke in načine spreminjanja ter doseganja podatkov. Stanje objekta opisujejo atributi, določeno pa je z vrednostjo atributov. Obnašanja objekta opisujejo operacije, ki predstavljajo način za pregledovanje in spreminjanje vrednosti atributov in s tem stanja objekta. V tem je bistvena novost objektnega pristopa, saj se na enem mestu združujejo podatki in načini za spreminjanje podatkov. Vsak objekt ima enolični identifikator⁴², ki ga loči od ostalih objektov, čeprav imajo morda enake vrednosti atributov.

4.1.2. Ograjevanje

Slika 8: Objekt vsebuje podatke in operacije



Vir: Črv, 2000, str. 9

⁴⁰ Model je poenostavitev realnosti, narejen za boljše razumevanje sistema, ki ga kreiramo (Booch, Rumbaugh, Jacobson, 1999, str. 463) Model je abstrakcija, predstavitev dela realnega sveta (Avison, Fitzgerald, 1996, str. 62)

⁴¹ Entiteta je stvar ali dogodek, ki ga opazujemo (Grad, Jaklič, 1996, str. 76)

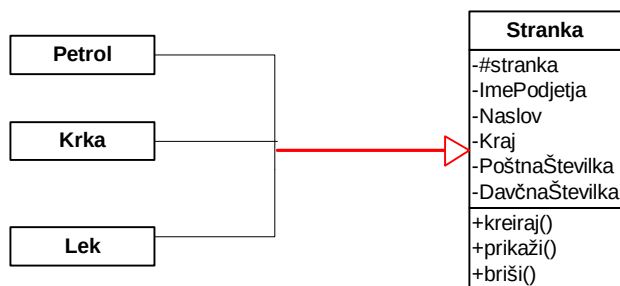
⁴² angleško: OID – object identifier

V objektno orientiranem sistemu so podatki shranjeni v notranjosti objekta. Edini način doseganja in spreminjanja podatkov je uporaba operacij, ki so definirane v tem objektu. Objekt vsebuje vse potrebne operacije za manipuliranje s podatki znotraj objekta. Operacije so implementirane z metodami, ki predstavljajo programsko kodo, ki izvaja opredeljene spremembe nad podatki. Objekt se navzven predstavlja le s svojimi operacijami, notranja struktura objekta in način implementacije operacij pa sta zunanjemu svetu (drugim objektom) skrita. Mehanizem skrivanja informacij pred zunanjim svetom⁴³ z izključnim dostopom do podatkov preko operacij objekta se imenuje ograjevanje⁴⁴. Drugi objekti lahko pridobijo informacijo ali spremenijo stanje objekta le s pomočjo klica operacije zadevnega objekta. Tak način dela ščiti vsebino objekta pred zunanjim vdorom, odpravlja pa tudi potrebo po tem, da vsi objekti poznajo notranjo strukturo drugih objektov (Taylor, 1992, str.50). Na ta način je omogočena lokalizacija sprememb, saj se morebitne spremembe načina obdelave podatkov implementirajo le na enem mestu.

4.1.3. Razred

Objekte, ki imajo podobne lastnosti in obnašanje, lahko združujemo v objektne tipe. Objekte, ki imajo poleg enakih operacij tudi enako notranjo strukturo, združujemo v razrede. Razred je ena od (morda mnogih) implementacij tipa (Jacobson et al., 1994, str. 50). Razred je opis niza objektov, ki imajo enake attribute, operacije, povezave in semantiko (Booch, Jacobson, Roumbaugh, 1999, str. 18). Razred tako predstavlja vzorec za objekte in opisuje njihovo interno strukturo. Objekti istega razreda imajo enako definicijo operacij in informacijske strukture (Jacobson et al., 1994, str. 50). Tako so posamezni objekti primerki⁴⁵ razreda.

Slika 9: Primerki razreda



Iz slike je razvidno, da so posamezni objekti – konkretna podjetja – primerki razreda Stranka, ki ima opredeljene attribute, ki ga opisujejo, in operacije.

Atribut je imenovana lastnost razreda, ki opisuje razpon vrednosti, ki ga primerki lastnosti lahko vsebujejo. Predstavlja neko lastnost stvari, ki jo modeliramo, ki je skupna vsem objektom razreda (Booch, Jacobson, Roumbaugh, 1999, str. 50).

Operacija je implementacija storitve, ki jo lahko drugi objekti zahtevajo od razreda, z namenom vplivanja na obnašanje (Booch, Jacobson, Roumbaugh, 1999, str. 51).

⁴³ angleško: information hiding

⁴⁴ angleško: encapsulation

⁴⁵ angleško: instance

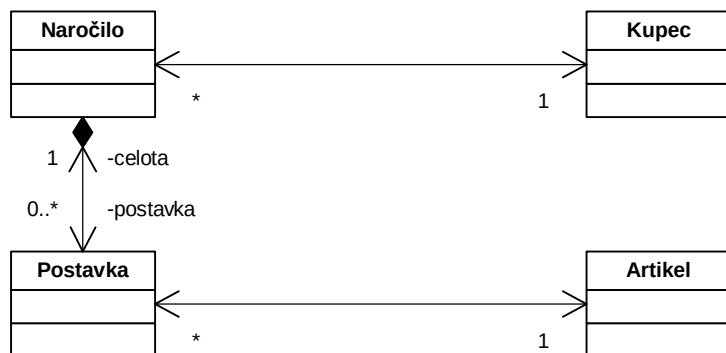
4.1.4. Asociacije

Asociacije predstavljajo povezave med primerki razredov (določena oseba dela za določeno podjetje). Na konceptualnem nivoju predstavljajo asociacije konceptualne povezave med razredi (Fowler, Scott, 2000, str. 52). Asociacija je strukturna povezava, ki pove, da so objekti ene stvari povezani z objekti druge (Booch, Rumbaugh, Jacobson, 1999, str. 65). Asociacija med seboj lahko povezuje dva ali več razredov. V prvem primeru jo imenujemo binarna asociacija, v drugem, izjemnem primeru pa n-arna asociacija. Grafično jo prikazujemo kot polno črto, ki povezuje razreda med seboj. Osnovne vrste asociacij predstavljajo naslednje strukture:

- »je-del« (angleško: is-part-of),
- »je-povezan-z« (angleško: is-associated-with),
- »je« (angleško: is-a).

Asociaciji lahko priredimo ime in smer asociacije. Asociacija je lahko eno-ali dvosmerna.

Slika 10: Primer asociacij med razredi



Na primeru lahko vidimo tip asociacije »je povezan z« med kupcem in naročilom ter postavko naročila in artiklom. V obeh primerih je asociacija dvosmerna, kar v implementaciji pomeni, da mora imeti v prvem primeru razred Naročilo metodo, ki ugotovi kateri kupec je dal naročilo, prav tako pa mora razred Kupec imeti metodo, ki ugotovi, katera naročila pripadajo določenemu kupcu. Opredeljena je tudi števnost⁴⁶, ki nam pove, da eno naročilo pripada natanko enemu kupcu, en kupec pa ima lahko več naročil.

Primer asociacije med naročilom in postavko naročila je vrste »je del«, ki predstavlja dejstvo, da so postavke sestavni del naročila. Lahko jo imenujemo agregacija. Prav tako je opredeljena števnost in obojestransko usmerjenost asociacije. Konkreten primer je posebna oblika agregacije, t.i. sestavljena agregacija⁴⁷, ki ima dodatno lastnost, da deli brez celote ne morejo obstajati, in je torej ob brisanju naročila treba izbrisati tudi vse postavke tega naročila.

Tip povezave »je« predstavlja generalizacijo, ki bo podrobneje predstavljena v okviru opisa dedovanja.

⁴⁶ Uveljavljena sopomenka je kardinalnost, (angleško: cardinality, multiplicity)

⁴⁷ angleško: composite aggregation

4.1.5. Sporočilo

Objekti komunicirajo med seboj s sporočili. Prenos sporočil je edini način, s katerim objekti sodelujejo med seboj. Objekt s sporočilom zahteva od drugega objekta izvedbo neke operacije. Tako je običajna vsebina sporočila:

- identifikacija objekta, ki naj izvede operacijo – prejemnik sporočila,
- ime operacije, ki naj se izvede,
- parametri operacije, če obstajajo.

Prejemnik izvede operacijo in vrne rezultat objektu, ki je zahtevo posredoval (pošiljatelju). Na ta način se omogoča ograjevanje in nedotakljivost notranjosti objekta s strani drugih objektov.

4.1.6. Mnogoličnost

Kadar ima operacija z istim imenom v različnih razredih različen pomen in implementacijo, govorimo o mnogoličnosti⁴⁸. Sporočilo lahko sproži različen odziv glede na razred, v katerem je klicana operacija implementirana. Na način izvedbe torej vpliva razred prejemnika sporočila.

4.1.7. Dedovanje

V primeru, ko imamo v našem modelu dva ali več razredov s podobnimi atributi in operacijami, lahko s postopkom generalizacije izločimo skupne attribute in operacije v poseben razred, ki ga z uporabo mehanizma dedovanja povežemo z razredi, ki ohranijo le tiste lastnosti, ki so jim lastne in jih tako izdelani nadrazred⁴⁹ nima. Obratno lahko s postopkom specializacije iz razreda, ki ima ustrezne lastnosti kreiramo podrazred⁵⁰, ki mu dodamo posebne operacije ali attribute, ki jih potrebujemo. Včasih prav z namenom kreiranja novih podrazredov kreiramo t.i. abstraktni razred, ki ima le funkcijo združevanja lastnosti, ki jih dedujejo njegovi potomci. Tak razred navadno nima primerkov, imajo jih le njegovi potomci. Znotraj specialnega razreda lahko poleg dodajanja atributov in operacij spremenimo tudi pomen in implementacijo določene operacije. Postopku pravimo predefiniranje⁵¹. Postopek je zelo učinkovit, vendar nam poslabša preglednost celotnega modela, saj na prvi pogled ni vidna različna funkcionalnost operacije z enakim imenom v dveh, z mehanizmom dedovanja povezanih razredih.

Potomec lahko deduje lastnosti od enega ali več prednikov. V prvem primeru govorimo o enkratnem, v drugem pa o večkratnem dedovanju. Večkratno dedovanje prav tako zmanjšuje preglednost modela in jo je treba posebej previdno uporabljati. Ob nepravilni uporabi lahko pride do ponavljajočega se dedovanja⁵², kjer enake operacije obstojajo v obeh prednikih izbranega razreda, ko imata oba prednika skupnega prednika, od katerega sta podedovala te operacije (Jacobson et al., 1994, str. 67).

⁴⁸ Uveljavljena sopomenka je »polimorfizem«

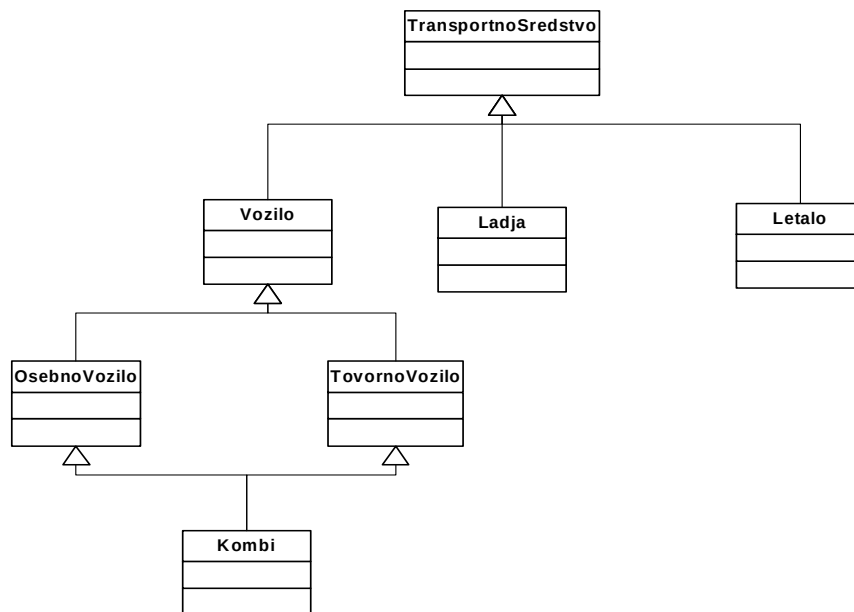
⁴⁹ angleško: superclass; Sopomenka je tudi prednik, angleško ancestor

⁵⁰ angleško: subclass; Sopomenka je potomec, angleško descendant

⁵¹ angleško: overriding

⁵² angleško: repeated inheritance

Slika 11: Primer dedovanja



Na sliki je prikazan primer nepravilne uporabe večkratnega dedovanja na primeru razreda »Kombi«, ki naj bi podedoval lastnosti od osebnega in tovornega vozila. Lastnosti, ki so podedovane iz razreda »Vozilo«, se bodo dvakrat dedovale v enaki obliki. Situacijo lahko rešimo tako, da na novo definiramo te operacije (predefiniranje), izberemo ene ali druge, ali pa spremenimo strukturo dedovanja npr. tako, da »Kombi« postane direkten potomec razreda »Vozilo«. Glede na potrebno preglednost strukture dedovanja je najboljša zadnja možnost.

Pozornost je treba nameniti razločevanju klasifikacije⁵³ in generalizacije. Na primeru izjav: Janez je človek (klasifikacija), človek je vrsta (generalizacija) in vrsta je življenjska oblika (generalizacija) lahko vidimo, da ima besedna zveza »človek je življenjska oblika« ustrezen smisel, zveza »Janez je življenjska oblika« pa ne. Generalizacija je tranzitivna, klasifikacija pa te lastnosti nima (Fowler, Scott, 2000, str. 95). Razločevanje lahko izvedemo, če si namesto povezave »je« poskusimo vprašati »je vrsta«⁵⁴.

Dedovanje ob pazljivi uporabi omogoča učinkovito vzdrževanje, saj se spremembe operacij vršijo na enem mestu, na najvišjem mestu v hierarhiji razredov. Tako spremenjeno operacijo podedujejo vsi potomci tega razreda. Dedovanje omogoča obsežno ponovno uporabo, saj se z dedovanjem lastnosti izbranega razreda in dodajanjem potrebnih funkcionalnosti bistveno skrajšuje čas implementacije.

⁵³ Klasifikacija je dejanje odstranjevanja različnosti med objekti, da lahko vidimo skupne lastnosti (Odell, 1999, str. 122)

⁵⁴ angleško: is kind of

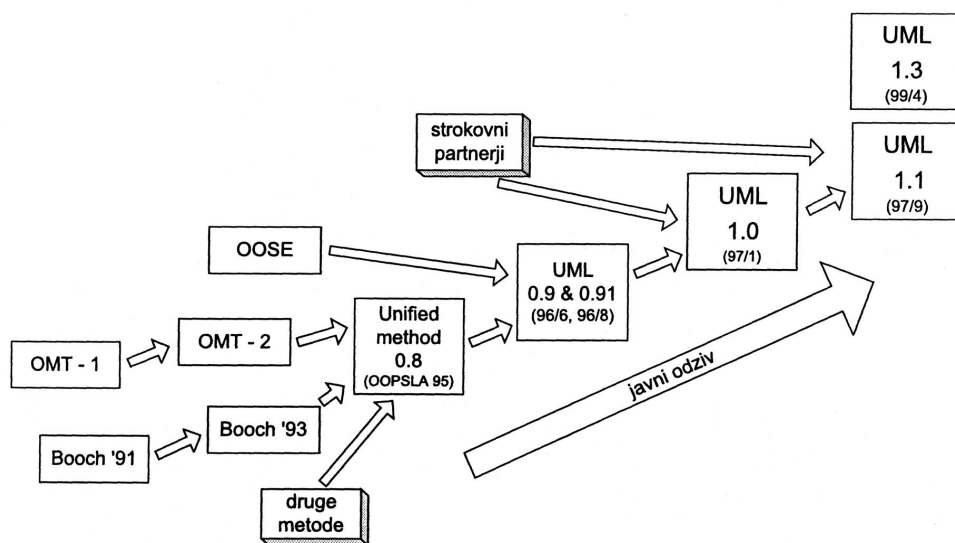
4.2. Poenoten jezik za modeliranje – UML

Poenoten jezik za modeliranje - UML⁵⁵ je nastal z združitvijo več metodologij za objektno orientirano analizo in načrtovanje, med katerimi so bile najpomembnejše:

- OMT, avtorja Jamesa Roumbaugh,
- OOSE, avtorja Ivarja Jacobsona,
- Boochova metodologija, avtor: Grady Booch.

UML je bil v okviru OMG (Object Management Group) standardiziran in je postal standardni jezik za modeliranje.

Slika 12: Razvoj jezika UML



Vir: Živkovič, Heričko, Rozman, 2000, str.201

UML je jezik za modeliranje in ne metodologija za razvoj informacijskih sistemov. Metodologije običajno poleg jezika vsebujejo tudi proces izgradnje sistema z opredeljenimi fazami in njihovimi izdelki. Avtorji (Roumbaugh, Jacobson, Booch) poleg jezika, ki je metodološko neodvisen, predlagajo še svoj proces razvoja informacijskih sistemov, ki temelji na postopno naraščajočem modelu in se imenuje Poenoten proces razvoja programske opreme⁵⁶. UML, uporabljen skupaj s tem procesom, lahko označimo za metodologijo.

UML kot jezik za modeliranje omogoča izgradnjo različnih vrst modelov, ki nam omogočajo (Booch, Jacobson, Roumbaugh, 1999, str. 6):

- vizualizacijo trenutnega ali bodočega sistema⁵⁷,
- prikaz strukture in obnašanja sistema,
- izdelavo vzorcev, ki nas vodijo pri konstrukciji sistema,
- dokumentiranje naših odločitev.

⁵⁵ angleško: unified modelling language

⁵⁶ The unified software development process

⁵⁷ angleška izraza: as-is, to-be

4.2.1 Osnovni gradniki UML

UML sestavljajo osnovni gradniki:

- stvari⁵⁸, imenovali jih bomo elementi,
- povezave⁵⁹,
- diagrami.

Opis gradnikov je povzet po viru The unified modeling language user guide (Booch, Jacobson, Roumbaugh, 1999, str. 18 – 26).

4.2.1.1. Elementi

Elementi so lahko namenjeni opisovanju strukture⁶⁰, obnašanja⁶¹, združevanja⁶² ali opomb⁶³.

Med strukturne elemente spadajo razred, vmesnik⁶⁴, sodelovanje⁶⁵, primer uporabe⁶⁶.

Razred je opis niza objektov, ki imajo enake attribute, operacije, povezave in semantiko. Razred implementira enega ali več vmesnikov. Grafično je predstavljen kot pravokotnik, ki običajno vsebuje ime, attribute in operacije.

Vmesnik je zbirka operacij, ki opredeljuje storitve, ki jih ponujata razred ali komponenta. Vmesnik opredeljuje specifikacijo in parametre operacije; izvedba operacije je skrita. Grafično je prikazan kot krog.

Sodelovanje opredeljuje interakcijo in je združba vlog in ostalih elementov, ki skupno delujejo v kooperativnem obnašanju. Razred je lahko del več sodelovanj. Sodelovanje je grafično predstavljeno kot črtkasta elipsa.

Primer uporabe je opis niza zaporednih korakov, ki jih izvaja sistem, in določenemu akterju prinašajo rezultat, ki ga lahko opazujemo. Grafično je prikazan kot elipsa.

Komponenta je fizični del sistema, ki zagotavlja uresničitev niza vmesnikov. Komponenta tipično predstavlja fizično pakiranje sicer logičnih elementov, kot so razredi, vmesniki in sodelovanja. Grafično je prikazana kot pravokotnik z zankama.

Vozlišče⁶⁷ je fizični element, ki obstaja v času izvajanja in predstavlja računalniški vir, ki ima lasten spomin in včasih tudi procesorsko moč. Grafično je prikazan kot kocka, ki vsebuje lastno ime.

⁵⁸ angleško: things

⁵⁹ angleško: relationships

⁶⁰ angleško: structural things

⁶¹ angleško: behavioral things

⁶² angleško: grouping things

⁶³ angleško: annotational things

⁶⁴ angleško: interface

⁶⁵ angleško: collaboration

⁶⁶ angleško: use case

⁶⁷ angleško: node

Elementa za opis vedenja sta interakcija in diagram stanj (angleško: state machine).

Interakcija predstavlja obnašanje, ki vsebuje niz sporočil, izmenjanih med objekti v določenem kontekstu za doseganje določenega namena. Interakcija vsebuje različne elemente, ki vključujejo sporočila, obnašanje, ki ga sporočilo sproži, in povezave med objekti. Grafično predstavljamo sporočilo kot usmerjeno daljico, ki skoraj vedno vsebuje ime operacije, ki jo sproža.

Diagram stanj predstavlja obnašanje, ki opredeljuje zaporedje stanj objekta, ki se menjajo v interakciji, kot odgovor na dogodke. Vključuje stanja, prehode (tok iz stanja v stanje), dogodke (ki sprožajo prehode) in aktivnosti (odgovore na prehode). Grafično stanje prikazujemo kot pravokotnik z zaobljenimi koti, ki vsebuje ime in podstanja, če obstajajo.

Mehanizem za grupiranje objektov v UML se imenuje paket⁶⁸. Paket je splošni mehanizem grupiranja elementov. Obstaja na konceptualnem nivoju, za razliko od komponente, ki obstaja v času izvajanja. Grafično je prikazan kot mapa.

Za dodajanje opomb in opisov v UML uporabljamo ustrezne elemente opomb, ki se imenujejo zapiski⁶⁹. Zapisek lahko pripnemo na element ali skupino elementov. Grafično je prikazan kot zavihan list.

4.2.1.2. Povezave

Povezave v UML lahko razdelimo na:

- odvisnosti,
- asociacije,
- generalizacije,
- realizacije.

Odvisnost⁷⁰ je semantična povezava med dvema elementoma, pri kateri sprememba neodvisnega elementa lahko povzroči spremembo odvisnega elementa. Grafično jo prikažemo kot črtkano puščico.

Asociacija je strukturna povezava, ki opisuje niz zvez⁷¹, ki povezujejo objekte. Poseben primer asociacije je agregacija, ki je opisana med osnovnimi koncepti na začetku poglavja. Asociacijo modeliramo kot polno črto med dvema strukturnima elementoma. V primeru agregacije je na strani črte, ki je pri elementu, ki predstavlja celoto, votel romb, v primeru sestavljene agregacije⁷² pa poln romb.

⁶⁸ angleško: package

⁶⁹ angleško: notes

⁷⁰ angleško: dependency

⁷¹ angleško: links

⁷² angleško: composite aggregation, je posebna oblika agregacije, kjer deli brez celote ne morejo obstajati. Primer sestavljene agregacije so postavke dokumenta in dokument.

Generalizacija predstavlja povezavo generalizacije in specializacije, ki sta prav tako opisani na začetku poglavja. Grafično se prikaže kot puščica z votlo konico, usmerjeno proti predniku.

Uresničitev je semantična povezava med elementoma, kjer prvi opredeljuje funkcijo, ki jo drugi izvede. Srečamo jo med vmesniki in razredi ter komponentami, ki jih uresničujejo in med primeri uporabe ter sodelovanji, ki jih uresničujejo. Grafično je prikazana kot črtkana puščica z votlo konico.

4.2.1.3. Diagrami

UML ponuja različne diagrame, ki nam služijo za prikaz različnih pogledov na obravnavani sistem. Tako predstavimo enoten model na več načinov, kar omogoča ločeno obravnavanje posameznih problemov in olajšuje modeliranje sistema. Na voljo imamo devet različnih diagramov. Diagrami, s katerimi prikazujemo statično strukturo sistema, so:

- razredni diagram,
- objektni diagram,
- diagram komponent,
- diagram porazdelitve⁷³.

Obnašanje sistema modeliramo z naslednjimi vrstami diagramov :

- diagram primerov uporabe
- diagrama interakcij sta:
 - diagram zaporedja⁷⁴
 - diagram sodelovanja
- diagram stanj
- diagram aktivnosti

Diagrami vsebujejo osnovne gradnike in jih uporabljamo za prikaz različnih pogledov na enoten model sistema, ki ga obravnavamo.

Razredni diagram

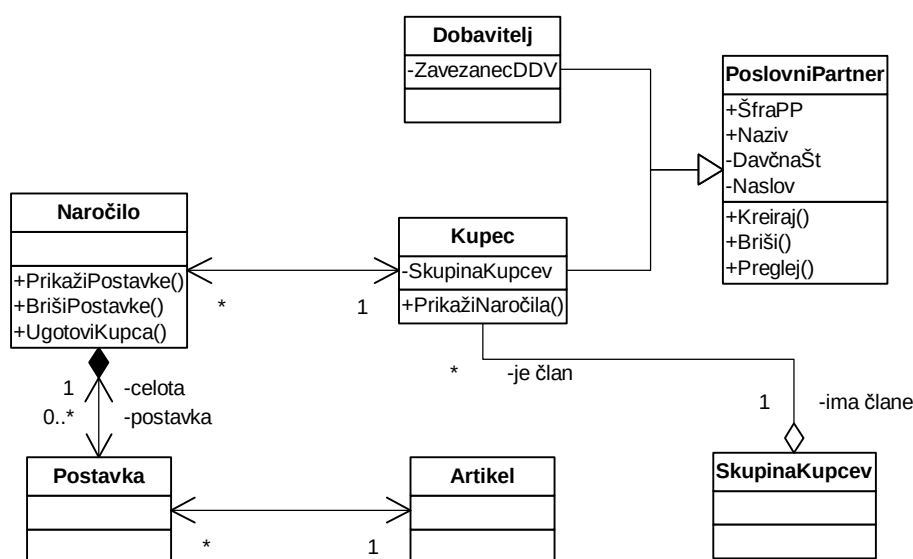
Razredni diagram prikazuje razrede, vmesnike, sodelovanja in njihove povezave. Poleg navedenih elementov lahko razredni diagram vsebuje tudi pakete, ki so namenjeni grupiranju elementov večjega modela v več manjših zaokroženih celot. V okviru modela prikazujemo razred kot pravokotnik, ki mu priredimo ime ter navedemo njegove attribute in operacije. Povezave med razredi so lahko asociacije, agregacije, sestavljene agregacije in generalizacije. Povezavam opredelimo števnost in smer, lahko jim tudi opredelimo vlogo⁷⁵, ki jo ima razred na posameznem zaključku povezave.

⁷³ angleško: deployment diagram

⁷⁴ angleško: sequence diagram

⁷⁵ angleško: role

Slika 13: Primer razrednega diagrama



Prikazani so vsi tipi povezav. Usmerjenost povezave modeliramo s puščico. Na primeru povezave kupec-naročilo vidimo, da mora zaradi obojestranske povezave kupec implementirati metodo za pregledovanje naročil kupca, na drugi strani pa ima naročilo metodo za branje podatkov o kupcu. Oba postopka seveda kličeta ustrezne elementarne postopke v drugem razredu in ne posegata direktno v podatkovno strukturo.

Objektni diagram

Objektni diagram prikazuje objekte, predstavlja trenutni posnetek⁷⁶ sistema v nekem trenutku delovanja. Vsebuje objekte in zveze⁷⁷.

Slika 14: Primer objektnega diagrama



Vidimo konkretno povezavo kupca »Janez« in skupine »Privatnik«. Notacija za prikaz konkretnega objekta je lahko objekt:razred, če pa prikazujemo splošni objekt kot predstavnika razreda, to zapišemo z :razred. Objektni diagrami se redkeje uporabljajo, predvsem za predstavitev povezav med konkretnimi primerki posameznih razredov.

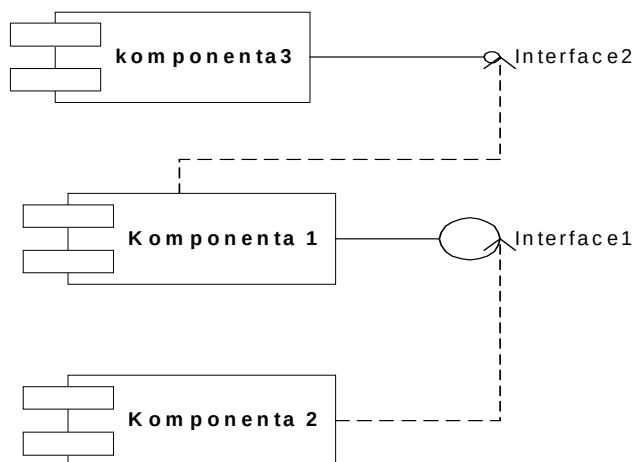
⁷⁶ angleško: snapshot

⁷⁷ angleško: link

Diagram komponent

Diagram komponent prikazuje komponente in povezave med njimi. Vsebuje komponente, vmesnike, povezave odvisnosti, asociacije, generalizacije in uresničitve. Z njim lahko modeliramo zaključene dele programske kode, ki implementirajo funkcionalnost izbranih razredov, in povezave med njimi. Komponente lahko združujemo v pakete za razdelitev sistema v več manjših zaključenih celot.

Slika 15: Diagram komponent

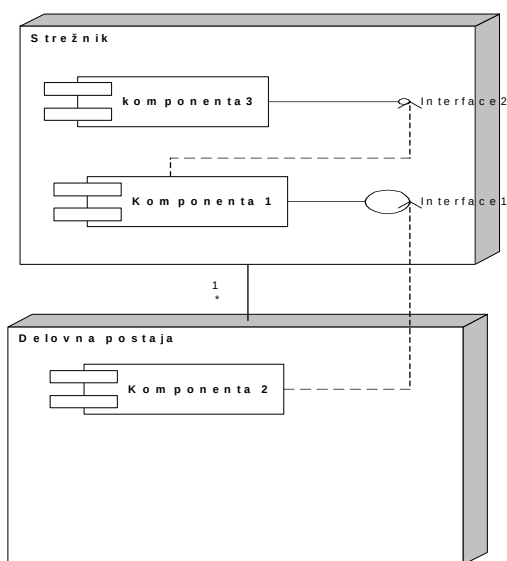


Prikazane so odvisnosti med komponentami in vmesniki kot način komuniciranja med njimi.

Diagram porazdelitve

Diagram porazdelitve⁷⁸ prikazuje namestitev komponent v vozlišča, ki predstavljajo strojno opremo, na kateri se komponente izvajajo. Vsebuje vozlišča in povezave med vozlišči, lahko pa prikazujemo tudi porazdelitev komponent med vozlišča.

Slika 16: Diagram porazdelitve



⁷⁸ angleško: deployment diagram

Diagram primerov uporabe

Diagram primerov uporabe prikazuje primere uporabe in akterje, ki so z njimi povezani. Izraža uporabniški pogled na sistem. Primeri uporabe predstavljajo potek dogodkov, ki prinašajo želene rezultate akterju, ki jih izzove. Akter je oseba ali drug sistem izven obravnavanega, ki vstopa v interakcijo s sistemom. Akter ni konkretna oseba, ampak predstavlja določeno vlogo, ki jo le-ta predstavlja (npr. prodajni predstavnik).

Primer uporabe predstavlja opis določenega dogajanja, ki nastane v interakciji akterja in sistema. Opišemo ga tekstualno, vendar opredelimo naslednje sestavine (Sturm, 1999, str. 60) :

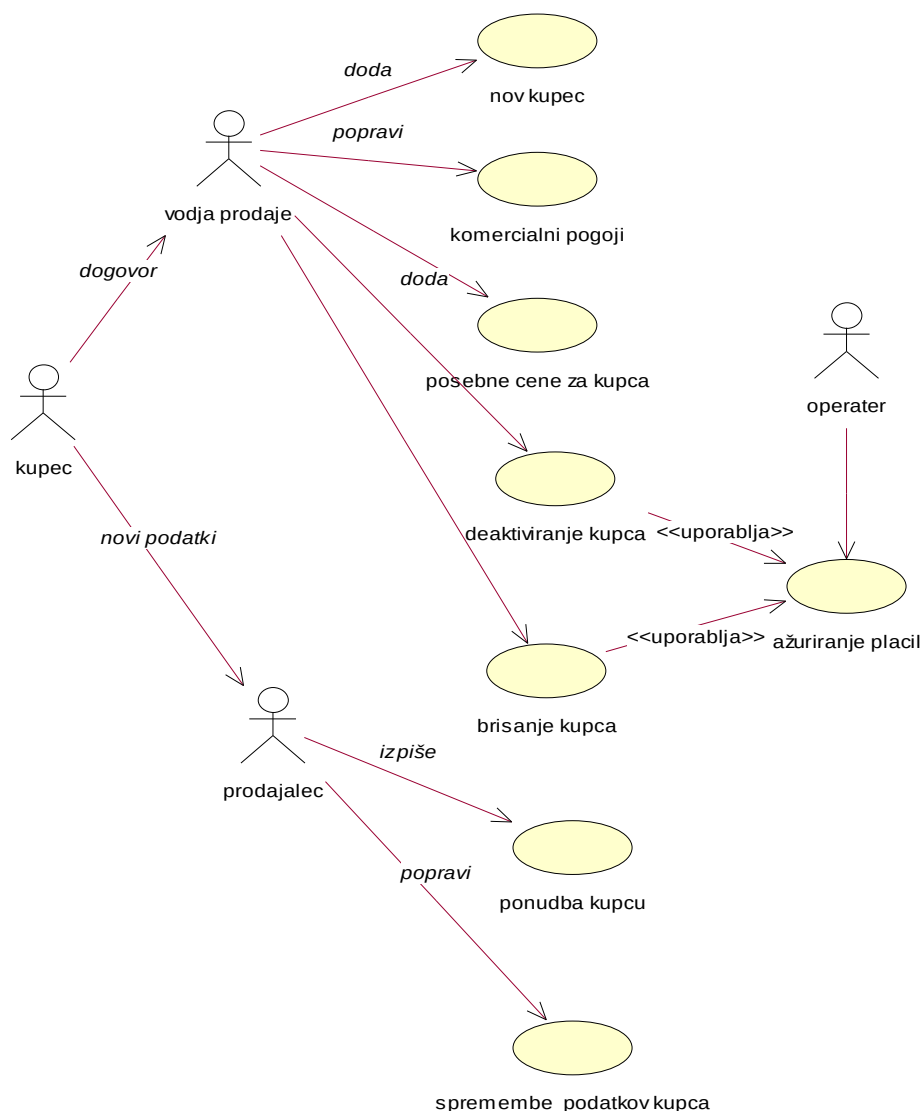
- ime primera uporabe,
- kratek opis namena,
- primarni akter - je tisti, zaradi katerega sistem gradimo,
- sekundarni akter, ki tudi sodeluje,
- začetek - je prva aktivnost primera uporabe,
- konec - je zadnja aktivnost primera uporabe,
- merljivi rezultat,
- potek dogodkov, kjer opišemo pozitivni scenarij,
- alternativni tok dogodkov, ko pogoji za izvedbo niso izpolnjeni,
- morebitni primer uporabe, ki se izvede iz opisanega,
- odprte zadeve.

Primer uporabe lahko opišemo na naslednji način:

Ime	Nov kupec
Opis	Namen je dodajanje novega kupca v sistem
Primarni akter	Vodja prodaje
Sekundarni akter	ni
Začetek	Kupec pride k akterju
Konec	Kupec je dodan v sistem, ali zavržen
Merljivi izid	Zapis kupca v sistemu
Potek dogodkov	Potencialni kupec pride k akterju. Ko akter z iskanjem po nazivu ugotovi, da želeni kupec ni prisoten v sistemu, pridobi od kupca registracijo in bonitetni obrazec. Če je boniteta v redu in registracija popolna, vpiše podatke o kupcu v sistem in potrdi zapis.
Alternativni potek dogodkov	Kupec je že v sistemu in ga akter ne doda. Registracija ni popolna, se zavrne. Boniteta ni v redu, se zavrne. Potencialni kupec je fizična oseba, se zavrne.
Povezani primeri	ni
Odprta vprašanja	ni

Na ta način se opišejo vsi primeri uporabe. Diagram primerov uporabe služi prikazovanju povezav med različnimi primeri uporabe in akterji, zato na enem diagramu prikažemo več povezanih primerov uporabe.

Slika 17: Diagram primerov uporabe



Na diagramu vidimo tudi uporabo stereotipa »uporablja⁷⁹«, ki v tem primeru pove, da primera uporabe brisanja in deaktiviranja kupca uporabljata primer uporabe, kjer se preverjajo odprte terjatve do obravnavanega kupca. Diagrame primerov uporabe uporabljamo za omejitev, kaj naj sistem dela; pravzaprav so način opredeljevanja zahtev do sistema. Primeri uporabe predstavljajo posamezne zaključene funkcionalnosti sistema.

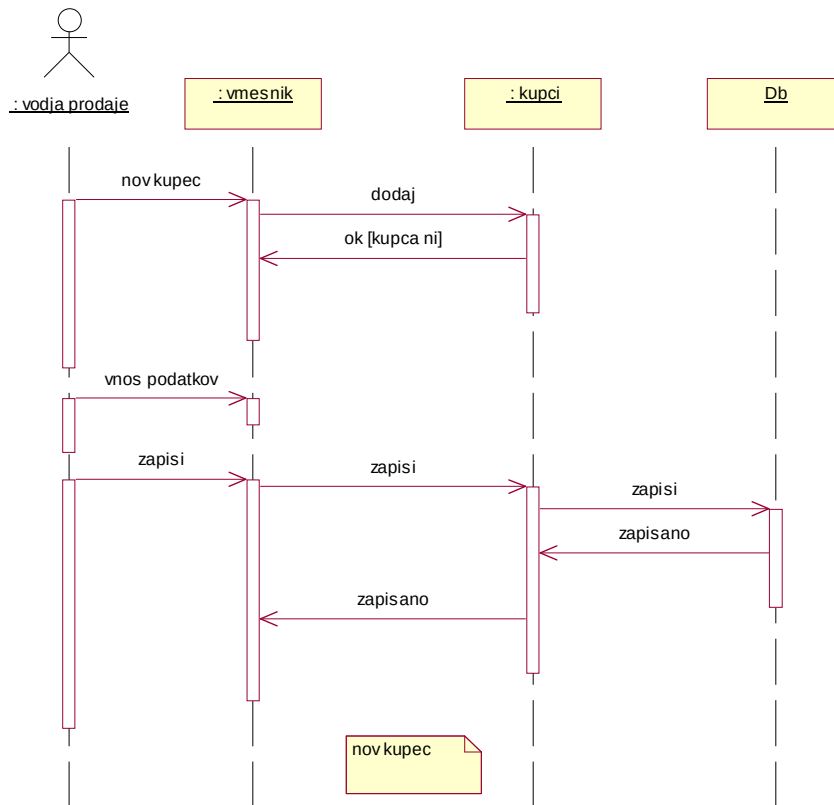
Za podroben opis izmenjevanja sporočil v sistemu uporabljamo interakcijske diagrame. Ločimo diagram zaporedja in diagram sodelovanja, ki sta dva pogleda iste vsebine in ju lahko med seboj neposredno prevedemo.

⁷⁹ angleško: uses

Diagram zaporedja

Diagram zaporedja⁸⁰ prikazuje časovno zaporedje izmenjave sporočil med objekti sistema in zunanjimi akterji. Vsebuje objekte, zveze in sporočila.

Slika 18: Diagram zaporedja



Navpična črtkana črta predstavlja življenjsko črto objekta in prikazuje čas sodelovanja objekta z drugimi objekti. Življenjska črta se konča z uničenjem objekta, ki jo označuje znak »X« na mestu brisanja. Navpični ozki pravokotniki so namenjeni označevanju časa, ko je objekt aktiven in kontrolira izvajanje dogajanja. Usmerjene puščice predstavljajo sporočila, ki se izmenjujejo med objekti. Sporočila v implementaciji naslavljaajo metode komponent, ki izvajajo zahtevane manipulacije notranje strukture objektov.

Diagram sodelovanja

Diagram sodelovanja predstavlja izmenjavo sporočil na drugačen način, ki bolj poudarja organizacijo objektov, ki sodelujejo v interakciji. V tem primeru je zaporedje pošiljanja sporočil opredeljeno z zaporedno številko sporočila, ki je pripisana sporočilu. Na diagramu sodelovanja so označene zveze⁸¹ med objekti, ki na diagramu zaporedja niso posebej prikazane.

⁸⁰ angleško: sequence diagram

⁸¹ angleško: link

Slika 19: Diagram sodelovanja

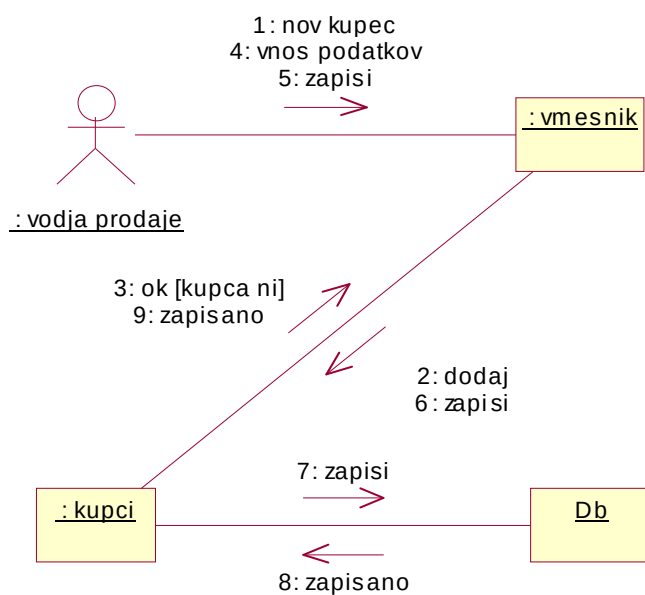
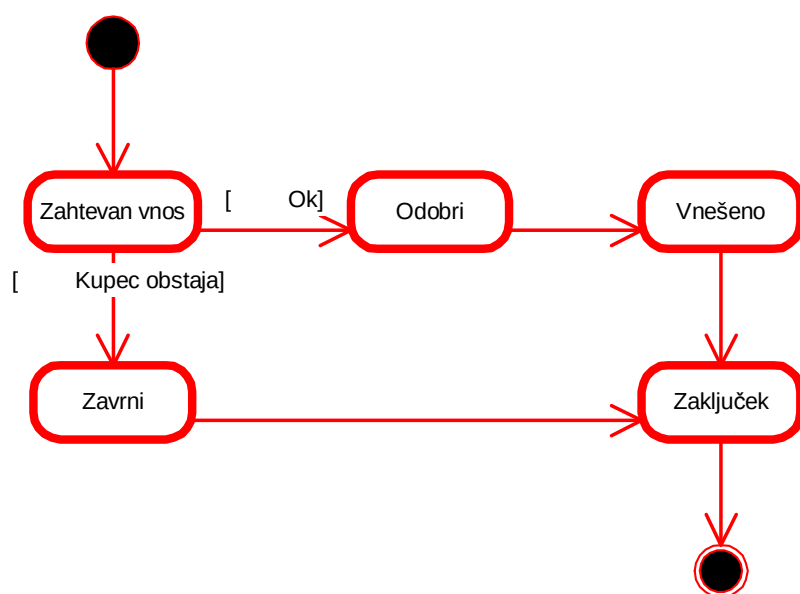


Diagram stanj

Diagram stanj prikazuje stanja, prehode, dogodke in aktivnosti. Uporablja se za prikaz spreminjanja stanja izbranega objekta glede na dogodke, ki spremembo stanj povzročajo.

Slika 20: Diagram stanj objekta :kupec

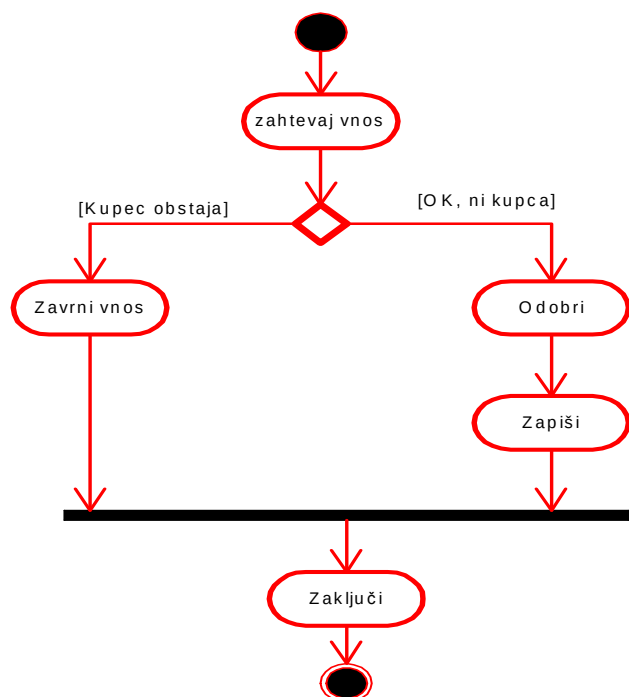


Na diagramu vidimo alternativne poti pri vnosu novega kupca glede na pogoj, da lahko novega kupca dodamo le, če ne obstaja. Ovalne oblike predstavljajo stanja, puščice predstavljajo prehode med stanji, temna kroga pa predstavljata začetno in končno stanje (znotraj krožnice).

Diagram aktivnosti

Diagram aktivnosti uporabljamo za prikaz zaporednih in vzporednih korakov v izvajanju procedure. Diagram vsebuje aktivnosti, prehode med njimi, odločitvene aktivnosti, ki glede na postavljene pogoje prehajajo v različne naslednje aktivnosti, ter uskladitvene črte⁸², v katerih se vzporedni tokovi izvajanja uskladijo in se opravi prehod do naslednje aktivnosti, ko so izpolnjeni določeni pogoji.

Slika 21: Diagram aktivnosti



Uskladitvena črta je prikazana kot debela vodoravna daljica, v katero vodita dva prehoda, iz nje pa vedno le en prehod. Uporaba uskladitvenih črt omogoča modeliranje vzporednega izvajanja procesov. Kadar z diagramom aktivnosti modeliramo prehajanje procesa med več različnimi enotami organizacije, ko gre za prehod odgovornosti, lahko z navpičnimi črtami⁸³ prikažemo tovrstno delitev.

Elementi, povezave in diagrami so osnovni gradniki poenotenega jezika za modeliranje, s katerim izdelujemo različne modele želenega informacijskega sistema glede na posamezno fazo razvojnega procesa.

4.2.2. Poenoteni proces razvoja programske opreme

Avtorji jezika UML kot način izgradnje informacijskih sistemov predlagajo poenoteni proces razvoja programske opreme⁸⁴, ki so ga opredelili kot drugi del metodologije, skupaj s poenotenim jezikom za modeliranje.

⁸² angleško: synchronization lines

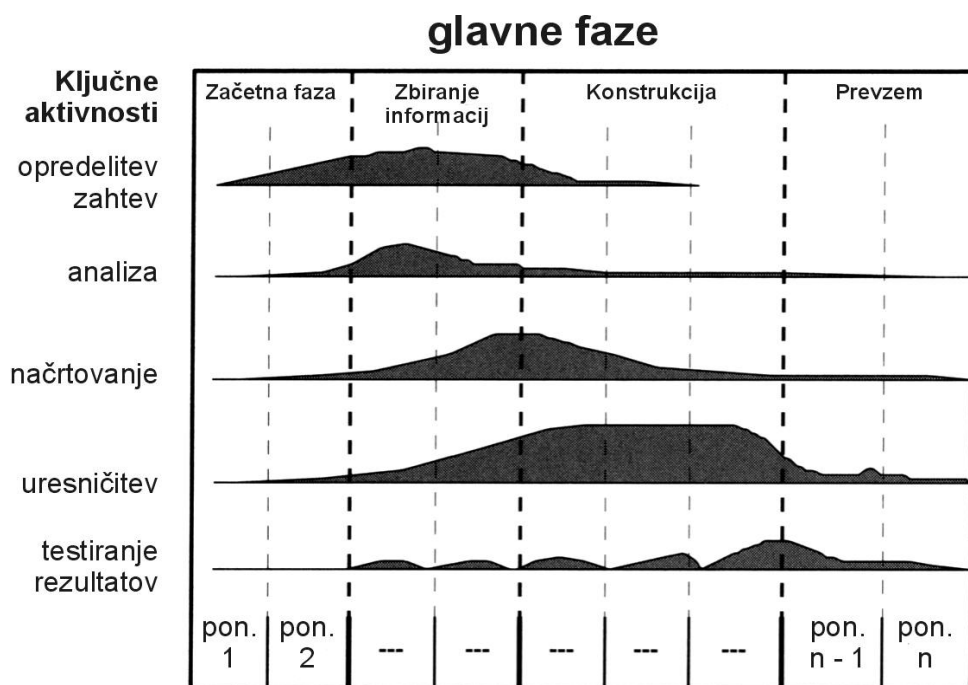
⁸³ angleško: swimlines

⁸⁴ angleško: The unified software development process

Proces razvoja programske opreme predstavlja opredelitev niza aktivnosti, ki so potrebne za spreminjanje potreb uporabnikov v konsistentni niz izdelkov, ki predstavljajo programsko rešitev in za spreminjanje dodatnih sprememb teh zahtev v nov, ponovno konsistentni niz izdelkov (Booch, Jacobson, Rumbaugh, 1999, str. 24). Izdelki poleg delujoče programske opreme obsegajo še niz med seboj konsistentnih modelov, ki prikazujejo različne poglede na sistem in so zapisani s pomočjo UML jezika. Proces je predstavljen kot iterativen in inkrementalen. Tako se celoten proces razvoja deli po dveh razsežnostih: časovno po fazah razvoja in po aktivnostih, ki jih opravljamo znotraj vsake ponovitve. Po časovni razsežnosti so opredeljene faze razvoja, ki zajemajo celoten cikel razvoja informacijskega sistema:

- začetna faza⁸⁵,
- zbiranje informacij⁸⁶,
- konstrukcija⁸⁷,
- prevzem⁸⁸.

Slika 22: Faze in aktivnosti poenotenega procesa



Vir: Booch, Jacobson, Rumbaugh, 1999, str. 104

4.2.2.1. Ključne aktivnosti procesa

Celoten razvoj se razbije na vrsto ponovitev glavnih faz procesa. Rezultat vsake ponovitve je nov prirastek⁸⁹ izdelkov, ki se kaže kot dodaten del kode, ki je pripravljena za produkcijo in polno integrirana, in razširjeni modeli, ki opisujejo različne poglede delovanja ter omogočajo enostaven prehod v naslednjo ponovitev, saj je delovanje sistema jasno prikazano.

⁸⁵ angleško: inception

⁸⁶ angleško: elaboration

⁸⁷ angleško: construction

⁸⁸ angleško: transition

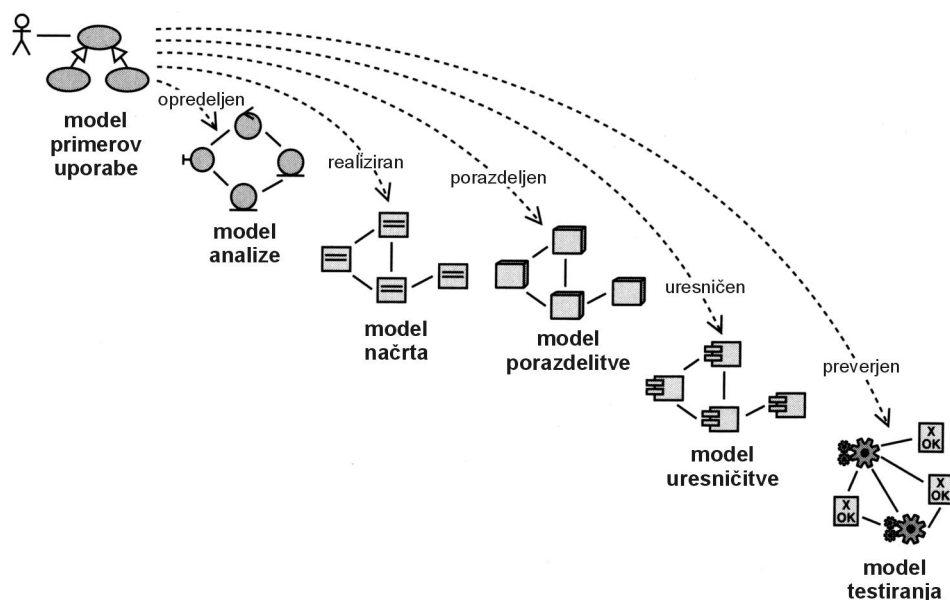
⁸⁹ sopomenka: inkrement

Znotraj vsake faze in ponovitve izvedemo zaporedje aktivnosti, ki predstavlja drugo razsežnost:

- opredelitev zahtev,
- analiza,
- načrtovanje,
- uresničitev⁹⁰,
- testiranje rezultatov.

Obseg porabe virov za vsako od teh aktivnosti je različen za različne glavne faze in je grafično predstavljen kot osenčeno področje pod krivuljami znotraj slike procesa. Za različne aktivnosti potrebujemo ustrezne modele, znotraj katerih prikažemo rezultate vsake aktivnosti. Tako so opredeljeni naslednji modeli:

Slika 23: Modeli poenotenega procesa



Vir: Booch, Jacobson, Rumbaugh, 1999, str. 10

Slika poudarja odvisnost med posameznimi modeli. Sprememba enega modela vpliva na druge. Tako se pri opredeljevanju zahtev uporablja model primerov uporabe, pri analizi model analize, za načrtovanje model načrta in model porazdelitve. Model uresničitve ustreza aktivnosti uresničevanja, model testiranja pa aktivnosti testiranja. Modeli vsebujejo diagrame, opredeljene znotraj poenotenega jezika za modeliranje.

Opredelitev zahtev

Namen opredelitve zahtev je s pomočjo primerov uporabe opredeliti model bodočega sistema. Zahtevano funkcionalnost opredelimo, ko opišemo načine uporabe sistema in rezultate, ki jih

⁹⁰sopomenka: implementacija

sistem nudi uporabnikom. Pomembno je identificirati vse različne vloge, ki komunicirajo s sistemom, s čimer opredelimo akterje. Nefunkcionalne zahteve, kot je na primer odzivnost sistema so prav tako različne pri različnih primerih uporabe in jih opišemo v njihovem okviru (Booch, Jacobson, Rumbaugh, 1999, str. 131). Primere uporabe povežemo v model s pomočjo diagramov primerov uporabe. Poleg primerov uporabe, ki so glavno orodje pri opredeljevanju zahtev, si pomagamo še z diagrami za opisovanje obnašanja sistema, predvsem z diagrami zaporedja. Poleg tega lahko naredimo tudi nekatere prototipe uporabniških vmesnikov, ki pomagajo k podrobnemu razumevanju načina interakcije uporabnika in sistema. Glede na to, da je model primerov uporabe orientiran proti uporabniku in njegovemu pogledu na delovanje sistema, pri opredeljevanju primerov uporabe pa lahko srečamo izrazoslovje, ki je specifično obravnavani problematiki, si v takem primeru pomagamo tako, da napišemo definicijo uporabljenih izrazov - pojmovnik. Primerom uporabe opredelimo pomembnost in jih razvrstimo v vrstni red za nadaljnji razvoj. Tako so glavni izdelki opredeljevanja zahtev naslednji:

- podrobno opredeljeni primeri uporabe,
- diagram primerov uporabe,
- ostali diagrami, ki pojasnjujejo posamezne posebnosti,
- seznam nefunkcionalnih zahtev,
- morebitni prototipi uporabniških vmesnikov,
- opredeljena prioriteta primerov uporabe,
- pojmovnik uporabljenih izrazov.

Na koncu opišemo model primerov uporabe kot celoto.

Analiza

Analiza predstavlja pogled na način realizacije zahtevane funkcionalnosti. Treba je podrobno analizirati uporabniške zahteve, opredeljene v predhodnem koraku. Model analize se razlikuje od modela primerov uporabe, saj (Booch, Jacobson, Rumbaugh, 1999, str. 175) :

- je opisan v formalnem jeziku, namenjenem razvijalcu sistema;
- prikazuje notranji pogled na sistem;
- sestavljen je iz razredov in paketov;
- ne vsebuje ponavljanj⁹¹ in je konsistenten;
- prikazuje način izvedbe funkcionalnosti znotraj sistema, je osnova za načrtovanje;
- opredeljuje način realizacije primerov uporabe s strani sistema.

Tako se v koraku analize opredelijo razredi in njihove odgovornosti, ki jih zagotavljajo operacije, vmesniki ter paketi, ki bodo zagotavljali določene funkcionalnosti. Opredelijo se tudi povezave med razredi ter njihovo združevanje v pakete. Na miselnem nivoju se preizkusi delovanje primerov uporabe skozi tako opredeljeni model. Tako določeni razredi in paketi omogočajo izvajanje primerov uporabe. Opredelimo tudi povezavo med primeri uporabe in

⁹¹ sopomenka: redundanca

elementi modela analize, ki omogočajo njihovo izvajanje. Za opisovanje modela analize uporabljamo:

- razredni diagram,
- diagram sodelovanja.

Odvisnosti med paketi, ki realizirajo posamezne primere uporabe, prikažemo v razrednem diagramu na višjem nivoju abstrakcije, kjer prikažemo pakete in njihove medsebojne odvisnosti. Model analize je prikazan na konceptualnem nivoju, podrobnosti uresničitve niso prikazane.

Načrtovanje

V koraku načrtovanja opredelimo fizični model sistema, ki bo osnova za uresničevanje. Opredelimo model načrta in model porazdelitve. V modelu načrta podrobno opredelimo razrede, povezave med njimi in dinamično obnašanje sistema. Dokončno opredelimo attribute razreda, opredelimo vidljivost⁹² in opišemo metode. Metode lahko opišemo v naravnem jeziku ali pseudo kodi ter prepustimo programiranje naslednjemu koraku, lahko pa jih takoj programiramo v izbranem programskem jeziku. Povezave med objekti dokončno opredelimo glede števности, smeri navigacije in imenovanja vlog glede na predvideno okolje uresničitve. Prav tako je treba upoštevati predvideno tehnologijo shranjevanja podatkov, ko načrtujemo razrede, ki izvajajo zapisovanje podatkov v okviru transakcije. Pakete preoblikujemo v podsisteme. Treba je zagotoviti čim manjšo odvisnost med podsistemi, opredeliti ustrezne vmesnike in zagotoviti, da podsistem ponuja pravilno realizacijo operacij, ki so opredeljene z njegovimi vmesniki.

Dinamično obnašanje sistema podrobno opišemo z diagrami zaporedja, diagrami prehodov stanj in diagrami aktivnosti. Tako je model načrta podrobna predstavitev sistema z upoštevanimi lastnostmi okolja, v katerem bo uresničen.

Model porazdelitve vsebuje vozlišča, njihove lastnosti in povezave ter začetno razporejanje razredov in podsistemov v vozlišča. Za opisovanje uporabimo diagram porazdelitve.

Uresničitev

V koraku uresničitve izdelamo posamezne programske komponente, ki implementirajo razrede in podsisteme, ki smo jih opredelili v modelu načrta. Razrede implementiramo kot posamezne komponente, ki vsebujejo izvorno kodo. Uresničitev poteka v več ponavljanjih in vzporedno, zato je pomembno načrtovati integracijo po vsaki ponovitvi. Predvidene so ponovitve, v okviru katerih se izdelajo obvladljivi kosi kode, ki se sproti integrirajo z že razvitim delom sistema iz prejšnjih ponovitev. V okviru uresničitve izvedemo testiranje posameznih komponent, preden jih integriramo in namestimo na ustrezno strojno opremo - vozlišča. Integralni in sistemski test izvedemo v koraku testiranja. V modelu uresničitve uporabljamo diagrama komponent in porazdelitve. Glavni rezultat je koda, ki uresničuje izdelke predhodnih korakov in omogoča delovanje sistema.

Testiranje rezultatov

Namen testiranja je preizkus delovanja sistema, tako glede pokrivanja zahtevanih funkcionalnosti kot tudi pravilnosti delovanja in izpolnjevanja nefunkcionalnih zahtev. Testiranje na nivoju komponent je opravljeno v koraku uresničitve, v koraku testiranja

⁹² Vidljivost lahko opredelimo kot javno, privatno in omejeno (angleško: public, private, protected)

izvajamo integracijski in sistemski test. Za urejen način testiranja pripravimo model testiranja, ki vsebuje (Booch, Jacobson, Roumbaugh, 1999, str. 313):

- testne primere, ki opredeljujejo vsebino testiranja,
- testne postopke, ki opredeljujejo način testiranja,
- testne komponente, ki omogočajo avtomatizacijo testnih postopkov.

Treba je narediti načrt testiranja, ki opredeljuje urnik testiranja, in osebe, ki bodo testiranje izvajale. Po izvedenem testiranju ocenimo uspešnost testiranja. V primeru uspešnega testa celotnega sistema je sistem pripravljen za predajo. Model testiranja dopolnjujemo in ga uporabimo ob vsaki novi ponovitvi znotraj življenjske dobe sistema.

4.2.2.2. Glavne faze procesa

Začetna faza

V začetni fazi je treba opredeliti poslovno potrebo in obseg načrtovanega sistema. Opredeliti je potrebno glavne funkcionalnosti bodočega sistema. Opredelimo jih s primeri uporabe. Poiskati poskusimo vse primere uporabe, podrobno pa opredelimo le najvažnejše. V okviru primerov uporabe identificiramo akterje, ki bodo sodelovali s sistemom. Poleg definiranja funkcionalnosti je treba opredeliti potrebne vire in oceniti pričakovano trajanje in stroške izgradnje. Pripravimo začetni načrt projekta. Tako opredeljene stroške primerjamo s pričakovanimi koristmi, ki jih naj sistem prinese, in se odločimo za potrditev ali zavrnitev nadaljnjega razvoja.

Zbiranje informacij

V tej fazi podrobno opredelimo večino primerov uporabe in jih analiziramo. Pretežne aktivnosti te faze so dokončno opredeljevanje zahtev, analiza in načrtovanje. V okviru načrtovanja načrtujemo sistem kot celoto in upoštevamo posebnosti opreme, ki jo nameravamo uporabiti. Izvedemo tudi prototip do te mere, da lahko prikažemo delovanje najpomembnejših primerov uporabe. Preverimo realnost projektnega načrta, kot smo ga opredelili v začetni fazi, in ga po potrebi popravimo. Sproti je treba oceniti morebitna nova tveganja, ki so se pokazala z boljšim razumevanjem obravnavane problematike. Na koncu faze imamo pripravljene naslednje izdelke:

- opredeljena je večina primerov uporabe,
- modela analize in načrtovanja za večino primerov uporabe,
- model uresničitve in testiranja v obsegu prvega prototipa,
- dopolnjen načrt projekta in oceno tveganja.

Ko smo prepričani, da izdelani modeli napovedujejo uresničitev sistema, ki bo izpolnjeval zahtevane funkcionalnosti, nadaljujemo z naslednjo fazo.

Konstrukcija

V fazi konstrukcije je pretežni del aktivnosti izveden v okviru načrtovanja, uresničitve in testiranja. Fazo izvedemo v več ponavljanjih, znotraj katerih postopno dograjujemo sistem. V vsaki ponovitvi dodamo funkcionalnost nekaj primerov uporabe, izvedemo testiranje in integracijo. Uporabniki sodelujejo pri testiranju vsake ponovitve in lahko vplivajo še na

zadnje korekcije izdelanih funkcionalnosti. Ob izdelani programski opremi pripravimo tudi ustrezno tehnično in uporabniško dokumentacijo. Dopolnimo tudi vse nedokončane modele iz prejšnjih faz. Na koncu faze je potrebno oceniti, ali je programska oprema primerna za vsakodnevno uporabo in predajo uporabnikom.

Prevzem

V fazi prevzema predamo izdelek v uporabnikovo okolje. Opravimo še zadnja testiranja in popravke. Po zagonu se lahko pokažejo težave, ki jih v fazi testiranja nismo uspeli identificirati. V kolikor so odstopanja večja, lahko odpravo problema posredujemo v naslednjo ponovitev izdelave sistema. Manjše popravke izvedemo kar znotraj faze prevzema. Tako dopolnimo model uresničitve in testiranja ter zabeležimo pridobljene izkušnje za izboljšanje procesa v naslednjih ponovitvah.

Razvijanje programske opreme na opisan način omogoča kontinuirano dopolnjevanje in spreminjanje funkcionalnosti glede na spremenjeno poslovno okolje in uporabniške zahteve. Ključnega pomena je postopnost in iterativnost ter močna povezanost posameznih modelov, ki omogoča enostavno odkrivanje potrebnih sprememb v modelih, ki so bližji implementaciji, glede na spremembe v modelu primerov uporabe.

Metodologija omogoča modeliranje na dokaj različne načine z uporabo različnih kombinacij diagramov. Pri razvoju programske opreme s pomočjo te metodologije je neobhodna uporaba ustreznega CASE orodja, ki sproti preverja konsistentnost posameznih modelov ter sprejetje načelnih usmeritev glede uporabe posameznih diagramov, v različnih fazah izgradnje informacijskega sistema.

5. METODOLOGIJA »TABELARNI RAZVOJ APLIKACIJ« (TAD)

Metodologija »Tabelarni razvoj aplikacij⁹³« (v nadaljevanju TAD) predstavlja alternativo metodologijam, ki za modeliranje realnega sveta uporabljajo različne vrste diagramskih tehnik. Za preslikavo realnosti v model uporablja metodologija TAD niz različnih tabel, v katerih se opisuje delovanje sistema.

5.1. Osnovne značilnosti metodologije

Metodologija TAD je objektno orientirana metodologija, namenjena razvoju informacijskih sistemov. S pomočjo predpisanih korakov lahko pokrijemo vse faze razvoja informacijskega sistema. Poleg tega je v metodologijo vključena faza, ki predvideva prenovno poslovnih procesov s pomočjo analize vsebine ustreznih tabel. Faze metodologije so:

- definiranje problema,
- analiza delovanja sistema,
- prenova poslovnih procesov,
- objektni model sistema,
- načrtovanje sistema,
- implementacija sistema.

V fazah definiranja problema, analize in prenove poslovnih procesov postopno dopolnjujemo in preoblikujemo tabele, z analizo katerih v naslednjih treh fazah izdelamo objektni model sistema, načrt sistema in na koncu sistem tudi implementiramo.

5.2. Definiranje problema

V okviru faze definiranja problema je potrebno identificirati in opredeliti problem, ki ga obravnavamo. V ta namen izvedemo niz intervjujev s predstavniki vodstva podjetja, na različnih ravneh odločanja. Najprej izvedemo intervjuje z najvišjim vodstvom podjetja⁹⁴. Namen teh intervjujev je določiti obseg problema, informacijske potrebe strateškega vodstva podjetja in spoznati strategijo in strukturo podjetja. V okviru intervjujev najvišjega vodstva identificiramo (T. Damij, 2001, str. 21):

- strateški načrt in cilje organizacije,
- vitalne izhode in analize, povezane s strateškim načrtom in cilji,
- probleme podpore odločanja,
- strukturno shemo organizacije.

Ko smo opredelili problem s perspektive najvišjega vodstva na podlagi odobritve in pooblastila s strani vodstva organizacije, organiziramo niz intervjujev z vodilnimi predstavniki na poslovni in operativni ravni odločanja. S prehodom na nižje nivoje vedno bolj podrobno opredeljujemo problem in spoznavamo delovanje organizacije, njenih

⁹³ angleško: Tabular application development

⁹⁴ angleško: top management

organizacijskih enot in posameznikov. Namen intervjujev vodstvenih delavcev na poslovnem in operativnem nivoju je pridobivanje informacij o:

- poslovnih ciljih,
- operativnih ciljih,
- strukturi oddelkov – organizacijskih enot (entitet),
- vitalnih izhodih in analizah, povezanih s poslovnimi in operativnimi cilji,
- problemih podpore odločanja za potrebe posameznih entitet.

Pomembno je ugotoviti, kako se strateški cilji uresničujejo v poslovnih in operativnih ciljih. Metodologija TAD uporablja izraz »entiteta« za opredeljevanje uporabnika, skupine uporabnikov, organizacijske enote ali česarkoli pomembnega za delovanje sistema (T. Damij, 2001, str. 22). Entiteta je vir informacij in je lahko del sistema ali pa je v interakciji s sistemom. Tako jih ločujemo na notranje in zunanje. Zunanja entiteta lahko sodeluje v izvajanju aktivnosti skupaj z notranjo entiteto, ali pa s sistemom izmenjuje informacije. Intervjuje izvedemo s predstavniki notranjih entitet, ki nam razkrijejo tudi način, kako sodelujejo z zunanjimi entitetami.

Rezultate opredeljevanja problema, entitet, ciljev in potrebnih analiz ter odločitvenih problemov prikažemo v tabeli entitet. Analize nam zagotavljajo informacije, s katerimi merimo doseganje opredeljenih ciljev.

Tabela 1: Tabela entitet

		Index	1	2	3	4	5	6	7	8	9
Req.	Index	Entity Analysis	vodja sdc	vodja nabave	nabavni komercialist	vodja prevzema	skladiščnik	vodja prodaje	disponent	vodja komisijarjev	administrator v odpremi pisarni
DS	1	nadzor statusov za vse odpreme cone	*							*	
	2	nadzor planiranih voženj	*						*	*	*
	3	Predlog signalnih zalog in naročilnih količin			*						
	4	Odstopanje signalnih zalog od predlaganih		*							
	5	Odstopanje dejanskih dostav strankam od	*						*		
Report	6	pregled predvidenih prevzemov	*			*					
	7	pregled nerazporejenih kom. listov								*	
	8	pregled raporejenih kom. listov po vožnjah								*	
	9	pregled kom. listov po komisijarjih, kontrolorjih	*							*	*
	10	pregled opravljenega dela po komisijarjih	*							*	
	11	pregled opravljenega dela po prevzemnikih	*			*					
	12	Nabava po dobaviteljih		*							
	13	sumarni podatki-prodaja	*					*			
	14	Pregled odobrenih popustov v prodaji						*			
	15	Prodaja po kupcih in artiklih						*			
	16	Zaloga po skladiščih in artiklih		*	*						
	17	Zaloga po rokih trajanja	*	*	*		*				
	18	Odpisi po skladiščih in artiklih	*								

Tabela v stolpcih vsebuje entitete, v vrsticah pa potrebne analize, ki jih entitete potrebujejo za svoje delo. Zvezdica (*) v presečišču vrstice(i) in stolpca(j) označuje, da entiteta(j) potrebuje analizo(i). Tabela je iz praktičnega primera, obdelanega v naslednjem poglavju. V tabeli so prikazane samo tiste entitete, ki potrebujejo katero od analiz.

5.3. Analiza delovanja sistema

V tej fazi podrobno opredelimo delovanje sistema, saj lahko le s podrobnim razumevanjem delovanja uspešno načrtujemo spremembe procesov in izvedemo nadaljnje korake pri izgradnji informacijskega sistema. Treba je poznati delovanje sistema kot celote, delovanje posameznih delov sistema ter tudi posameznih skupin uporabnikov in posameznikov, ki bodo uporabljali bodoči informacijski sistem. V ta namen organiziramo niz intervjujev s predstavniki vseh notranjih entitet sistema, skupin uporabnikov in po potrebi posameznih uporabnikov. Namen intervjujev je spoznati elementarne aktivnosti sistema, njihovo povezavo v delovne procese ter združevanje delovnih procesov v poslovne procese. Fazo analize izvedemo v naslednjih korakih:

- opredelitev aktivnosti,
- opredelitev nalog v okviru opredeljenih aktivnosti,
- združevanje aktivnosti v delovne in poslovne procese.

Izdelka te aktivnosti sta tabela aktivnosti in tabela nalog.

5.3.1. Opredelitev aktivnosti

Pri opredeljevanju aktivnosti si pomagamo s posebno tabelo, imenujemo jo tabela aktivnosti. Tabela aktivnosti vsebuje v stolpcih notranje in zunanje entitete, v vrsticah pa so navedene posamezne aktivnosti. V presečišču vrstice(i) in stolpca (j) definiramo posamezno nalogo. Najprej definiramo pojma naloga in aktivnost (T. Damij, 2001, str. 27) :

- Naloga (angleško task) je elementarno delo, ki ga izvaja določena entiteta v okviru neke aktivnosti.
- Aktivnost je zbirka nalog, povezanih z določenim vhodom ali izhodom, in jo izvaja ena ali več entitet.

Aktivnost je tudi niz zaporednih nalog, ki spremenijo vhode v izhode, porabljajo vire, uporabljajo zmožnosti in sposobnosti ter se odzivajo na sistemske signale za večjo ali manjšo proizvodnjo izhodov (Watson, 1994, str. 68).

Vzporedno s tabelo aktivnosti kreiramo tudi tabelo nalog, v kateri podrobno opišemo naloge. V tabeli aktivnosti naloge med seboj povežemo vodoravno in navpično na način, kot si zaporedje aktivnosti sledi v realnem svetu. Povezave izvedemo na naslednji način:

- Vodoravno povezavo med aktivnostjo in entitetami, ki jo opravljajo prikažemo na način, da entiteto, ki sproži aktivnost, označimo z oznako S_j (S-source, izvor) v presečišču (i, j). Entiteto, ki prejme izhod izvirne entitete, označimo z T_j (T-target, cilj). Indeks j predstavlja zaporedno številko izvirne entitete v tabeli aktivnosti.

- Navpično povezavo med aktivnostmi, ki jih opravlja ista entiteta s P_i (P-predecessor, predhodnik), in U_i (successor, naslednik) glede na časovno zaporedje izvajanja aktivnosti v okviru entitete. Indeks i predstavlja zaporedno številko predhodne aktivnosti v tabeli aktivnosti.

Da lahko vzpostavimo te povezave, moramo sistem spoznati tako podrobno, da nam je popolnoma razumljivo zaporedje nalog, ki jih izvajajo posamezne entitete.

Tabela 2: Del tabele aktivnosti logistike

			Entity Internal												External		
			Index	9	10	11	12	13	14	15	16	17	18	20	23	24	
Business Process	Work Process	Index	Activity	vodja prodaje	prodajni referent	disponent	vodja komisinarjev	komisinar	kontrolor	administrator v odpremi pisarni	odpremnik	prevzemnik embalaže	referent za embalažo in reklamacije	dobavitelj	kupec	kupčev	
prodajanje	pripravljanje kupca	29	pridobitev kupca	T23 P29											S23		
		30	vnos matičnih podatkov kupca	S9, T9 U29, P30													
		31	sprememba matičnih podatkov kupca	T23 U30											S23		
	komercialni pogoji kupca	32	splošni pogoji kupca	S9, T9 U30													
		33	vnos skupine kupcev za cenik	S9, T9 P33													
		34	uvrstitev kupca v skupino za cenik	S9, T9 U33, P34													
		35	poseben cenik kupca	S9 U30, P35, U34, U36	T9 P35												
		36	nadziranje sprememb cen-popravljanje cenikov	S9, T9 U12, U35, P36													
	pridobivanje naročil kupcev	37	ponudba kupcu		S10 U35											T10	
		38	sprejem naročil kupcev		T23, T24, S10 P38	T10 P38										S23	S24
	nadzor izpolnitve naročila	39	pregled statusa naročila		S10, T10 U38, P39, U40												
		40	odgovor kupcu o predvideni izdobi		T23, S10 U39, P40											S23, T10	

Prikazan je del tabele aktivnosti iz praktičnega dela naloge. Zaradi omejenega prikaza so izpuščene nekatere entitete.

5.3.2. Opredelitev nalog

Sočasno s tabelo aktivnosti razvijamo tabelo nalog, v kateri podrobno opredelimo posamezne naloge. Naloge podrobno opredelimo za vsako neprazno križišče v tabeli aktivnosti. Vrstica tabele nalog predstavlja nalogo v okviru določene aktivnosti. Označimo jo s simbolom K_{ij} , kjer K pomeni nalogo, definirano v okviru aktivnosti, i in j pa sta indeksa aktivnosti in entitete v tabeli aktivnosti, ki označujeta nalogo. Za vsako nalogo v stolpcih opredelimo naslednje lastnosti:

- opis naloge, ki kratko in jedrnato opisuje izbrano nalogo,
- čas, ki je potreben za izvedbo naloge,
- pogoj, ki mora biti izvršen (eden ali več), da se naloga lahko izvede,
- vhode in izhode, ki jih entiteta potrebuje za izvedbo naloge, ali pa so rezultat izvedbe naloge.

Vsaka naloga zavzema v tabeli eno vrstico. Ker je aktivnost praviloma sestavljena iz več nalog, zavzema ena aktivnost več vrstic v tabeli nalog. Zato v prvem stolpcu navedemo tudi

aktivnost, v okviru katere se naloga izvaja, zaradi boljšega razumevanja povezav med aktivnostmi in nalogami.

Tabela 3: Del tabele nalog logistike

Activity	Task Code	Description	Time	Condition	Input/Output
29 - pridobitev kupca	K(29,9)	vodja prodaje pridobi kupca, preveri njegovo boniteto in registracijo		boniteta in registracija v redu	podatki o kupcu
30 - vnos matičnih podatkov kupca	K(30,9)	vodja prodaje vnese matične podatke kupca		kupec sprejet	podatki o kupcu
31 - sprememba matičnih podatkov kupca	K(31,9)	kupec posreduje spremembe podatkov v pisni obliki, vodja prodaje spremembe vnese v računalnik		kupec je v bazi	sprememba matičnih podatkov kupca
32 - splošni pogoji kupca	K(32,9)	vodja prodaje dogovori splošne pogoje (rok plačila, rabat) in jih vnese v bazo		kupec v bazi	splošni pogoji kupca
33 - vnos skupine kupcev za cenik	K(33,9)	Vodja prodaje vnese skupino kupcev, v katero se združuje kupce, ki dobijo isti cenik		skupina ne obstaja	skupina za cenik
34 - uvrstitev kupca v skupino za cenik	K(34,9)	vodja prodaje uvrsti kupce v skupino za isti cenik		skupina, kupec obstaja	skupina za cenik prirejena kupcu
35 - poseben cenik kupca	K(35,9)	Vodja prodaje po dogovoru s kupcem izdela poseben cenik izbranih artiklov za kupca / skupino kupcev. Cene vnese v bazo.		kupec, artikli, skupina kupcev obstajajo	cenik za skupino kupcev, kupec uvrščen v skupino
	K(35,10)	cenik za kupca dostopen v bazi			cenik za kupca
36 - nadziranje sprememb cen-popravljanje cenikov	K(36,9)	vodja prodaje spremlja evidenco predvidenih sprememb cen in po potrebi v dogovoru s kupcem popravlja cenike za kupce, ko je marža prenizka zaradi povečanja vhodnih cen		marža < želene	evidenca sprememb cen, cenik skupine kupcev
37 - ponudba kupcu	K(37,10)	prodajni referent izpiše in posreduje ponudbo kupcu		artikli, kupec, cenik obstajajo	ponudba kupcu
38 - sprejem naročil kupcev	K(38,10)	naročilo se prepíše iz kupčevega IS ali vnese s strani referenta		prispelo naročilo	naročilo vnešeno
	K(38,11)	naročilo v sistemu			naročilo v sistemu
39 - pregled statusa naročila	K(39,10)	za izbrano naročilo v IS pregleda status, v katerem je naročilo - čaka, razporejeno, v komisioniranju, v kontroli, v odpremi, odpremljeno		naročilo v sistemu	podatki o statusu naročila in računa
40 - odgovor kupcu o predvideni izdobi	K(40,10)	kupec poizveduje o statusu naročila, referent preveri in mu odgovori			informacija o predvideni dobavi

V tabeli so opredeljene naloge, ki ustrezajo prikazanemu delu tabele aktivnosti. V fazi analize delovanja sistema v več ponavljanih dopolnjujemo tabele aktivnosti in nalog, dokler nimamo dovolj podrobnega pogleda na obravnavani sistem.

5.3.3. Opredelitev delovnih in poslovnih procesov

Ko smo skupaj s predstavniki uporabnikov sistema dokončali dopolnjevanje tabele aktivnosti in tabele nalog, skupaj ponovno pregledamo tabelo aktivnosti, da se prepričamo o pravilnosti vzdolžnih in navpičnih povezav med nalogami. Ko smo prepričani, da nastale tabele odražajo dejansko stanje v resničnem svetu, ki ga modeliramo, je treba aktivnosti združevati v smiselne skupine, imenovane delovne procese.

Delovni proces je zbirka ene ali več aktivnosti, ki si sledijo v določenem vrstnem redu v izvajanju različnih nalog za izdelavo dobro definiranega izhoda (T. Damij, 2001, str. 37).

Delovni proces opredeljujejo vhodi, lastnik procesa, prevzemniki, omejitve, aktivnosti, dodana vrednost, strošek, čas, ključni dejavniki uspeha ter izhodi (Kovačič, 1998, str. 94).

Poslovni proces je torej primerna zbirka delovnih procesov (T. Damij, 2001, str. 37), prav tako je poslovni proces zbirka aktivnosti, ki pretvarja eno ali več vrst vhodov in kreira izhod,

ki predstavlja vrednost za stranko (Hammer, Champy, 2001, str. 38). Opredelimo ga lahko tudi kot vodoravni proces, ki povezuje različne funkcionalne aktivnosti, ki proizvajajo izloček podjetja (Watson, 1994, str. 36).

Tako lahko zaključimo, da delovni proces obsega eno ali več vrstic v tabeli aktivnosti. V nadaljevanju poskušamo združiti delovne procese na naslednjem višjem nivoju v enega ali več poslovnih procesov.

Tako združene aktivnosti ponovno pregledamo s predstavniki notranjih entitet, da se prepričamo v pravilnost vsebine tabel. Posebej pregledamo način združevanja aktivnosti v prej opisane procese. Na koncu faze potrdimo, da model sistema, opisan v tabelah entitet, aktivnosti in nalog, korektno prikazuje obravnavani sistem.

5.4. Prenova poslovnih procesov

Ko smo zaključili fazo analize delovanja sistema in podrobno razumemo sedanje delovanje sistema, imamo priložnost, da pred nadaljnjim razvojem informacijskega sistema ponovno preučimo in po potrebi spremenimo delovanje sistema. To nam omogoča model sistema v obliki tabel entitet, aktivnosti in nalog ter podrobno poznavanje obravnavanih procesov. Faza poteka v dveh korakih:

- analiza in prenova tabel entitet, aktivnosti in nalog,
- prevedba vsebine tabele aktivnosti v procesni model sistema.

Prenova poslovnih procesov⁹⁵ je nov pristop k izboljševanju delovanja podjetij in drugih organizacij, pomeni pa analiziranje in spreminjanje celotnega poslovnega procesa v podjetjih (Kovačič, 1998, str. 84). Opredelimo jo lahko kot temeljito preverjanje procesov, postopkov in aktivnosti in njihovo korenito spremembo, ki jo sprožimo z namenom, da bi dosegli pozitivne rezultate na področjih, kot so zniževanje stroškov, povečanje kakovosti izdelkov in storitev, skrajševanje dobavnih rokov in podobno (Kovačič, 1998, str. 90).

5.4.1. Analiza in prenova tabel

Prenovo poslovnih procesov v primeru uporabe metodologije TAD začnemo s podrobno analizo tabel entitet, aktivnosti in nalog. V ta namen ustanovimo projektni team, ki se sestoji iz analitika, predstavnikov vodstva podjetja ter vodilnih na poslovni in operativni ravni (T. Damij, 2001, str. 41). Projektni team analizira obstoječe cilje na ravni podjetja, poslovni in operativni ravni in jih po potrebi spremeni. Zaradi pomembnosti naloge je potrebno vključevanje najvišjih predstavnikov vodstva podjetja. Spremembe ciljev posledično povzročijo spremembo vsebine tabel, zaradi spreminjanja delovnih in poslovnih procesov. Podrobno se analizirajo odgovornosti posameznih notranjih entitet in razdelitev nalog med entitete. Tako v procesu prenove opravimo naslednje (T. Damij, 2001, str. 42) :

- opredelimo nove strateške cilje, spreminjamo ali umikamo obstoječe cilje,
- smiselno povežemo spremenjene strateške cilje z dodajanjem, spremembo ali ukinjanjem poslovnih procesov,

⁹⁵ angleško: Business process reengineering, BPR

- povežemo poslovne in delovne procese na način, ki omogoča prehod sprememb na operativno raven,
- ukinemo podvojene naloge in aktivnosti,
- prenesemo aktivnosti ali naloge od ene k drugi entiteti, če s tem dosežemo večjo učinkovitost izvajanja,
- skrajšujemo čas pri aktivnostih, ki so časovno potratne,
- ukinjamo nepotrebne entitete,
- ustanovljamo nove entitete, če je to potrebno.

Prenova tabel se izvaja v več ponovitvah. V vsaki ponovitvi projektni team dodatno optimira procese, dokler ne doseže želenega rezultata.

5.4.2. Procesni model

Procesni model je neobvezen korak v okviru metodologije TAD, pomaga pa pri boljši predstavi povezav med nalogami, ki so zabeležene v tabeli aktivnosti. Za predstavitev procesnega modela je uporabljen tehnika diagramov tokov podatkov (DTP, angleško data flow diagram – DFD). Elementi diagrama, ki se uporabljajo za predstavitev povezav znotraj tabele aktivnosti so:

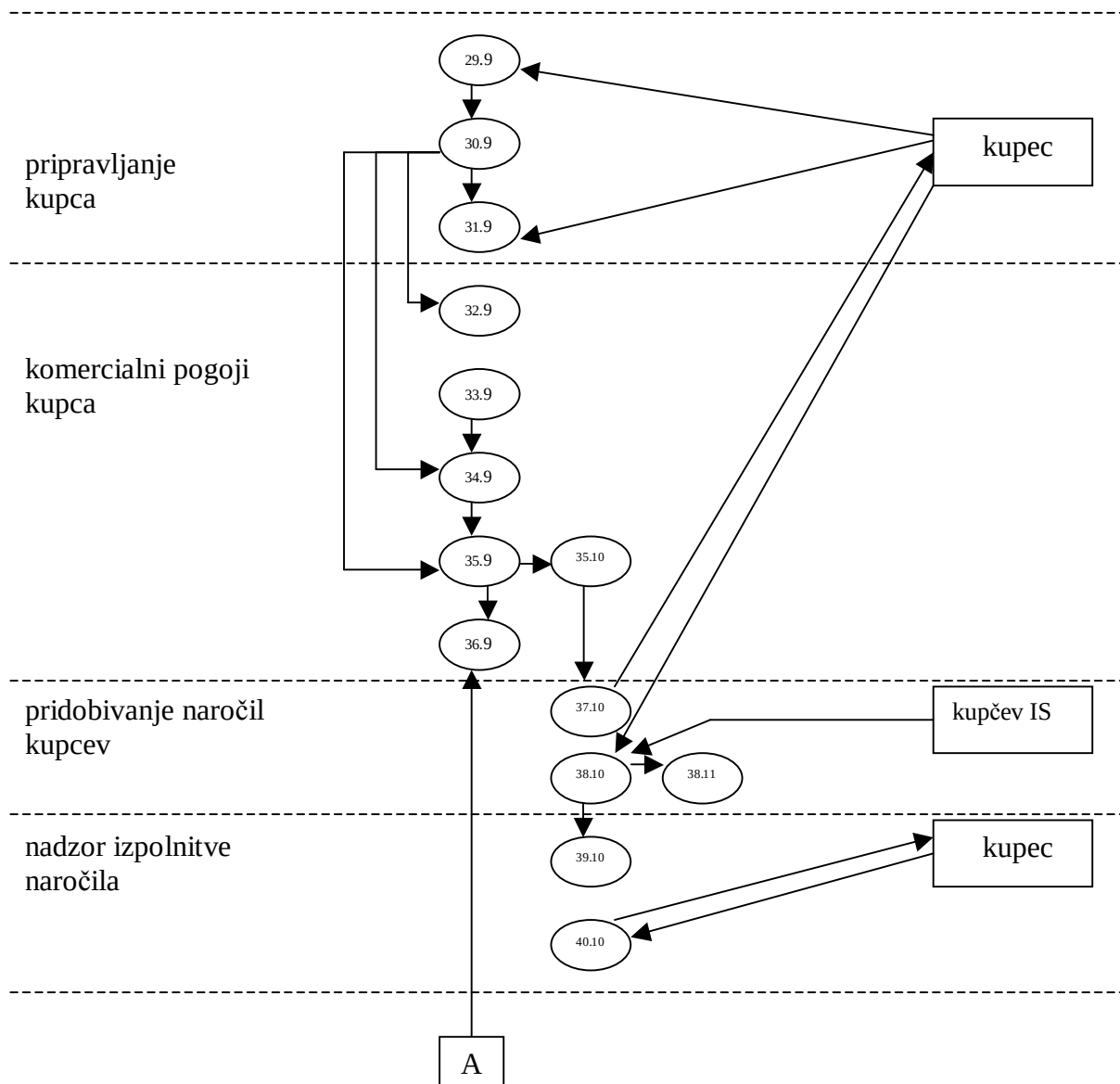
- pravokotnik – predstavlja vir, zunanjo entiteto, ki je povezana z obravnavanim sistemom, ter z njim izmenjuje določene podatke ali dokumente;
- oval – predstavlja proces, v našem primeru konkretno nalogo K_{ij} , ki jo izvaja entiteta j v okviru aktivnosti i ;
- usmerjena daljica – predstavlja tok podatkov od vira proti cilju, glede na zaporedje nalog, definirano v tabeli aktivnosti;

Procesni model oblikujemo iz tabele aktivnosti na naslednji način:

1. Vsako neprazno križišče (i, j) v tabeli aktivnosti pretvorimo v proces – oval, v kolikor je entiteta notranja, ali v vir – pravokotnik, če je entiteta zunanja. Procese označimo z indeksoma (i, j) , vir pa z imenom entitete(j), npr. kupec.
2. Procese povežemo z usmerjenimi daljicami vodoravno in navpično in sicer v smeri od S_j proti T_j pri vodoravni povezavi, in v smeri od P_i proti U_i pri navpični povezavi. Tako povežemo naloge v okviru iste aktivnosti, vrstice i (npr. S_1-T_1 , S_2-T_2), in navpično v okviru iste entitete, stolpca j (npr. P_1-U_1 , P_2-U_2).
3. Na koncu označimo še meje delovnih procesov in virov, ki pripadajo posameznemu procesu. Na podoben način zamejimo poslovne procese.

Celoten diagram predstavlja vsebino tabele aktivnosti, posamezni deli pa predstavljajo poslovne in delovne procese.

Slika 24: Del poslovnega procesa prodajanje



Slika 24 prikazuje del poslovnega procesa prodajanje, ki je opredeljen v predhodnih tabelah.

5.5. Objektni model sistema

V četrti fazi metodologije TAD je treba izdelati objektni model sistema. Fazo izvedemo v šestih korakih. V prvih petih korakih identificiramo objekte, njihove attribute in povezave – asociacije. Tako dobimo začetni objektni model obravnavanega sistema. V šestem koraku poiščemo takšne razrede, ki imajo skupne lastnosti, in jih hierarhično razporedimo z opredelitvijo dedovanja med razredi.

5.5.1. Razvoj začetnega objektnega modela

Kandidate za razrede objektov dobimo z analizo tabel entitet, aktivnosti in nalog. Posebej pomembna je tabela nalog v delu, ki opisuje vhode in izhode, navadno v obliki dokumentov. Vsak dokument podrobno analiziramo.

V **prvem koraku** opredelimo funkcionalne odvisnosti, ki obstajajo med atributi, ki enolično identificirajo dokument, in ostalimi atributi na dokumentu. Iz vsebine dokumenta razberemo identifikacijske attribute posameznih razredov, navadno so v obliki šifer dokumentov, artiklov, ipd. Nato poiščemo funkcionalne odvisnosti med ostalimi atributi, ki jih vsebuje dokument in izbranimi identifikacijskimi atributi.

V **drugem koraku** opredelimo povezave, ki obstajajo med posameznimi identifikacijskimi atributi. Opredelimo števnost povezave, torej ugotovimo, ali gre za povezavo v razmerju 1:1, 1:n, ali m:n. Povezavo m:n nadomestimo z dvema povezavama 1:n; kot identifikator novega razreda se pojavita združeno v sestavljen ključ oba identifikatorja razredov, ki sta bila povezana s povezavo m:n. Ko smo analizirali prvi dokument, nadaljujemo z analizo ostalih dokumentov.

V **tretjem koraku** integriramo analizo vseh obravnavanih dokumentov. Grupiramo funkcionalne odvisnosti vseh analiziranih dokumentov, ki so povezane z istimi identifikacijskimi atributi. Tako dobimo pregledno sliko povezav med vsemi obravnavanimi dokumenti.

V **četrtm koraku** zamenjamo imena identifikacijskih atributov z imeni njihovih razredov, povežemo vse attribute z njihovimi razredi in spremenimo povezave med razredi v ustrezno strukturo semantičnega mrežnega modela. Povezave pretvorimo v naslednji obliki povezav:

- »Je-povezan-z« (angleško: Is-associated-with) in
- »Je-del« (angleško: Is-part-of).

Identifikacijske attribute prav tako povežemo z njihovim razredom s povezavo »je-del«. Izbira med variantama »Je-povezan-z« in »je-del« je odvisna od vsebinskega razmerja med posameznimi razredi, ki opredeljuje razmerje celota-del, ali pa le prisotnost atributov določenega razreda na dokumentu.

V **petem koraku** pretvorimo rezultate dosedanjih korakov analize v začetni objektni model obravnavanega sistema.

5.5.2. Končni objektni model

V šestem koraku začetni objektni model dopolnimo z razredi, ki jih odkrijemo z dodatno analizo tabel aktivnosti, nalog in entitet ter z uporabo izkušenj in znanja analitika.

Z uporabo mehanizma dedovanja⁹⁶ razrede hierarhično razvrstimo v primeru, ko imajo skupne attribute ali povezave. Uporabimo tako združevanje v nadrazred – generalizacijo, kot tudi specializacijo, odvisno od razporeditve razredov v začetnem modelu. Na ta način dobimo končni objektni model. Začetni in končni objektni model sta prikazana v naslednjem poglavju.

5.6. Načrtovanje

V fazi načrtovanja pripravimo načrt sistema in pripravimo sistem za implementacijo. Fazo izvedemo v dveh korakih:

- opredelitev operacij,
- izdelava aplikacijskega modela.

5.6.1. Opredelitev operacij

Pri opredeljevanju operacij podrobno analiziramo tabelo nalog, vsako nalogo posebej. Za beleženje operacij kreiramo tabelo operacij (angleško: operation table). Vrstice tabele operacij predstavljajo posamezne operacije, v okviru naloge, v stolpcih pa beležimo:

- oznako naloge K_{ij} ,
- operacijo,
- razred, znotraj katerega je operacija.

Z analizo opisa, vhodov in izhodov ter pogojev iz tabele nalog identificiramo operacije, ki so potrebne za izvajanje posamezne naloge, pa tudi razred, ki jih vsebuje. Operacije poiščemo tako, da opredelimo povezavo med nalogo in razredom, ki se pojavlja kot vhod ali izhod naloge. Poleg operacij, ki so vezane na naloge imajo običajno vsi razredi operacije za kreiranje, spreminjanje in brisanje objektov. Za izvajanje ene naloge je lahko potrebna ena ali več operacij. V primeru, da je za eno nalogo potrebno več operacij, ena naloga zavzema več vrstic v tabeli operacij.

Ko smo identificirali in v tabelo operacij zapisali vse operacije, jih vnesemo še v končni objektni model, nato pa podrobneje definiramo algoritme vsake operacije. Algoritmi imajo naslednje lastnosti (T. Damij, 2001, str. 69) :

- Napisani so tako, da uresničujejo opis naloge, ki je podan v stolpcu opisa v tabeli nalog.
- Uresničujejo pogoje, povezane z zadevno operacijo, ki so opredeljeni v stolpcu pogojev tabele nalog.
- Določajo vhode in izhode operacije glede na vsebino stolpca vhodov in izhodov tabele nalog.

⁹⁶ povezava »je« (angleško is-a structure)

5.6.2. Aplikacijski model

Aplikacijski model zgradimo na podlagi analize tabele entitet in tabele aktivnosti z vpisanimi delovnimi in poslovnimi procesi. Model je sestavljen iz več delov.

Prva dva dela modela sta sestavljena iz poročil in odločitvenih problemov, ki jih potrebujejo oziroma rešujejo različni vodstveni nivoji v podjetju. Strukturo modela lahko razberemo iz tabele entitet, ki opredeljuje potrebna poročila in probleme odločanja.

Tretji del aplikacijskega modela razberemo iz prvega stolpca v tabeli aktivnosti. V aplikacijskem modelu strukturiramo poslovne procese, znotraj njih na nižjem nivoju delovne procese in na najnižjem nivoju aktivnosti. Tako dobimo pregledno strukturno sliko zahtev in odločitvenih problemov, pa tudi hierarhično shemo procesov in aktivnosti.

V nadaljevanju je za posamezne dele aplikacijskega modela treba definirati uporabniške vmesnike, ki bodo uporabnikom omogočili interakcijo s sistemom.

Tabela operacij in aplikacijski model sta prikazana v praktičnem delu naloge.

5.7. Implementacija sistema

V zadnji fazi metodologije TAD izvedemo fizično implementacijo sistema, ki smo ga razvijali v predhodnih fazah. Vhodi v fazo implementacije so:

- objektni model,
- aplikacijski model,
- algoritmi, ki smo jih opredelili v obeh modelih.

V fazi implementacije najprej izberemo ustrezno podatkovno bazo in programski jezik, v katerem bomo kodirali algoritme. Algoritme, opredeljene v objektnem in aplikacijskem modelu, kodiramo in prevedemo. V bazo podatkov ustrezno preslikamo opredeljene razrede.

Nazadnje opredelimo še pravice dostopa uporabnikov do posameznih funkcij sistema in analiz. Podatke o tem dobimo iz tabele entitet, ko gre za analize, in iz tabele aktivnosti, ko gre za naloge v okviru aplikacijskega modela.

Ko smo sistem implementirali, sledijo običajne aktivnosti:

- testiranja,
- izobraževanja uporabnikov in
- prehoda v delovanje.

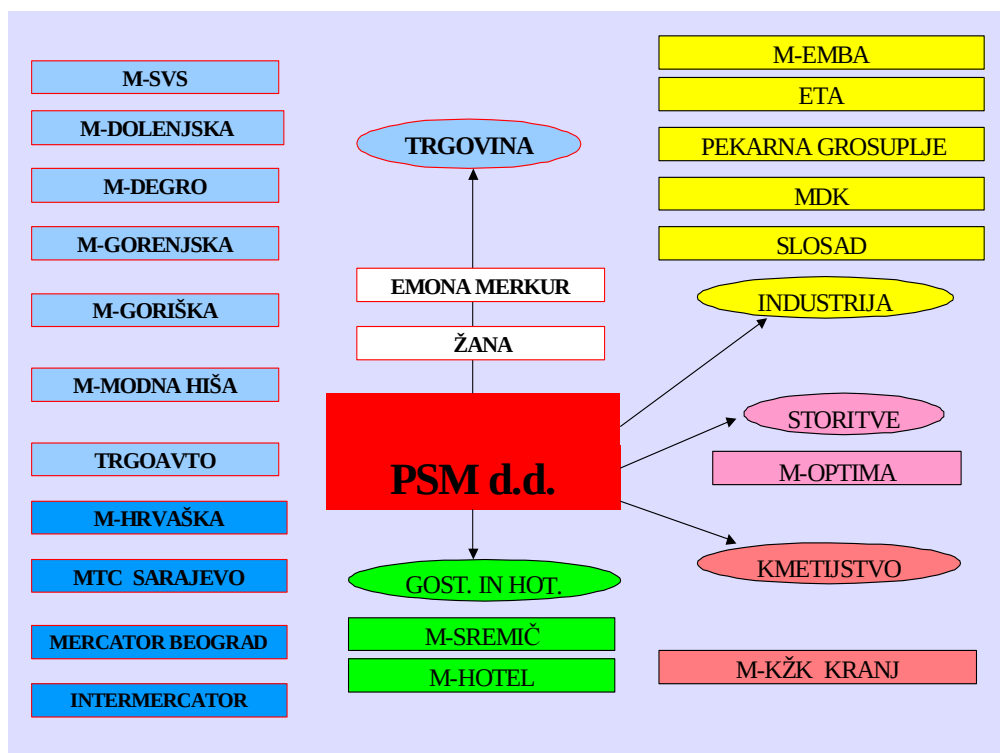
6. RAZVOJ INFORMACIJSKEGA SISTEMA LOGISTIKE

6.1. Predstavitev podjetja in namena delovanja distribucijskega centra

Podjetje Poslovni sistem Mercator, d.d. (krajše PSM d.d.) je delniška družba, ki v okviru Skupine Mercator obvladuje podjetja s področja trgovine, storitev, industrije ter pridelave hrane. Skupina Mercator zaposluje preko 13000 ljudi in je po prihodku trenutno največje podjetje v Sloveniji. Strateške usmeritve Skupine Mercator najbolje definira izjava o poslanstvu skupine:

Skupina Mercator je vodilna trgovska veriga na območju Slovenije in želi postati pomembna trgovska veriga na Hrvaškem, v Bosni in Hercegovini ter v Jugoslaviji, ki s kakovostjo svoje ponudbe učinkovito zadovoljuje potrebe in želje potrošnikov, hkrati pa se s povečevanjem poslovne učinkovitosti primerja z najuspešnejšimi evropskimi trgovskimi verigami in dolgoročno povečuje vrednost premoženja lastnikov, pri čemer izpolnjuje obveznosti do zaposlenih in vseh ostalih interesnih skupin.

Slika 25: Sestava Skupine Mercator



Pretežni del prihodkov skupine ustvarijo trgovske družbe, med katerimi je največja PSM d.d. Pretežna dejavnost PSM d.d. je prodaja na drobno v lastni maloprodajni mreži in mreži franšiznih prodajaln. Poleg prodaje na drobno je pomemben vir prihodkov tudi veleprodaja, katere kupci so iz trgovske in gostinske dejavnosti, pa tudi iz državnih ustanov preko izpolnjevanja javnih razpisov. Predmet trgovanja je trgovsko blago različnih blagovnih skupin, namenjenih končni potrošnji prebivalstva. S skupno oznako jih imenujemo »market program«. Poleg market programa, ki predstavlja pretežni del trgovanja, je v ponudbi

Mercatorja tudi »nemarket program«, ki vsebuje blago trajnejše uporabe v gospodinjstvu in materiale za nadaljnjo vgradnjo. Opredelimo lahko naslednje blagovne skupine:

1. market program

- o sadje in zelenjava
- o sveži program
 - meso in mesni izdelki
 - mleko in mlečni izdelki
 - zamrznjena hrana
- o živila dnevne potrošnje – suhi program
- o neživila za dnevno uporabo v gospodinjstvu

2. nemarket program

- o športna oprema
- o tekstilna galanterija
- o tehnični izdelki za dom
- o gradbeni material
- o pohištvo

Organizacijska struktura podjetja v področju trženja sledi razvrstitvi blagovnih skupin tako, da so znotraj področja trženja oblikovani sektorji, katerih naloga je skrb za izbrane blagovne skupine, od nabave do prodaje potrošniku.

Vsakodnevni stik s končnimi kupci v maloprodajnih enotah je posebnega pomena. Tako so vse trgovine nadzirane centralno, v okviru sektorja maloprodaje. Za pridobivanje veleprodajnih kupcev in pridobivanje njihovih naročil so zadolženi sodelavci v sektorju eksterne trgovine. Pri veleprodajnem poslovanju je ključnega pomena ocena kupčeve plačilne sposobnosti, kajti prodaja je uspešna le takrat, ko podjetje dobi tudi plačilo za prodano blago.

V podjetju ima ključno izvajalno vlogo sektor logistike, katerega naloga je zagotavljanje nemotenega toka blaga od dobaviteljev do maloprodajnih enot in v primeru veleprodajnega poslovanja do končnih kupcev.

Shema organiziranosti družbe Mercator, d.d. je prikazana na naslednji strani.

Slika 26: Organizacijska shema

Za potrebe oskrbovanja lastnih in franšiznih maloprodajnih enot ter distribucijo blaga veleprodajnim kupcem ima Mercator lastne skladiščno distribucijske centre, v katerih se prevzema in skladišči nabavljeno blago, ki se ga glede na naročila kupcev pripravi v ustrezne pošiljke naročenih artiklov. Pošiljke iz skladiščno distribucijskih centrov do kupcev dostavijo pretežno z lastnimi vozili, deloma pa tudi s pomočjo najetih prevoznikov.

Kot poslovno okolje problema naloge sem izbral centralni distribucijski center za market program v Ljubljani, ki dnevno sprejme in dostavi do 500 naročil različnim poslovnim enotam Mercatorja in zunanjim kupcem. Skupna dobavljena količina znaša od 300 do 600 ton blaga dnevno.

6.2. Umestitev obstoječega IS logistike v širši kontekst IS podjetja

Obstoječa informacijska podpora distribucijskega centra obsega podporo nabave, prodaje in logistike. Poslovni proces logistike obsega delovne procese prevzemanja blaga od dobaviteljev, skladiščenja in manipulacije z blagom ter pripravo blaga za kupca. Če poskusimo procese v logistiki združiti v smiselne skupine, dobimo dve glavni skupini procesov:

- izpolnjevanje nabavnih naročil, ki obsega pot blaga od dobavitelja do končnega uskladiščenja in priprave za prodajo,
- izpolnjevanje prodajnih naročil, ki obsega obdelavo naročil kupcev, pripravo in odpremo blaga kupcu.

Blago se kupcu dostavi s potrebno spremljajočo dokumentacijo.

Informacijski sistem podpira poleg navedenih poslovnih procesov tudi procese nabavljanja in prodajanja na operativnem komercialnem nivoju. Tako so poleg kreiranja vseh potrebnih matičnih podatkov podprti procesi nastavljanja cenikov in kalkulacij, proces obnavljanja zalog, proces zbiranja naročil ter notranji postopki v skladiščno distribucijskem centru namenjeni notranjim premikom blaga, odpisovanju, inventuri, ipd.

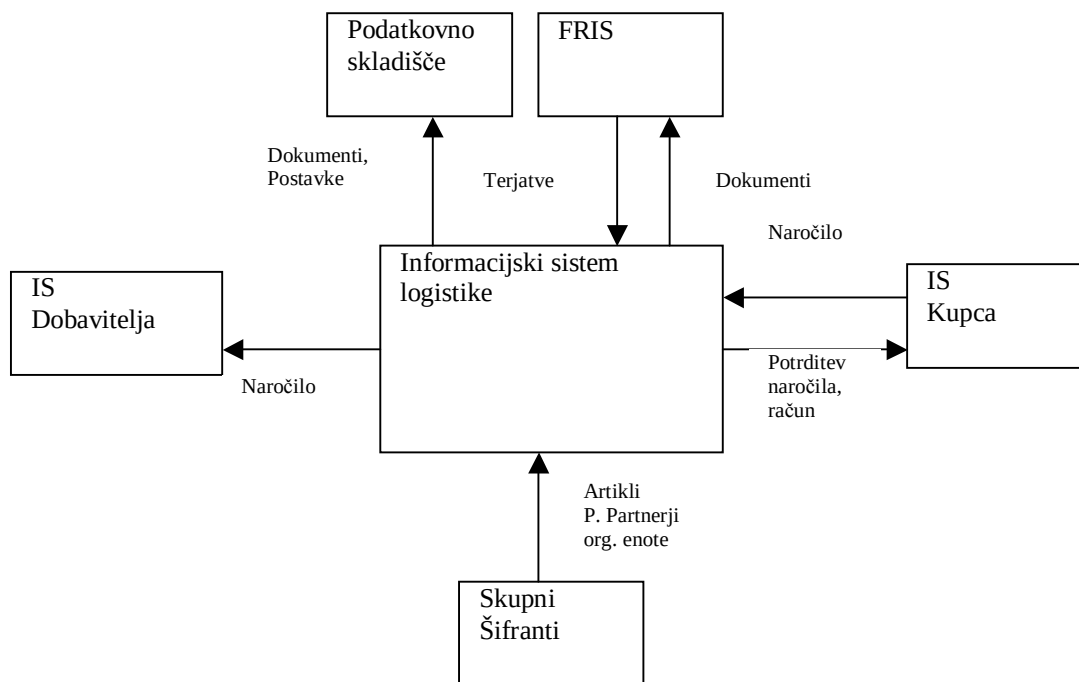
V širšem kontekstu informacijskega sistema podjetja je informacijski sistem logistike preko vmesnikov povezan z drugimi informacijskimi sistemi v podjetju, pa tudi izven podjetja. Tako je preko računalniške izmenjave podatkov po standardu EDIFACT povezan z informacijskimi sistemi največjih dobaviteljev in nekaterih velikih kupcev. Za zbiranje naročil iz lastne mreže poslovalnic ter prenos podatkov o dobavljenem blagu je v uporabi interni standard vmesnika in prenos preko internega sporočilnega sistema.

Registracija matičnih podatkov o artiklih, kupcih, dobaviteljih, organizacijskih enotah, zaposlenih, ipd. poteka s povezavo na centralne šifrate podjetja, ki so skupni vsem informacijskim sistemom.

V podatkovnem skladišču se zbirajo podatki o nabavi in prodaji blaga ter zalogah na podrobnem nivoju postavk dokumentov. Z agregacijo podrobnih podatkov po izbranih dimenzijah v posebne tabele so podatki uporabni za sprejemanje odločitev pri upravljanju z

blagovnimi skupinami. Podrobni podatki se zbirajo iz informacijskih sistemov distribucijskih centrov preko ustreznega vmesnika. Podatki na nivoju dokumentov se posredujejo v finančno računovodski informacijski sistem (FRIS), kjer se izvaja likvidacija računov dobaviteljev, plačevanje kupcev, vodenje poslovnih knjig in analize iz zbranih podatkov. V obratni smeri se iz finančno računovodskega IS posredujejo podatki o odprtih terjatvah do kupcev, ki so namenjene zadrževanju naročil kupcev, dokler ne izboljšajo plačevanja svojih obveznosti. Shematsko lahko prikažemo povezave informacijskega sistema logistike na naslednji način:

Slika 27: Povezave IS logistike z drugimi informacijskimi sistemi



Informacijski sistem za podporo logistike bo zaradi novih zahtev po sledljivosti blaga od proizvodnje do potrošnika potrebno dograditi na način, ki bo omogočal spremljanje posameznih količin prevzetega blaga do izdaje znanemu kupcu. V tem primeru gre za sledljivost živil živalskega izvora, za katere proizvajalci pridobijo posebno potrdilo veterinarske uprave, ki ga izda pristojni veterinar za vsako proizvodnjo serijo oziroma šaržo. Zahteva zakonodajalca predvideva sledenje podatkov o potrdilu ali proizvodnji seriji od proizvodnje, preko distribucijskega centra in trgovine do končne prodaje (neznane) potrošniku.

Obstoječi informacijski sistem predvideva vodenje zalog istega blaga na več skladiščih, ne omogoča pa spremljanja zaloge istega blaga v več delnih količinah (npr. paletah), ki bi jim lahko enoznačno pripisali podatek o proizvodnji seriji. Tovrstna zahteva pomeni večji poseg v zgradbo informacijskega sistema, ki jo je potrebno ustrezno načrtovati.

Obstoječa aplikacija je napisana v COBOL-u, podatki se hranijo v podatkovni bazi Pervasive SQL2000, na Novell strežniku. Glede na strateški načrt informatike v podjetju Mercator je

predvidena prenova informacijske podpore distribucijskega centra na osnovi modernejših tehnologij in uvedba nove enotne podpore v vseh distribucijskih centrih Mercatorja. Projekt se bo predvidoma pričel v drugi polovici leta 2002. V okviru projekta bo najprej opravljena odločitev o nakupu ali lastni izdelavi. Predvidena je zamenjava operacijskega sistema in baze podatkov.

V primeru odločitve za lastno izdelavo bomo informacijsko podporo tudi zgradili. V vmesnem času bo potrebno dograditi obstoječo podporo na način, ki bo omogočal zagotavljanje sledljivosti blaga do kupca. V ta namen bo uporabljena tehnologija prenosnih radio frekvenčnih terminalov (v nadaljevanju RF terminalov), s katerimi lahko omogočimo sprotno beleženje premikov posameznih palet blaga v distribucijskem centru, kar do sedaj ni bilo mogoče. Uvedba informacijske tehnologije v poslovni proces je eden od načinov prenove poslovnih procesov, ki je lahko (Kovačič, 1998, str. 99) :

- celovita ali strateška prenova poslovanja,
- preureditev ali prenova in informatizacija posameznih poslovnih procesov ali njihovih delov.

V primeru uvedbe sprotnega beleženja premikov blaga s pomočjo prenosnih RF terminalov se bodo bistveno spremenili postopki manipuliranja z blagom znotraj skladišča, predvideva se tudi zmanjšanje števila napak zaradi zamenjav blaga pri izdaji blaga.

Glede na navedena dejstva je treba analizirati obstoječi poslovni proces logistike v distribucijskem centru. Tovrstna analiza je nujno potrebna, ne glede na končni izbor variante realizacije novega informacijskega sistema. Za analizo procesov in načrt prenove informacijskega sistema je treba izbrati metodologijo, ki pokriva vse faze razvoja informacijskega sistema. Metodologija naj omogoča izgradnjo preglednega modela sistema, iz katerega so razvidni poslovni in delovni procesi tako, da je možno procese znotraj modela prenoviti in si ustvariti sliko bodočega sistema. V ta namen bo uporabljena metodologija TAD (T. Damij, 2001). Izgrajeni model naj omogoča:

- prikaz sprememb poslovnega procesa, ki so potrebne za zagotovitev zahtevane sledljivosti blaga,
- primerjavo procesov s procesi, ki jih podpira možna kupljena rešitev,
- nadaljnjo izgradnjo novega informacijskega sistema v primeru lastnega razvoja,
- boljše razumevanje delovanja obstoječega informacijskega sistema.

Z boljšim razumevanjem delovanja obstoječega informacijskega sistema je mogoče oceniti tudi velikost sprememb obstoječega informacijskega sistema, ki so potrebne za uvedbo spremenjenih procesov.

6.3. Definicija problema

Definicija problema predstavlja prvo izmed šestih faz metodologije TAD. V okviru te faze je potrebno spoznati problemsko domeno in poslovno okolje sistema dovolj podrobno, da lahko opredelimo naslednje (T. Damij, 2001, str. 21-22) :

Na strateškem nivoju:

- strateški načrt in cilje organizacije,
- vitalne izhode in analize, povezane s strateškim načrtom in cilji,
- probleme podpore odločanja,
- strukturno shemo organizacije.

Na poslovnem in operativnem nivoju:

- poslovne cilje,
- operativne cilje,
- vitalne izhode in analize, povezane s poslovnimi in operativnimi cilji,
- strukturo oddelkov – organizacijskih enot (entitet),
- probleme podpore odločanja posameznih entitet.

Način za pridobivanje informacij je uporaba metode intervjuja tako, da se za definiranje izhodov, pomembnih za strateški nivo, intervju izvede z vrhovnim poslovodstvom, za poslovni in operativni nivo pa z vodji ustreznih organizacijskih enot.

V obravnavanem primeru so bili organizirani intervjuji z vodstvom sektorja logistike in eksterne prodaje ter njihovimi odgovornimi sodelavci za posamezna področja. Glede na to, da je problem omejen na logistični podsistem in da so strateški cilji podjetja jasno zapisani v okviru sistema kakovosti ISO9001, je bil opravljen le krajši intervju s članom uprave, pristojnim za obravnavano področje. Rezultati intervjujev in proučevanja dokumentacije sistema kakovosti so prikazani v nadaljevanju.

Strateški cilji podjetja so naslednji:

1. povečevanje kakovosti ponudbe in zadovoljstva potrošnikov,
2. zagotavljanje rasti prodaje in povečevanje tržnega deleža,
3. povečevanje poslovne učinkovitosti,
4. dolgoročni partnerski odnosi s slovenskimi proizvajalci ter s proizvajalci s področja Hrvaške, Bosne in Hercegovine ter Jugoslavije,
5. zagotavljanje socialne varnosti zaposlenih,
6. zagotavljanje kakovostnih in cenovno konkurenčnih proizvodov in storitev v okviru netrgovske dejavnosti,
7. povezovanje s tujimi strateškimi partnerji,

8. povečevanje vrednosti premoženja lastnikov in zagotavljanje varnosti naložb posojilodajalcev,
9. skrb za okolje.

Strateški cilji so definirani tudi v okviru ciljev kakovosti (Poslovnik kakovosti QM-4-01-01).

Cilji kakovosti

Cilje določamo na osnovi trendov kupcev, makroekonomske situacije in konkurenčnosti ter meritev zadovoljstva kupcev in zaposlenih. Pri postavljanju ciljev upoštevamo dosežene rezultate iz preteklega leta. Dolgoročni cilji so:

- Popolno zadovoljstvo kupcev: Kupčeva pričakovanja želimo uresničiti, zato vedno in povsod prisluhnemo zahtevam in željam kupca, tako da je naša storitev vredna njegovega denarja in zaupanja.
- Popolno zadovoljstvo zaposlenih: Ustvariti možnost in vzdušje za delovno sodelovanje ter izpolniti sistem motivacije za ustvarjalno delo, kar je mogoče zagotoviti le z usklajenim delovanjem vseh sodelavcev. Zaposleni se morajo usposablјati in izobraževati, saj bodo le tako kos nalogam in novim izzivom.
- Popolno zadovoljstvo lastnikov: Povečevati vrednosti premoženja lastnikov in zagotavljati varnost naložb posojilodajalcev. Zagotoviti je potrebno rast prodaje in povečevanje tržnega deleža ter poslovno učinkovitost.
- Popolno zadovoljstvo poslovnih partnerjev in okolja: Graditi dolgoročni partnerski odnos s slovenskimi proizvajalci in povezovanje s tujimi strateškimi partnerji za uvoženo blago. Uspešno sodelovati z vsemi ostalimi interesnimi skupinami.

Načini doseganja kakovosti

Kakovost dosegamo z:

- Organiziranostjo poslovnega procesa: Sistem obvladovanja kakovosti je vgrajen v vse organizacijske enote Mercatorja. Vsaka organizacijska enota je odgovorna za uresničevanje ciljev na svojem delovnem področju. Neposredno odgovornost nosi vodja enote, vsak zaposleni pa je odgovoren za ustrezno kakovost dela. Dobro organiziran poslovni proces je jamstvo za kakovost.
- Optimizacijo stroškov poslovanja: Ena od pomembnih nalog vodstva Mercator je poslovanje z optimalnimi stroški. Za doseganje kakovosti je potrebno optimalno izkoristiti vse potenciale.
- Sodelovanjem zaposlenih: Vsi zaposleni dejavno sodelujejo in zagotavljajo odlično kakovost. Definirane so zahteve po kakovosti dela, delavci so usposobljeni in motivirani za opravljanje dela.
- Poslovanjem s kupci in dobavitelji: Osnovni cilj Mercatorja je izgradnja dolgoročnega poslovnega sodelovanja z našimi kupci in dobavitelji, ki temelji na medsebojnem spoštovanju, poštenosti, in upoštevanju kupčevih potreb in želja.

V nadaljevanju opredelimo **poslovne cilje logistike** :

- povečati poslovno učinkovitost, da bo primerljiva z najuspešnejšimi evropskimi trgovskimi verigami,
- učinkovito izkoristiti vse potencialne in tako optimizirati stroške poslovanja,
- razvoj mora potekati poenoteno, skladno z razvojno strategijo Mercatorja,
- temeljiti mora na skupnih standardih.

Nekatere **operativne cilje logistike** lahko opredelimo takole :

- vzpostaviti učinkovit in integralni informacijski sistem,
- naloga organizacijskih enot oz. poslovnih funkcij logistike je kvalitetna in pravočasna oskrba kupcev ob optimalnih stroških poslovanja,
- vzpostaviti učinkovit sistem nadzora nad artikli za zagotovitev zdravstvene ustreznosti živil
 - dobra higienska praksa,
 - nadzor po sistemu HACCP,
 - sistem sledljivosti.

V okviru strukture ciljev lahko prepoznamo potrebo po prenovi informacijskega sistema, prav tako pa tudi nerešen problem sistema sledljivosti blaga za zagotavljanje zdravstvene ustreznosti živil.

Organizacijska shema podjetja je prikazana v okviru točke 5.1.

Naslednja slika prikazuje strukturo procesov, ki so razdeljeni med naslednje **organizacijske enote (OE)** :

- OE, ki blago nabavljajo – programski sektorji,
- OE, ki blago prodaja in nadzira zbiranje ter izpolnjevanje naročil lastnih poslovalnic – eksterna prodaja,
- OE logistike, ki skrbi za izpolnjevanje nabavnih in prodajnih naročil.

Potrebni izhodi v obliki analiz in poročil ter problemi podpore odločanju so zbrani v tabeli entitet. Tabela entitet je sestavljena na naslednji način:

- V kolonah so navedene entitete. V okviru TAD metodologije je entiteta uporabnik ali skupina uporabnikov, oddelek ali karkoli ključnega za delovanje sistema. Entiteta je vir informacij, ki je lahko del sistema (interna entiteta), ali pa je v interakciji s sistemom (eksterna entiteta).
- V vrsticah so navedene potrebne analize in problemi podpore odločanju.
- V presečišču vrstice in kolone znak * pomeni, da entiteta kolone j potrebuje analizo, navedeno v vrstici i.

Glede na specifičnost obravnavane problematike je treba predstaviti vsaj naslednji entiteti:

- **Komisionar** je zaposleni v skladišču, ki je zadolžen za pripravo blaga, ki ga naročijo kupci. Naročilo kupca razpade na posamezne **komisionirne liste**, ki so v bistvu delovni nalogi za zlaganje blaga iz **lokacij v regalih – paletnih** mest v posebne **prevozne vozičke (angleško roll container – RLC)** ter dostavo tako pripravljenega blaga na vnaprej določeno **lokacijo na odpremni coni – boks**.
- **Disponent** je zaposleni v distribucijskem centru, ki je zadolžen za obdelavo prispelih naročil, razporejanje naročil, sestavljenih iz RLC na razpoložljive tovarnjake, določanje termina in smeri vožnje ter lokacije na odpremni coni, kjer bo pripravljeno blago čakalo na odpremo.

Tabela 4: Tabela entitet

Req.	Index	Index Entity Analysis	1 vodja sdc	2 vodja nabave	3 nabavni komercialist	4 vodja prevzema	5 skladiščnik	6 vodja prodaje	7 disponent	8 vodja komisionarjev	9 administrator v odpremni pisarni
DS	1	nadzor statusov za vse odpreme cone	*							*	
	2	nadzor planiranih voženj	*						*	*	*
	3	Predlog signalnih zalog in naročilnih količin			*						
	4	Odstopanje signalnih zalog od predlaganih		*							
	5	Odstopanje dejanskih dostav strankam od	*						*		
Report	6	pregled predvidenih prevzemov	*			*					
	7	pregled nerazporejenih kom. listov								*	
	8	pregled raporejenih kom. listov po vožnjah								*	
	9	pregled kom. listov po komisionarjih, kontrolorjih	*							*	*
	10	pregled opravljenega dela po komisionarjih	*							*	
	11	pregled opravljenega dela po prevzemnikih	*			*					
	12	Nabava po dobaviteljih		*							
	13	sumarni podatki-prodaja	*					*			
	14	Pregled odobrenih popustov v prodaji						*			
	15	Prodaja po kupcih in artiklih						*			
	16	Zaloga po skladiščih in artiklih		*	*						
	17	Zaloga po rokih trajanja	*	*	*		*				
	18	Odpisi po skladiščih in artiklih	*								

6.4. Analiza delovanja sistema

Faza analize delovanja sistema je sestavljena iz treh korakov. Najprej je potrebno identificirati aktivnosti, nato identificiramo in opredelimo naloge v okviru prikazanih aktivnosti. V tretjem koraku smiselno združujemo aktivnosti v delovne procese in delovne procese v poslovne procese. Za podrobno opredelitev aktivnosti in nalog je treba opraviti serijo intervjujev s predstavniki posameznih entitet, v okviru katerih podrobno spoznamo funkcioniranje sistema kot celote, funkcioniranje posameznih delov sistema in delo skupin uporabnikov, po potrebi tudi posameznega uporabnika (T. Damij, 2001, str. 25).

V obravnavanem primeru smo opravili intervjuje s ključnimi uporabniki obstoječega sistema, v nekaterih primerih je bilo opravljenih več intervjujev z istim uporabnikom, v več iteracijskih ciklikih.

Pred nadaljevanjem je treba opredeliti nekatere pojme, ki se bodo pojavljali v okviru tabel aktivnosti in nalog in so opredeljeni v okviru organizacijskega predpisa logistike.

Definicije pojmov in kratic

Termin - pojma/kratica	Razlaga
vhodna dokumentacija	prednaročilo dobavitelju in dobaviteljeva dobavnica
komisionirna dokumentacija	komisionirni list, delovni nalog za pripravo blaga
odpremna dokumentacija	dobavnica-račun in nakladalna lista
transportna dokumentacija	potni nalog
odpremni prostor - box	prostor na odpremni coni, kjer je blago, ki je pripravljeno za odpremo oziroma transport
skladiščni prostor	prostor v skladišču, ki je razdeljen na komisionirno in rezervno cono
komisionirna cona	del skladiščnega prostora, namenjen za komisioniranje (lokacije, paletna mesta, regali, police, ...)
rezervna cona	prostor v skladišču, namenjen za shranjevanje blaga (rezerva)
logistika	je upravljanje blagovnih tokov v času in prostoru in z njimi povezanih informacijskih tokov z namenom zagotovitve oskrbe kupcev
SDC	skladiščno distribucijski center
SSCC koda	serijska številka kontejnerja pošiljke (Serial Shipping Container Code ali SSCC) je standard sistema EAN, namenjen identifikaciji individualnih transportnih pakiranj (EAN International, 2000)
umaknjeno blago iz prometa	blago, ki ga je potrebno po direktivi (komerciale, inšpekcije,...) umakniti iz prometa

6.4.1. Tabela aktivnosti

V okviru predhodno omenjenih intervjujev se v več iteracijah izpolnjujejo hkrati tabela aktivnosti in tabela nalog. Tabela aktivnosti vsebuje v stolpcih entitete (notranje in zunanje), v vrsticah pa so navedene posamezne aktivnosti. Aktivnosti povežemo vodoravno in navpično na naslednji način:

- Vodoravno povezavo med aktivnostjo in entitetami, ki jo opravljajo, prikažemo na način, da entiteto, ki sproži aktivnost, označimo z oznako S_j (S-source, izvor) v presečišču (i, j). Entiteto, ki prejme izhod izvirne entitete, označimo s T_j (T-target, cilj). Indeks j predstavlja zaporedno številko izvirne entitete v tabeli aktivnosti.

- Navpično povezavo med aktivnostmi, ki jih opravlja ista entiteta, označimo s P_i (P-predecessor, predhodnik) in U_i (Successor, naslednik), glede na časovno zaporedje izvajanja aktivnosti v okviru entitete. Indeks i predstavlja zaporedno številko predhodne aktivnosti v tabeli aktivnosti.

V okviru analize delovanja obravnavanega sistema smo identificirali 18 notranjih in 6 zunanjih entitet ter 101 aktivnost.

6.4.2. Tabela nalog

Hkrati z tabelo aktivnosti razvijemo tabelo nalog, v kateri podrobno opredelimo posamezne naloge. Vrstica tabele predstavlja nalogo v okviru aktivnosti. Označimo jo s simbolom K_{ij} , kjer K pomeni nalogo, definirano v okviru aktivnosti, i in j pa sta indeksa vrstice in kolone v tabeli aktivnosti, ki označujeta nalogo. Za vsako nalogo v stolpcih definiramo:

- opis naloge, ki kratko in jedrnato opisuje izbrano nalogo,
- čas, ki je potreben za izvedbo naloge,
- pogoj, ki mora biti izvršen (eden ali več), da se naloga lahko izvede,
- vhode in izhode, ki jih entiteta potrebuje za izvedbo naloge, ali pa so rezultat izvedbe naloge.

V nadaljevanju so prikazane tabele aktivnosti in del tabel nalog obravnavanega sistema. Tabela nalog je v celoti prikazana v prilogi dela.

(tabela5.pdf)

Tabela 5: Tabela aktivnosti - obstoječe stanje 1/2

(tabela6.pdf)

Tabela 6: *Tabela aktivnosti – obstoječe stanje 2 / 2*

Tabela 7: Tabela nalog - obstoječe stanje 1/4

Activity	Task Code	Description	Time	Condition	Input/Output
1 - vnos matičnih podatkov dobavitelja	K(1,2)	na osnovi sklenjene pogodbe vodja nabave posreduje podatke o dobavitelju nabavnemu komercialistu		sklenjena pogodba	podatki o dobavitelju
	K(1,3)	nabavni komercialist vnese matične podatke dobavitelja			podatki dobavitelja
2 - spremembe podatkov o dobavitelju	K(2,2)	Dobavitelj sporoči spremembo podatkov, npr. naslova. Vodja nabave posreduje podatke komercialistu			podatki o dobavitelju
	K(2,3)	komercialist vnese spremenjene podatke dobavitelja		dobavitelj v bazi	podatki o dobavitelju
3 - pridobitev ponudbe novih artiklov	K(3,2)	vodja sprejme predlog novih artiklov in ponudbo dobavitelja			predlog novih artiklov
	K(3,3)	nabavni komercialist prejme ponudbo novih artiklov, predlaga izbor in ga posreduje vodji nabave		dobavitelj v bazi	ponudba
4 - odobritev novih artiklov	K(4,2)	vodja nabave odobri nove artikle			odobren predlog
	K(4,3)	nabavni komercialist sprejme odobreni predlog novih artiklov			odobreni predlog
5 - vnos matičnih podatkov artikla	K(5,3)	nabavni komercialist vnese matične podatke artikla		odobren nov artikel	matični podatki artikla
6 - vnos osnovne kalkulacije	K(6,3)	nabavni komercialist vnese osnovno kalkulacijo		artikel v bazi	kalkulacija artikla
7 - pridobitev cenika dobavitelja	K(7,2)	vodja nabave sprejme cenik			cenik dobavitelja
	K(7,3)	dobavitelj najavi spremembo cen, nabavni komercialist najavo posreduje vodji nabave			cenik dobavitelja
8 - odobritev sprememb cen	K(8,2)	Vodja nabave odobri spremembo cen ali pa intervenira pri dobavitelju. Odločitev posreduje nabavnemu komercialistu		ali je napoved pravočasna (1 mesec)?	cenik, predvidena uveljavitev
	K(8,3)	nabavni komercialist sprejme odobren/zavržen cenik dobavitelja			cenik dobavitelja
9 - vnos sprememb kalkulacije	K(9,3)	nabavni komercialist vnese spremenjeno kalkulacijo		odobrena sprememba	kalkulacija artikla
10 - pridobitev predloga ukinjanja artikla	K(10,3)	dobavitelj posreduje napoved ukinitve artikla			napoved ukinjenih artiklov
11 - sprememba statusa aktivnosti artikla	K(11,3)	nabavni komercialist spremeni status artikla		artikel v bazi	status artikla
12 - obvestilo o spremembi asortimana/cen	K(12,2)	vodja nabave sprejme obvestilo			obvestilo o spremembi asortimana/cen
	K(12,3)	nabavni komercialist obvesti vodjo nabave in prodaje o napovedanih spremembah		spremenbe	obvestilo o spremembah asortimana in cen, za prihodnje obdobje, evidenca predvidenih sprememb cen
12 - obvestilo o spremembi asortimana/cen	K(12,9)	vodja prodaje prejme obvestilo			obvestilo o spremembah asortimana/cen
13 - analiza pretekle prodaje	K(13,3)	nabavni komercialist izpiše in preuči gibanje prodaje skupin artiklov, za katere je zadožben			analiza prodaje in predlog nivojev signalnih in maksimalnih zalog
14 - nastavljanje signalnih zalog	K(14,3)	nabavni komercialist na osnovi predloga nastavi in ročno popravi signalne in maksimalne zaloge		predlog drugačen od trenutnih vrednosti	predlog nivojev zalog, signalna, maksimalna zaloga artikla na skladišču
15 - kreiranje predloga - signalne zaloge	K(15,3)	glede na trenutne zaloge nabavni komercialist kreira predlog naročila		zaloga <= signalni	predlog naročila
16 - korekcija naročila	K(16,3)	glede na zunanje dejavnike popravi predlog naročila			naročilo dobavitelju
17 - potrditev in posredovanje naročila	K(17,3)	komercialist pošlje prednaročilo dobavitelju in v skladišče		potrjeno prednaročilo	stiskano prednaročilo v skladišču, naročilo pri dobavitelju
	K(17,4)	vodja prevzema prevzame prednaročila in jih razporedi po datumih		stiskana prednaročila v skladišču	izpisano prednaročilo dobavitelju
18 - dodeljevanje prednaročila prevzemniku	K(18,4)	vodja prevzema na določen datum preda prednaročila prevzemnikom		datum=datum predvidenega prevzema	
	K(18,5)	prevzemnik prevzame prednaročilo na podlagi katerega opravi prevzem blaga			prednaročilo dobavitelju
19 - sprejem dobaviteljevega tovarnjaka	K(19,5)	prevzemnik sprejme dobaviteljev tovarnjak, za katerega ima razpisano prednaročilo dobavitelju.		prednaročilo za ta prevzem	prednaročilo dobavitelju
20 - količinsko in kakovostno preverjanje blaga	K(20,5)	prešteje količino, preveri deklaracije, rok uporabnosti, fizične lastnosti blaga		blago na prevzemni rampi	prevzeta Q vpisana na prednaročilu
21 - vračanje praznih palet dobavitelju	K(21,5)	prevzemnik vrne enako število praznih palet kot je prevzel polnih			
22 - podpisovanje dokumentov	K(22,5)	prevzemnik podpiše dobaviteljevo dobavnico,		vse blago je prevzeto	dobavnica dobavitelja
23 - preverjanje in vnos manjkajčih log. podatkov	K(23,5)	Prevzemnik preveri log. podatke in vnese popravke oz. manjkajoče podatke		ali podatki manjkajo?	podatki vnešeni v IS
24 - označevanje palete za skladiščenje	K(24,5)	Prevzemnik na paletni označevalni listek napiše datum, kom. lokacijo		blago prevzeto	označevalni listek napisan
	K(24,7)	viličarist prevzame označevalni listek in blago			označevalni listek na paleti
25 - vskladiščenje palete na lokacijo	K(25,7)	Viličarist paleto odpejle na komisionirno lokacijo.		prazna kom. lok. če polna: rezervna lok+karton	evidenčni kartonček: rezervna lokacija, količina, rok

6.4.3. Opredelitev delovnih in poslovnih procesov

Ko smo skupaj s predstavniki uporabnikov sistema dokončali dopolnjevanje tabele aktivnosti in tabele nalog, skupaj ponovno pregledamo tabelo aktivnosti, da se prepričamo o pravilnosti vzdolžnih in navpičnih povezav. Ko smo prepričani, da nastale tabele odražajo dejansko stanje v resničnem svetu, ki ga modeliramo, je potrebno aktivnosti združevati v smiselne skupine, imenovane delovni proces.

Tako združene aktivnosti ponovno pregledamo s predstavniki notranjih entitet, da se prepričamo v pravilnost vsebine tabel. Posebej pregledamo način združevanja aktivnosti v prej opisane procese.

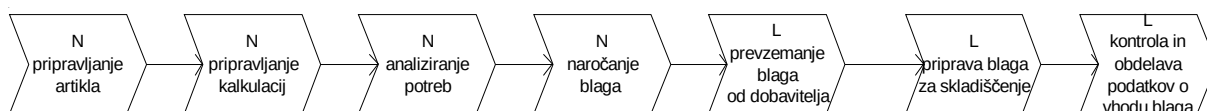
V obravnavanem primeru smo identificirali naslednje delovne in poslovne procese :

Poslovni proces nabavljanja blaga vsebuje delovne procese

- pripravljanje dobavitelja,
- pripravljanje artiklov,
- pripravljanje kalkulacij,
- izločanje artiklov iz asortimana,
- obveščanje o spremembah,
- ocenjevanje nabavnih potreb,
- naročanje blaga dobavitelju,
- prevzemanje blaga od dobavitelja,
- priprava blaga za skladiščenje,
- kontrola in obdelava podatkov o vhodu blaga.

Osnovni tok procesa je prikazan na sliki, delovni procesi označeni z »N« spadajo v organizacijsko enoto nabave, procesi, označeni z »L«, pa v logistiko.

Slika 28: Prikaz procesa nabavljanja



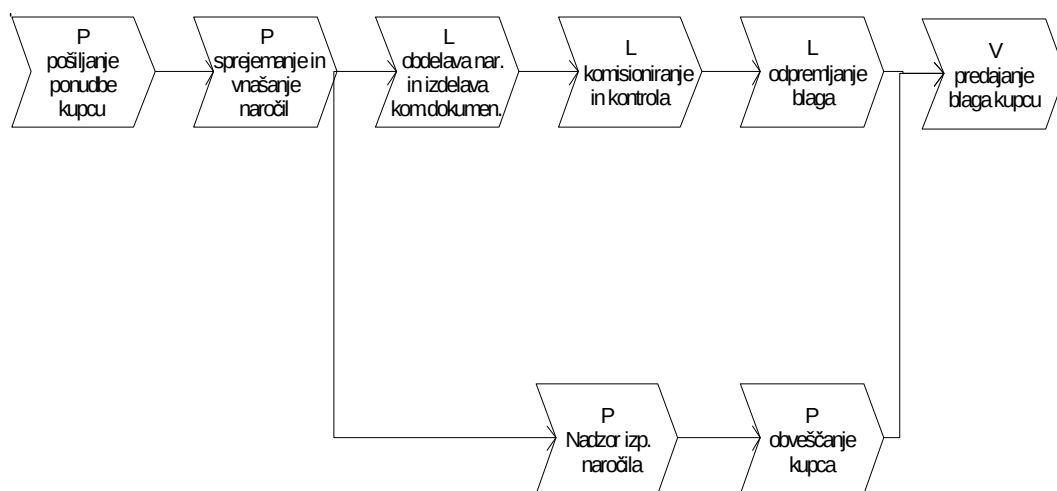
Poslovni proces prodajanja blaga vsebuje delovne procese

- pripravljanje kupca,
- komercialni pogoji kupca,
- pridobivanje naročil kupcev,

- nadzor izpolnjevanja naročila,
- obdelovanje naročil,
- komisioniranje,
- odpremljanje blaga,
- zaključevanje vožnje,
- vračanje blaga kupcev.

Osnovni tok procesa je prikazan na sliki, delovni procesi označeni s »P« spadajo v organizacijsko enoto prodaje, procesi, označeni z »L«, pa v logistiko.

Slika 29: Prikaz procesa prodajanja



Proces predajanja blaga kupcu izvaja zunanja entiteta voznik. Posebej pomemben je nadzor izpolnjevanja naročila, ki poteka vzporedno z glavnim procesom priprave blaga.

Poslovni proces izvajanja notranjih aktivnosti vsebuje delovne procese:

- premiki blaga med lokacijami,
- izločanje neustreznega blaga,
- odstranjevanje neustreznega, neželenega blaga.

Tako opredeljeni delovni in poslovni procesi so vneseni v tabelo aktivnosti.

6.5. Prenova poslovnih procesov

Prenovo poslovnih procesov opravimo, ko smo dokončali izdelavo tabel, ki opisujejo sedanje stanje (angleško as-is). V fazi prenove podrobno analiziramo tabele aktivnosti in nalog ter poskušamo identificirati priložnosti za optimiranje poslovnih in delovnih procesov. Tehnike, s katerimi si pomagamo pri prenovi poslovnih procesov (angleško business process reengineering – BPR), so naslednje (Taylor, 1995, str.107) :

1. poenostavitev operacij z izločanjem nepotrebnih odobritev, poročil in drugih organizacijskih aktivnosti,
2. skrajševanje časa poslovnega cikla vseh poslovnih procesov podjetja, povečanje odzivnosti in zniževanje stroškov,
3. povečanje dodane vrednosti vsake operacije s postopnim dvigom kakovosti proizvedenega izdelka ali storitve,
4. zniževanje stroškov posameznih operacij na način, ki ne poslabšuje kakovosti ali dobavnih rokov,
5. povečanje zanesljivosti poslovnih procesov, doslednosti v procesih in kakovosti izdelkov ter storitev,
6. urejanje relacij z dobavitelji na način, ki izboljšuje delovanje procesov, vezanih na zunanje vire,
7. osredotočenje na lastne ključne sposobnosti in prenos izvajanja procesov, v katerih nismo najboljši (best of breed) na zunanje izvajalce.

Pri izvedbi prenove poslovnih procesov je nujna podpora vodstva podjetja.

6.5.1. Analiza in prenova tabel

Za analizo in prenovu tabel organiziramo projektni team, ki poleg analitika vključuje predstavnike vodstvenega, poslovnega in operativnega nivoja podjetja. Ponovno se analizira struktura ciljev na vseh treh nivojih podjetja, podrobno pregledajo tabele in po potrebi spremenijo na način, ki bo omogočal take poslovne procese, ki bodo izpolnjevali spremenjene cilje. Posledično se v tabelah ukinjajo in spreminjajo aktivnosti, spreminjajo se izvajalci posameznih aktivnosti, brišejo in po potrebi se dodajajo nove entitete. Prav tako se aktivnosti lahko združujejo v delovne in poslovne procese na nov način.

V obravnavanem primeru uvedba drugačne informacijske tehnologije omogoča spremembo načina izvajanja aktivnosti v vseh delovnih procesih, ki so povezani z manipulacijo z blagom v skladišču. Tako so spremenjeni procesi:

- Prevezemanje blaga od dobavitelja, kjer se namesto papirne oblike prednaročila uporablja neposredni vnos količin preko RF terminala in s tem odpravlja naknadni vnos prevzetih količin.

- Priprava blaga za skladiščenje, kjer se spremeni:
 - o način označevanja palet, zamenja se ročno napisan list s tiskano SSCC kodo, ki predstavlja temelj sistema sledljivosti blaga v sistemu, kar odgovarja novo definiranimu cilju zagotavljanja zdravstvene neoporečnosti živil,
 - o način beleženja premikov blaga v skladišču z uporabo RF terminala namesto pisanja evidenčnih kartončkov lokacij, kar omogoča centralno evidenco rokov blaga na zalogi,
 - o opisani spremembi se analogno uporabita v vseh postopkih poslovnih procesov prodajanja in izvajanja notranjih aktivnosti, kjer gre za označevanje blaga in beleženje premikov blaga znotraj sistema.
- Komisioniranje, kjer se spremeni:
 - o način razporejanja in zadolževanja komisionirnih listov, ki so sedaj v elektronski obliki,
 - o aktivnost priprave postavke, kjer se dejansko identificira paleta, s katere bo vzeto blago, ter se podatki o seriji in roku trajanja vežejo na postavko računa,
 - o zaključevanje komisioniranja in kontrola, kjer s kontrolnim tehtanjem identificiramo le tiste RLC, ki niso v tolerančnih mejah za podroben pregled s strani kontrolorja. V starem procesu so se kontrolirali vsi RLC, pripravljeni za odpremo, kar je predstavljalo zahtevo po večjem številu kontrolorjev.
- Odpremljanje, kjer postopek odpremljanja s pomočjo nakladalne liste in prepisovanja zadolženih transportnih enot v računalnik zamenja neposredna identifikacija odpremljenih RLC z RF terminalom.
- Izločanje neustreznega blaga, kjer se fizično pregledovanje barvnih datumskih oznak na paletnih listih zamenja z izpisom ustreznega poročila iz računalniške evidence.
- Odstranjevanje neustreznega blaga, kjer se pisanje na papirne dokumente predloga odpisa ali vračila blaga dobavitelju zamenja z direktnim vnosom s pomočjo RF terminala.

Opisane spremembe aktivnosti in delovnih procesov ustrezajo tehnikam BPR od 1 do 5, saj:

- izločajo nepotrebne aktivnosti,
- skrajšujejo poslovni cikel z izločanjem nepotrebne kontrole in tako znižujejo stroške,
- dvigajo kakovost opravljene storitve, saj se s pomočjo vsakokratne identifikacije izdanega blaga zmanjša možnost za zamenjave blaga. Posledično se zmanjša število reklamacij kupcev, kar spet znižuje stroške, pa tudi povečuje ugled podjetja pri kupcih.
- Vzpostavljeni sistem sledljivosti predstavlja dodano vrednost za kupca, ki je tako lahko prepričan v zdravstveno neoporečnost živil, kupljenih v obravnavanem podjetju.

Prenovljene tabele aktivnosti in nalog so prikazane v nadaljevanju.

6.5.2. Procesni model

Procesni model je neobvezen korak v okviru metodologije TAD, pomaga pa pri boljši predstavi povezav, ki so zabeležene v tabeli aktivnosti. Za predstavitev procesnega modela je uporabljena tehnika diagramov tokov podatkov (DTP, angleško data flow diagram – DFD).

V nadaljevanju sledijo prikazi prenovljenih tabel aktivnosti, prvi del prenovljene tabele nalog in procesnega modela obravnavanega sistema. Prenovljena tabela nalog in procesni model sta v celoti prikazana v prilogi dela.

(tabela8.pdf)

Tabela 8: Prenovljena tabela aktivnosti 1 / 2

(tabela9.pdf)

Tabela 9: Prenovljena tabela aktivnosti 2/2

Tabela 10: Prenovljena tabela nalog 1 / 4

Activity	Task Code	Description	Time	Condition	Input/Output
1 - vnos matičnih podatkov dobavitelja	K(1,2)	na osnovi sklenjene pogodbe vodja nabave posreduje podatke o dobavitelju nabavnemu komercialistu		sklenjena pogodba	podatki o dobavitelju
	K(1,3)	nabavni komercialist vnese matične podatke dobavitelja			podatki dobavitelja
2 - spremembe podatkov o dobavitelju	K(2,2)	Dobavitelj sporoči spremembo podatkov, npr. naslova. Vodja nabave posreduje podatke komercialistu			podatki o dobavitelju
	K(2,3)	komercialist vnese spremenjene podatke dobavitelja		dobavitelj v bazi	podatki o dobavitelju
3 - pridobitev ponudbe novih artiklov	K(3,2)	vodja sprejme predlog novih artiklov in ponudbo dobavitelja			predlog novih artlov
	K(3,3)	nabavni komercialist prejme ponudbo novih artiklov, predlaga izbor in ga posreduje vodji nabave		dobavitelj v bazi	ponudba
4 - odobritev novih artiklov	K(4,2)	vodja nabave odobri nove artikle			odobren predlog
	K(4,3)	nabavni komercialist sprejme odobreni predlog novih artiklov			odobreni predlog
5 - vnos matičnih podatkov artikla	K(5,3)	nabavni komercialist vnese matične podatke artikla		odobren nov artikel	matični podatki artikla
6 - vnos osnovne kalkulacije	K(6,3)	nabavni komercialist vnese osnovno kalkulacijo		artikel v bazi	kalkulacija artikla
7 - pridobitev cenika dobavitelja	K(7,2)	vodja nabave sprejme cenik			cenik dobavitelja
	K(7,3)	dobavitelj najavi spremembo cen, nabavni komercialist najavo posreduje vodji nabave			cenik dobavitelja
8 - odobritev sprememb cen	K(8,2)	Vodja nabave odobri spremembo cen ali pa intervenira pri dobavitelju. Odlučitev posreduje nabavnemu komercialistu		ali je napoved pravočasna (1 mesec)?	cenik, predvidena uveljavitev
	K(8,3)	nabavni komercialist sprejme odobreni/zavrjeni cenik dobavitelja			cenik dobavitelja
9 - vnos sprememb kalkulacije	K(9,3)	nabavni komercialist vnese spremenjeno kalkulacijo		odobrena sprememba	kalkulacija artikla
10 - pridobitev predloga ukinjanja artikla	K(10,3)	dobavitelj posreduje napoved ukinitve artikla			napoved ukinjenih artiklov
11 - sprememba statusa aktivnosti artikla	K(11,3)	nabavni komercialist spremeni status artikla		artikel v bazi	status artikla
12 - obvestilo o spremembi asortimana/cen	K(12,2)	vodja nabave sprejme obvestilo			obvestilo o spremembi asortimana/cen
	K(12,3)	nabavni komercialist obvesti vodjo nabave in prodaje o napovedanih spremembah		spremembe	obvestilo o spremembah asortimana in cen, za prihodnje obdobje, evidenca predvidenih sprememb cen
	K(12,9)	vodja prodaje prejme obvestilo			obvestilo o spremembah asortimana/cen
13 - analiza pretekle prodaje	K(13,3)	nabavni komercialist izpiše in preuči gibanje prodaje skupin artiklov, za katere je zadožen			analiza prodaje in predlog nivojev signalnih in maksimalnih zalog
14 - nastavljanje signalnih zalog	K(14,3)	nabavni komercialist na osnovi predloga nastavi in ročno popravi signalne in maksimalne zaloge		predlog drugačen od trenutnih vrednosti	predlog nivojev zalog, signalna, maksimalna zaloga artikla na skladišču
15 - kreiranje predloga - signalne zaloge	K(15,3)	glede na trenutne zaloge nabavni komercialist kreira predlog naročila		zaloga <= signalni	predlog naročila
16 - korekcija naročila	K(16,3)	glede na zunanje dejavnike popravi predlog naročila			naročilo dobavitelju
17 - potrditev in posredovanje naročila	K(17,3)	komercialist pošlje prednaročilo dobavitelju in v skladišče		potrjeno prednaročilo	potrjeno prednaročilo v bazi, naročilo pri dobavitelju
	K(17,4)	vodja prevzema pregleduje evidenco izdanih prednaročil			izpisano prednaročilo dobavitelju, prednaročilo v bazi
18 - dodeljevanje prednaročila prevzemniku	K(18,4)	vodja prevzema na določen datum dodeli prednaročilo prevzemnikom		datum=datum predvidenega prevzema	vnos prevzemnika na prednaročilo
	K(18,5)	prevzemnik z RF terminalom izbere prednaročilo prispelega dobavitelja			prednaročilo dobavitelju
19 - sprejem dobaviteljevega tovornjaka	K(19,5)	prevzemnik sprejme dobaviteljev tovornjak, za katerega ima razpisano prednaročilo dobavitelju.		prednaročilo za ta prevzem	prednaročilo dobavitelju
20 - količinsko in kakovostno preverjanje blaga	K(20,5)	prešteje količino, preveri deklaracije, rok uporabnosti, fizične lastnosti blaga		blago na prevzemni rampi, postavka obstaja	skupna prevzeta količina vnešena, postavka potrjena
21 - vračanje praznih palet dobavitelju	K(21,5)	prevzemnik vrne enako število praznih palet kot je prevzel polnih			
22 - podpisovanje dokumentov	K(22,5)	prevzemnik podpiše dobaviteljevo dobavnico,		vse blago je prevzeto	dobavnica dobavitelja
23 - preverjanje in vnos manjkajočih log. podatkov	K(23,5)	Prevzemnik preveri log. podatke in vnese popravke oz. manjkajoče podatke z RF		ali podatki manjkajo?	podatki vnešeni v IS
24 - kreiranje postavk delnih količin	K(24,5)	za vsako postavko prevzema kreiramo delne postavke za vsako paletu, dodelimo SSCC kode za palete		skupna količina = vsota delnih	delne postavke: SSCC, količina, rok trajanja, serija
25 - označevanje palete za skladiščenje	K(25,5)	prevzemnik izpiše označevalne SSCC nalepke in jih nalepi na palete		delne postavke kreirane	SSCC nalepke
	K(25,7)	viličarist skenira SSCC, odčita lokacijo, ki se določi glede na prostost komisijne lokacije in rok trajanja			SSCC nalepka

6.6. Objektni model sistema

V četrti fazi metodologije TAD je treba izdelati objektni model sistema. Fazo izvedemo v šestih korakih, ki sledijo v nadaljevanju.

6.6.1. Začetni objektni model

V **prvem koraku** opredelimo funkcionalne odvisnosti, ki obstajajo med atributi, ki enolično identificirajo dokument, in ostalimi atributi na dokumentu. Dokumente, ki jih bomo analizirali, identificiramo s pomočjo analize tabele nalog v stolpcu, ki opredeljuje vhode in izhode posameznih nalog.

V obravnavanem primeru najprej analiziramo dokument Naročilo, ki se uporablja za naročanje kupcev distribucijskega centra. Naročilo enolično opredeljuje kombinacija atributov #Skladišče in #NarŠt, saj se naročila zbirajo ločeno po skladiščih. Prodajna cena artikla je lahko za vsakega kupca drugačna, glede na datum dobave in veljavni cenik skupine kupcev, ki ji kupec pripada.

Z analizo dokumenta Naročilo lahko opredelimo naslednje funkcionalne odvisnosti med identifikacijskimi atributi in ostalimi atributi, prisotnimi na dokumentu:

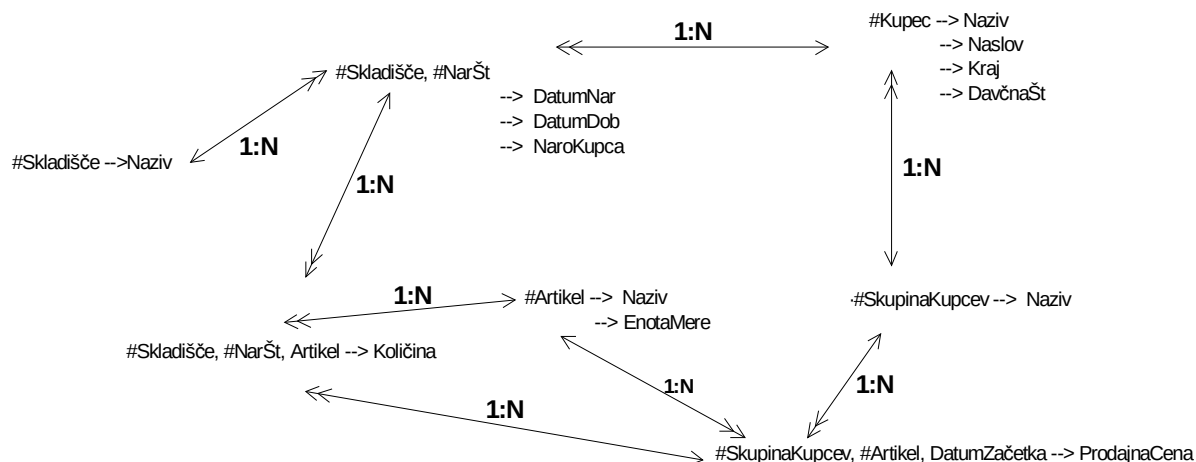
Slika 31: Funkcionalne odvisnosti med atributi

#Artikel	→ Naziv → EnotaMere
#Kupec,	→ Naziv → Naslov → Kraj → DavčnaŠt
#Skladišče, #NarŠt	→ DatumNar → DatumDob → Količina → NaroKupca
#SkupinaKupcev, #Artikel, DatumZačetka	→ ProdajnaCena
#SkupinaKupcev	→ Naziv
#Skladišče,	→ Naziv

V **drugem koraku** opredelimo povezave, ki obstajajo med posameznimi identifikacijskimi atributi. Opredelimo števnost povezave, s čimer ugotovimo, ali gre za povezavo v razmerju 1:1, 1:n ali m:n. Povezavo m:n nadomestimo z dvema povezavama 1:n. V obravnavanem primeru imamo povezavo m:n med razredoma Naročilo in Artikel, zato opredelimo nov razred PostNaročila, ki ga identificira kombinacija atributov #Skladišče, #NarŠt in #Artikel.

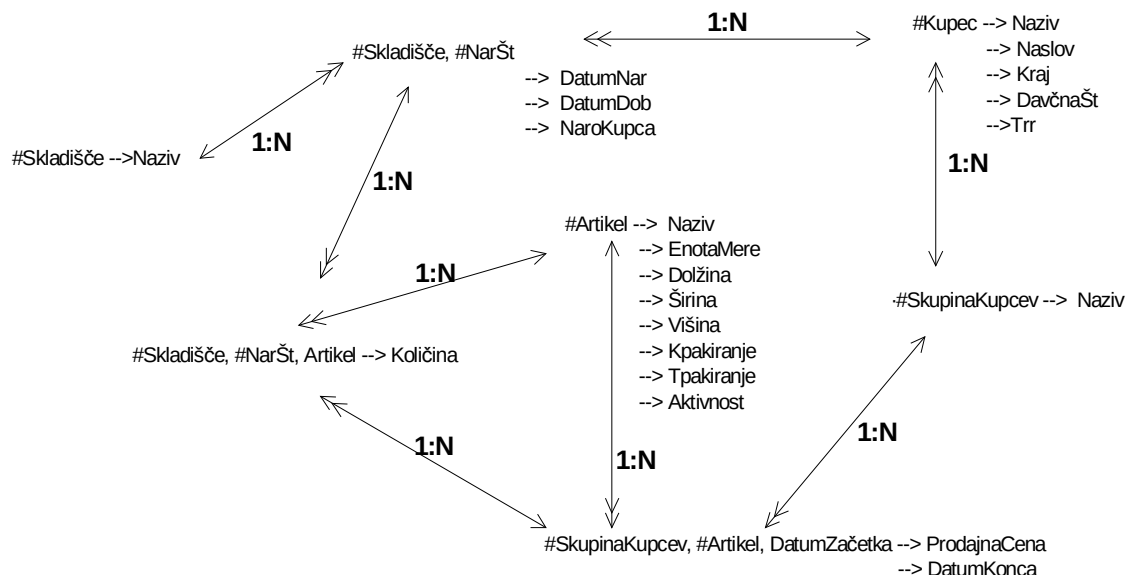
Tako dobimo naslednjo shemo povezav med identifikacijskimi atributi:

Slika 32: Povezave med identifikacijskimi atributi



V tretjem koraku integriramo analizo vseh obravnavanih dokumentov. Grupiramo funkcionalne odvisnosti vseh analiziranih dokumentov, ki so povezane z istimi identifikacijskimi atributi. Tako v obravnavanem primeru dopolnimo manjkajoče attribute, ki niso bili uporabljeni v dokumentu Naročilo. Dodani so manjkajoči atributi artikla, ki se nanašajo na pakiranje in logistične podatke, končni datum veljavnosti cene za kupca ter kupčev transakcijski račun. Tako dobimo integrirano shemo analize.

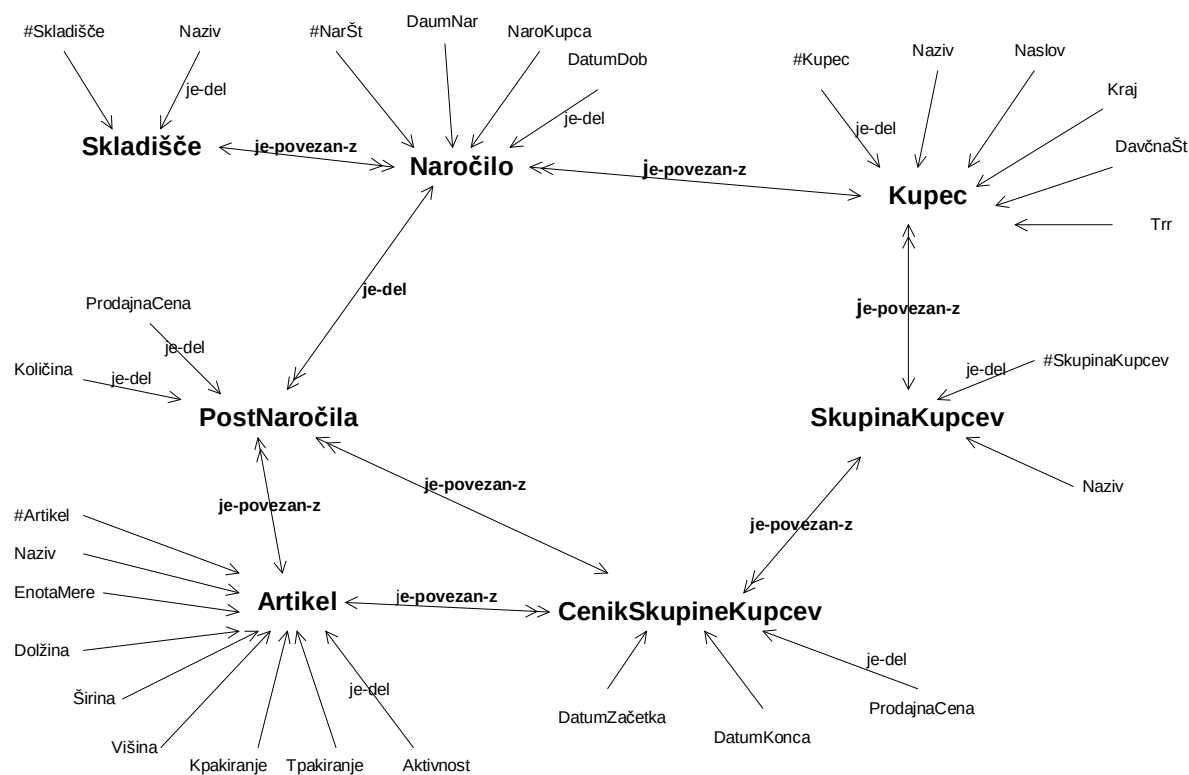
Slika 33: Integracija analize



V četrtem koraku zamenjamo imena identifikacijskih atributov z imeni njihovih razredov, povežemo vse attribute z njihovimi razredi in spremenimo povezave med razredi v oblike povezav »Je-povezan-z« ali »Je-del«. Identifikacijske attribute prav tako povežemo z

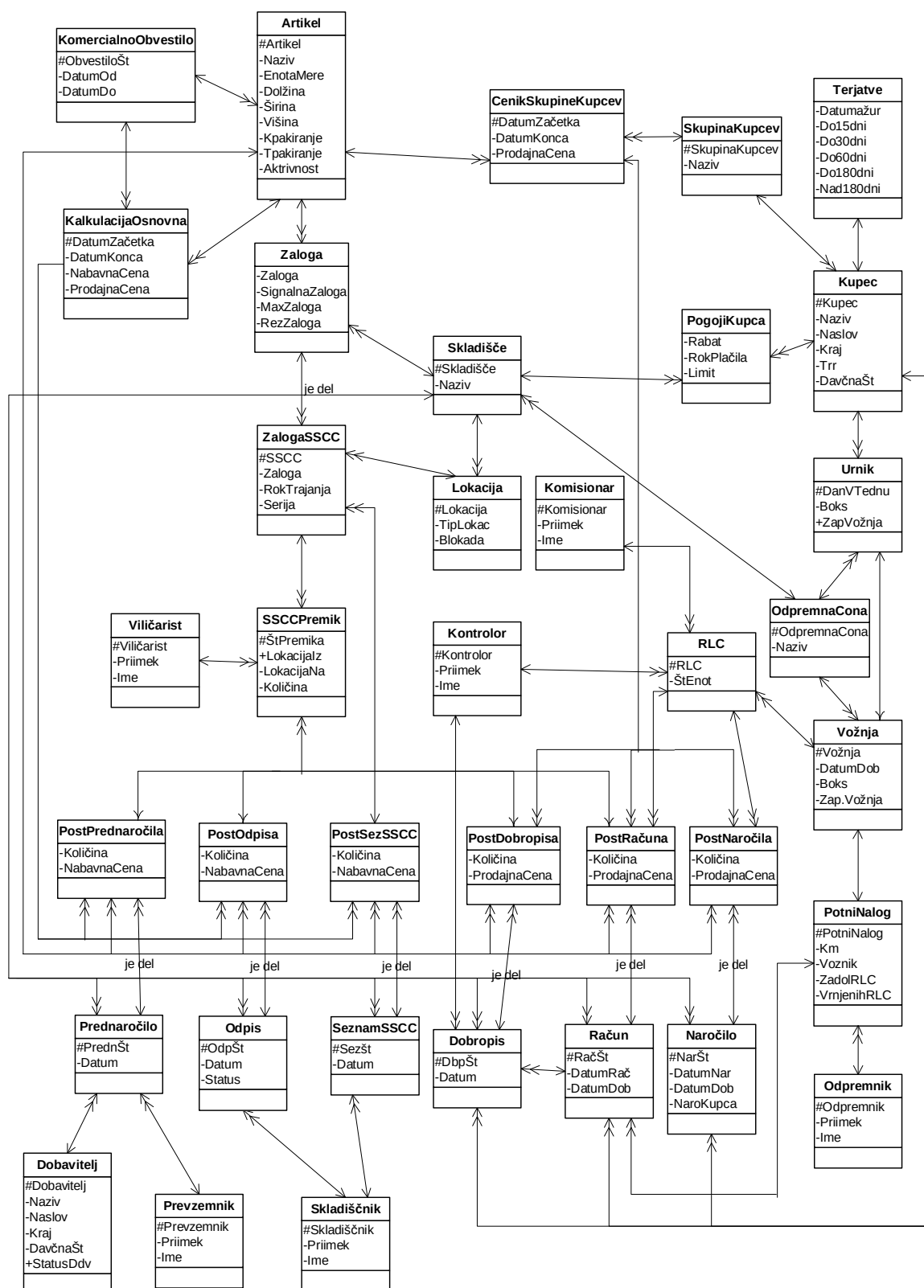
razredom, ki ga določajo s povezavo »Je-del«. Tako dobimo opredelitev razredov in povezav med njimi.

Slika 34: Razredi in njihove medsebojne povezave



V petem koraku pretvorimo rezultate dosedanjih korakov analize v začetni objektni model obravnavanega sistema. Podrobno je skozi prve štiri korake analiza prikazana le za del objektnega modela. Celotni začetni objektni model je prikazan na naslednji strani.

Slika 35: Začetni objektni model



6.6.2. Končni objektni model

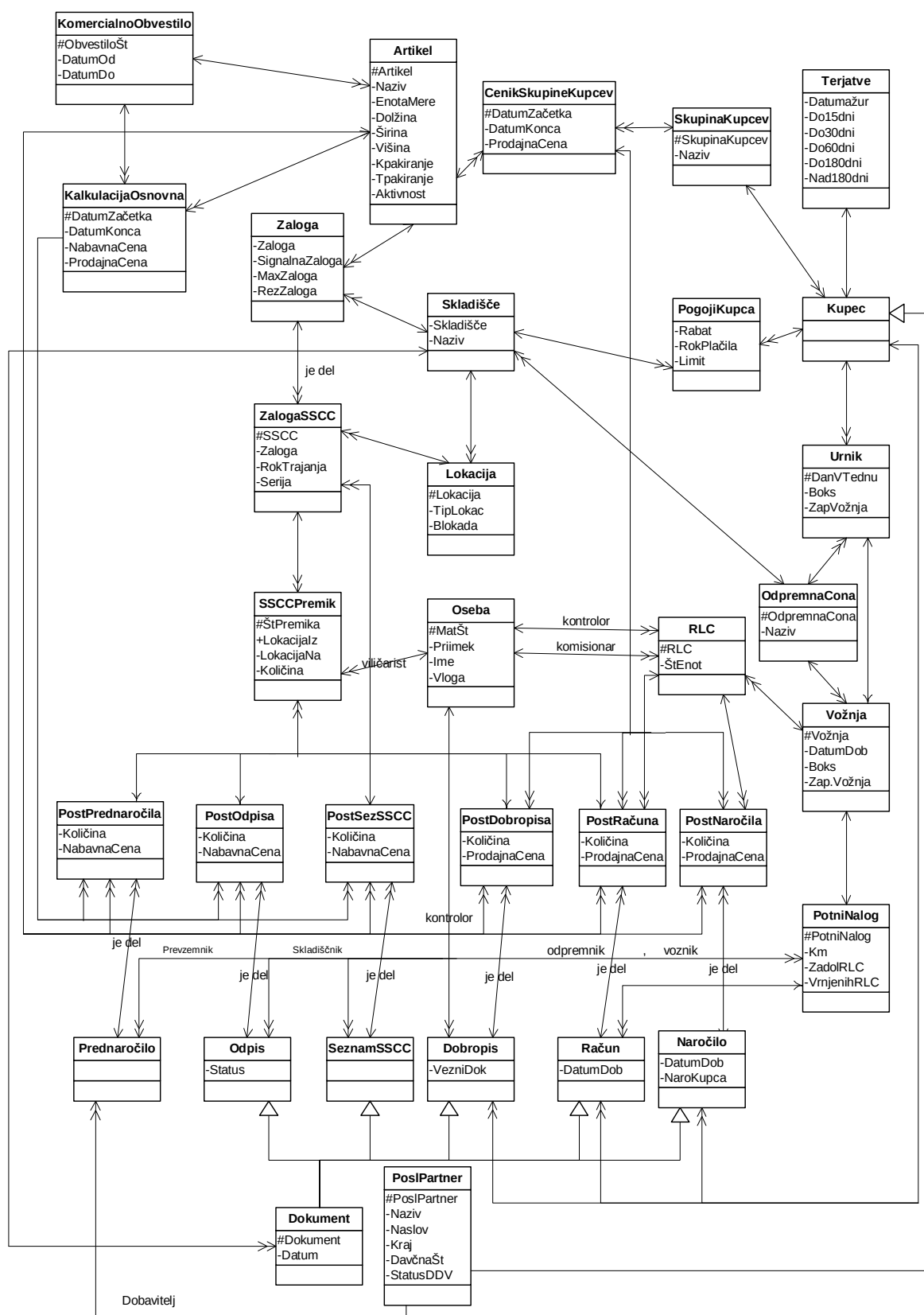
V nadaljevanju je treba poiskati razrede, ki imajo skupne lastnosti, katere lahko izločimo v nadrazrede, ali primere, kjer lahko iz razrede kreiramo specializirane razrede. Ponovno analiziramo tabele nalog in aktivnosti ter začetni objektni model. Po potrebi dodamo razrede, ki smo jih naknadno opredelili.

V obravnavanem primeru lahko definiramo naslednje nadrazrede:

- **Dokument:** #ŠtDokumenta, Datum, ki združuje skupne lastnosti dokumentov Prednaročilo, Odpis, SeznamSSCC, Dobropis, Račun in Naročilo
- **PoslPartner:** #PoslPartner, Naziv, Naslov, Kraj, DavčnaŠt, StatusDDV, ki združuje lastnosti razredov Kupec in Dobavitelj in
- **Oseba:** #MatŠt, Priimek, Ime, Vloga, ki nadomesti razrede Komisionar, Odpremnik, Kontrolor, Prevzemnik, Skladiščnik, Viličarist.

Končni objektni model je prikazan na naslednji strani.

Slika 36: Končni objektni model



6.7. Načrtovanje

Fazo načrtovanja izvedemo v dveh korakih. V prvem z analizo tabel aktivnosti in nalog identificiramo operacije, v drugem pa oblikujemo aplikacijski model.

6.7.1. Definiranje operacij

Pri definiranju operacij podrobno analiziramo tabelo nalog, vsako nalogo posebej. Za beleženje operacij kreiramo tabelo operacij (angleško: operation table).

Z analizo opisa, vhodov, izhodov in pogojev iz tabele nalog identificiramo operacije, ki so potrebne za izvajanje posamezne naloge, in razred, ki jih vsebuje. Za izvajanje ene naloge je lahko potrebna ena ali več operacij. V drugem primeru ena naloga zavzema več vrstic v tabeli operacij.

Ko smo identificirali vse operacije, jih vnesemo v končni objektni model, nato pa podrobneje definiramo algoritme vsake operacije.

V obravnavanem primeru identificirane operacije so zbrane v tabeli operacij, katere del je prikazan na naslednji strani, celotna tabela pa je prikazana v prilogi. Poleg prikazanih operacij imajo vsi razredi tudi običajne operacije za kreiranje, spreminjanje, brisanje in pregledovanje objektov.

6.7.2. Aplikacijski model

Aplikacijski model zgradimo na podlagi analize tabele entitet in tabele aktivnosti z vpisanimi delovnimi in poslovnimi procesi.

Prva dva dela modela sta sestavljena iz poročil in odločitvenih problemov, ki jih potrebujejo oziroma rešujejo različni vodstveni nivoji v podjetju. Strukturo modela lahko razberemo iz tabele entitet, ki opredeljuje potrebna poročila in probleme odločanja.

Tretji del aplikacijskega modela razberemo iz tabele aktivnosti. V aplikacijskem modelu strukturiramo poslovne procese, znotraj njih na nižjem nivoju delovne procese in na najnižjem nivoju aktivnosti. Tako dobimo pregledno strukturno sliko zahtev in odločitvenih problemov ter hierarhično shemo procesov in aktivnosti.

V nadaljevanju je za posamezne dele aplikacijskega modela treba definirati uporabniške vmesnike, ki bodo uporabnikom omogočili interakcijo s sistemom.

V nadaljevanju sta prikazana prvi del tabele operacij ter aplikacijski model. Tabela operacij je v celoti prikazana v prilogi dela.

Tabela 11: Tabela operacij 1 / 4

Naloga	Operacija	Razred
K1,3	KreirajDobavitelja	Dobavitelj
K2,3	SpremeniDobavitelja	Dobavitelj
K5,3	KreirajArtikel	Artikel
K6,3	KreirajKalkulacijo	KalkulacijaOsnovna
K9,3	SpremeniKalkulacijo	KalkulacijaOsnovna
K11,3	DeaktivirajArtikel	Artikel
K12,2	PreglejObvestilo	KomercialnoObvestilo
K12,3	KreirajObvestilo	KomercialnoObvestilo,
K12,3	PreglejArtikel	Artikel
K12,3	PreglejKalkulacijo	KalkulacijaOsnovna
K12,9	PreglejObvestilo	KomercialnoObvestilo
K13,3	PreglejProdajo	Artikel, PostRačuna
K14,3	SpremeniZalogo	Zaloga
K15,3	KreirajPrednaročilo	Prednaročilo, PostPrednaročila
K15,3	PredlagajKoličino	Zaloga
K16,3	SpremeniPrednaročilo	Prednaročilo, PostPrednaročila
K17,3	PotrdiPrednaročilo	Prednaročilo, PostPrednaročila
K17,4	PreglejNovaPrednaročila	Prednaročilo
K18,4	DoločiPrevzemnika	Prednaročilo
K18,5	IzberiPrednaročilo	Prednaročilo
K19,5	PreglejPrednaročilo	Prednaročilo
K20,5	VnesiKoličino	PostPrednaročila
K22,5	VnesiDobavnico	Prednaročilo
K23,5	SpremeniLogPodatke	Artikel
K24,5	KreirajSSCC	SSCCPremik
K24,5	DoločiLokacijo	SSCCPremik, Lokacija, ZalogaSSCC
K25,5	IzpišiSSCC	SSCCPremik
K25,7	IzberiSSCC	SSCCPremik
K26,7	PotrdiPremikSSCC	SSCCPremik, ZalogaSSCC
K28,8	ZaključiPrevzem	Prednaročilo, PostPrednaročila
K28,8	SpremeniZalogo	Zaloga
K31,9	KreirajKupca	Kupec
K32,9	SpremeniKupca	Kupec
K33,9	KreirajPogojeKupca	PogojiKupca, Kupec,
K34,9	KreirajSkupinoKupcev	SkupinaKupcev
K35,9	DoločiSkupinoKupcev	Kupec
K36,9	KreirajCenikSkupineKupcev	CenikSkupineKupcev,
K37,9	SpremeniCenikSkupineKupcev	CenikSkupineKupcev, KalkulacijaOsnovna
K38,10	UgotoviCenoKupca	Artikel, CenikSkupineKupcev

Slika 37: Aplikacijski model

ISL	Reports	Nabava po dobaviteljih	
		Odpiri po skladiščih in artiklih	
		pregled kom. listov po komisionarjih, kontrolorjih	
		pregled nerazporejenih kom. listov	
		Pregled odobrenih popustov v prodaji	
		pregled opravljenega dela po komisionarjih	
		pregled opravljenega dela po prevzemnikih	
		pregled predvidenih prevzemov	
		pregled raporejenih kom. listov po voznjeh	
		Prodaja po kupcih in artiklih	
	Decision Suport	sumarni podatki-prodaja	
		Zaloga po rokih trajanja	
		Zaloga po skladiščih in artiklih	
		nadzor planiranih voženj	
		nadzor statusov za vse odpreme cone	
		Odstopanje dejanskih dostav strankam od	
		Odstopanje signalnih zalog od predlaganih	
		Predlog signalnih zalog in naročilnih količin	
	nabavljanje	pripravljanje dobavitelja	vnos matičnih podatkov dobavitelja spremembe podatkov o dobavitelju
		pripravljanje artiklov	pridobitev ponudbe novih artiklov odobritev novih artiklov vnos matičnih podatkov artikla vnos osnovne kalkulacije
		sprememba kalkulacije	pridobitev cenika dobavitelja odobritev sprememb cen vnos sprememb kalkulacije
		izločanje artikla iz asortimana	pridobitev predloga ukinjanja artikla sprememba statusa aktivnosti artikla
		obveščanje o spremembah artiklov/cen	obvestilo o spremembi asortimana/cen analiza preteklih prodaje
		ocenjevanje nabavnih potreb	nastavljanje signalnih zalog
		obnavljanje zalog	kreiranje predloga - signalne zaloge korekcija naročila potrditev in posredovanje naročila
		prevzemanje blaga od dobavitelja	dodeljevanje prednaročila prevzemniku sprejem dobaviteljevega tovornjaka količinsko in kakovostno preverjanje blaga vračanje praznih palet dobavitelju podpisovanje dokumentov
		priprava blaga za skladiščenje	preverjanje in vnos manjkajočih log. podatkov kreiranje postavk delnih količin označevanje palete za skladiščenje vskladiščenje palete na lokacijo
		kontrola in obdelava podatkov o vходу blaga	Dopolnjevanje dobaviteljeve dokumentacije kompletiranje, vnos in obdelava prevzema pošiljanje vhodne dokumentacije v likvidaturo
	prodajanje	pripravljanje kupca	pridobitev kupca vnos matičnih podatkov kupca sprememba matičnih podatkov kupca
		komercialni pogoji kupca	splošni pogoji kupca vnos skupine kupcev za cenik uvrstitev kupca v skupino za cenik poseben cenik kupca
		pridobivanje naročil kupcev	nadziranje sprememb cen-popravljanje cenikov ponudba kupcu sprejem naročil kupcev
		nadzor izpolnitve naročila	pregled statusa naročila odgovor kupcu o predvideni izdobi
		obdelovanje naročil	selekcija naročil za obdelavo združevanje naročil rezervacija zalog razbijanje naročil računalniško tehtanje blaga izpisovanje liste za disponiranje dokončno razporejanje naročil na vožnje obveščanje kupca
		komisioniranje	izpis kom. listov, izdelava zapisa fakture razporejanje komisionirnih listov med komisionarje komisioniranja-priprava postavke kontrolno tehtanje, zaključek komisioniranja kontrola RLC
		odpremljanje blaga	izbor paketa RLC za odpremo predaja blaga vozniku kontrola voznika za potni nalog izpisovanje računov izpisovanje potnega naloga za vožnjo predajanje potnega naloga, računov vozniku
		zaključevanje vožnje	vračanje prazne embalaže kompletiranje odpreme dokumentacije arhiviranje računov, komisionirnih listov zaključevanje potnega naloga izdaja dobropisa za embalažo kontrola salda transportne embalaže za vožnjo arhiviranje dobropisov za embalažo, potnih nalogov
		vračanje blaga kupcev	vračanje blaga od kupca ločevanje vrnjenega blaga glede na uporabnost dopolnjevanje obrazca za neskladnost izdaja dobropisa kupcu arhiviranje dobropisa, obrazca o neskladnosti označevanje blaga za redno vskladiščenje predaja vrnjenega blaga skladiščniku

V nadaljevanju preidemo na opis faze implementacije.

6.8. Implementacija

V zadnji fazi metodologije TAD izvedemo fizično implementacijo sistema, ki smo ga razvijali v predhodnih fazah.

V fazi implementacije najprej izberemo ustrezno podatkovno bazo in programski jezik, v katerem bomo kodirali algoritme. Algoritme, opredeljene v objektnem in aplikacijskem modelu, kodiramo in prevedemo. V bazo podatkov ustrezno preslikamo opredeljene razrede. Vsak razred iz diagrama razredov se preslika v tabelo relacijske podatkovne baze in vsak atribut razreda se preslika v atribut tabele (Lahajnar, 2001, str. 20).

V obravnavanem primeru potekata vzporedno dve aktivnosti:

- Pripravljalne aktivnosti za projekt prenove informacijskega sistema logistike, bodisi z nakupom rešitve ali lastno izdelavo.
- Dograjevanje obstoječe rešitve za potrebe zagotavljanja sledljivosti.

V okviru dograditve obstoječe rešitve bomo lahko uporabili izdelke, nastale v okviru predhodnih faz metodologije. Predvsem bodo zelo uporabni opisi bodočega procesa, ki je opredeljen ne glede na izbiro variante implementacije rešitve. Pri dograditvi obstoječe rešitve iz objektnega modela lahko razberemo, da sta potrebni dve novi tabeli in sicer za beleženje posameznih postavk zalog po paletah ter za beleženje premikov palet v sistemu. V modelu sta to razreda ZalogaSSCC in SSCCPremik. Iz modela so razvidne tudi povezave z ostalimi razredi, kateri se v obstoječem sistemu realizirajo v obliki tabel. Nadalje bo treba kodirati nekaj univerzalnih podprogramov – modulov za manipuliranje s premiki in zalogami, ki jih bo treba vključiti v glavne programe, ki izvajajo prometne transakcije. Tako bo način dostopa do novih tabel določen na enem mestu. Zaradi uporabe RF terminalov, ki imajo ekran omejen na 16 vrstic s po 40 znaki, bo treba na novo izdelati nekaj uporabniških vmesnikov, za namensko uporabo na tej opremi.

Pred dejansko implementacijo sprememb bo treba še bolj podrobno opredeliti potrebne spremembe obstoječe rešitve in upoštevati njene posebnosti. Projekt prenove se bo izvajal neodvisno od dograjevanja obstoječe rešitve.

7. ZAKLJUČEK

V okviru dela sem prikazal različne metodologije za izgradnjo informacijskih sistemov. Predstavljeni so osnovni koncepti objektnega pristopa in poenoteni jezik za modeliranje UML, ki skupaj s procesom razvoja informacijskih sistemov, kot ga predlagajo avtorji UML, predstavlja najbolj razširjeno metodologijo za objektni način izgradnje informacijskih sistemov.

Podrobno je predstavljena metodologija TAD, ki z uporabo posebnih tabel v fazah izgradnje informacijskega sistema predstavlja drugačno pot od uporabe diagramskih tehnik, ki jih predpisuje UML. Metodologija TAD je nadalje uporabljena na praktičnem primeru prenove obstoječega informacijskega sistema logistike, kjer sem z uporabo predpisanih faz in korakov uspel analizirati obstoječe poslovne procese, opredeliti strukturo ciljev in informacijskih potreb ter v fazi prenove poslovnih procesov spremeniti procese na način, ki omogoča povečanje kakovosti opravljenega dela ob povečani produktivnosti dela. Razvit je tudi objektni model novega informacijskega sistema, iz katerega so razvidni novi razredi, ki jih je treba implementirati v obstoječem informacijskem sistemu. V koraku načrtovanja je poleg opredelitev operacij prikazan pregleden aplikacijski model, ki v skupnem pregledu združuje opredeljene analize in probleme odločanja ter predvideno strukturo aplikacije.

Metodologija TAD pokriva vse faze razvoja informacijskega sistema. Njena praktična uporaba se izkaže izrazito neproblematična zaradi dobro definiranih faz in korakov metodologije. Prehodi med posameznimi fazami so dobro opredeljeni in gladki. Podrobno je opredeljen tudi proces izgradnje objektnega modela. Strnjen prikaz aktivnosti in nalog v tabelah omogoča celovit pogled analitika na obravnavano problematiko, prav tako pa je možno tabele razvijati po delih, za posamezne poslovne procese. Morebitna pomanjkljivost metodologije v primerjavi z UML je v tem, da trenutno poleg orodja, ki omogoča razvoj tabel na enostaven način, ni ustrezne programske podpore metodologije kot jo ima UML. Po zadnjih informacijah je ustrezno orodje v zaključni fazi razvoja.

V primerjavi z UML se metodologija TAD izkaže predvsem v dobri definiciji faz in strnjenem prikazu problematike ter enostavnem razumevanju opredeljenih tabel. Za uporabo UML je potrebno bistveno več predhodnega znanja, tako s strani analitika kot tudi predstavnikov uporabnikov, ki sodelujejo na projektu.

V okviru dela sem potrdil osnovno hipotezo, da je mogoče z uporabo metodologije TAD analizirati obstoječe procese, jih v fazi prenove spremeniti tako, da odgovarjajo novo opredeljenim ciljem ter pripraviti načrt novega informacijskega sistema, ki ga je mogoče uporabiti tudi pri dograditvi obstoječega sistema.

8. LITERATURA IN VIRI

8.1. Literatura:

1. Avison D. E., Fitzgerald G.: Information Systems Development: Methodologies, Techniques and Tools, 2nd ed. London: McGraw-Hill, 1996. 505 str.
2. Booch, Grady: Object-oriented analysis and design with applications. Redwood city: The Benjamin/Cummings, 1994. 589 str.
3. Booch Grady: Object solutions: managing the object-oriented project. Menlo Park: Addison-Wesley, 1996. 323 str.
4. Booch Grady, Rumbaugh James, Jacobson Ivar: The Unified Modeling Language User Guide. Reading: Addison Wesley Longman, Inc. 1999. 482 str.
5. Črv, Milan: Objektno orientirani pristopi s primerom uporabe pri razvoju informacijskega sistema bolnice. Magistrsko delo. Ljubljana: EF, 1997. 109 str.
6. Črv, Milan: Objektni pristop k prenovi poslovnih procesov in izgradnji informacijskega sistema - metodološki vidiki. Doktorska disertacija. Ljubljana: EF, 2000. 202 str.
7. Damij Nadja: Uporaba metodologije TAD in UML za razvoj aplikacije Tranzit. Ljubljana: Uporabna informatika, številka 2, 2001, str. 86-94.
8. Damij Talib: An Object-Oriented Metodology for Information Systems development and Business Process Reengineering. Journal of Object-Oriented Programming. Fort Lee, Vol. 13, No. 4, 2000, str. 23-34.
9. Damij Talib: Tabular Application Development for Information Systems. New York: Springer Verlag, 2001. 256 str.
10. Donaldson Dewitz Sandra: Systems Analysis and Design and Transition to Objects. New York: McGraw-Hill, 1996. 192 str.
11. Fowler Martin, Scott Kendall: UML Distilled, Second Edition. New York: Addison Wesley Longman, 2000. 186 str.
12. Grad Janez, Jaklič Jurij: Baze podatkov. Ljubljana: EF, 1996. 260 str.
13. Hammer Michael, Champy James: Reengineering the Corporation: A Manifesto for Business Revolution. New York: Harper Business, 2001. 257 str.
14. Harmon Paul, Hall Curtis: Intelligent Software Systems Development – An IS Manager's Guide. New York: John Wiley & Sons, 1993. 496 str.
15. Jacobson Ivar: Object-oriented software engineering: a use case driven approach. Workingham: Addison-Wesley, 1996. 528 str.

16. Jacobson Ivar, Booch Grady, Rumbaugh James: The Unified Software Development Process. Reading: Addison Wesley Longman, Inc. 1999. 463 str.
17. Jacobson Ivar, Ericsson Maria, Jacobsson Agneta: The Object Advantage: Business process reengineering with object technology. New York: ACM Press, 1995, 347 str.
18. Katzenbach Jon R., Smith, Douglas K.: The wisdom of teams. New York: Harper Collins Publishers Inc., 1999. 320 str.
19. Kovačič Andrej: Informatizacija poslovanja. Ljubljana: EF, 1998. 214 str.
20. Kovačič Andrej, Vintar Mirko: Načrtovanje in gradnja informacijskih sistemov. Ljubljana: Dzs, 1994. 316 str.
21. Komel Janez: Približevanje relacijskih in objektnih baz. Ljubljana: Dnevi slovenske informatike - Zbornik posvetovanja 1998, str. 19-26.
22. Lahajnar Sebastian: Načrtovanje relacijskih podatkovnih baz z UML. Ljubljana: Uporabna informatika, številka 1, 2001, str. 16-23.
23. Martin James: Information Engineering: Introduction. Upper Saddle River: Prentice Hall, 1989. 178 str.
24. Martin James: Rapid Application Development. New York: Macmillan Publishing Company, 1991. 788 str.
25. Martin James: Enterprise Engineering. Carnforth, England: Savant Institute, 1994. Vol 1-5, 775 str.
26. Murch Richard: Project Management: Best practices for IT professionals. Upper Saddle River: Prentice Hall, 2001. 247 str.
27. Odell James J.: Advanced Object-Oriented Analysis and Design Using UML. Cambridge: Cambridge university press, 1999. 246 str.
28. Robinson Keith, Berrisford Graham: Object-oriented SSADM. New York: Prentice Hall, 1994. 524 str.
29. Roblek Dominik: Objektno relacijsko preslikovanje. Ljubljana: Dnevi slovenske informatike - Zbornik posvetovanja 2001, str. 117-124.
30. Rumbaugh James, Jacobson Ivar, Booch Grady: The Unified Modelling Language Reference Manual. Reading: Addison Wesley Longman, Inc. 1999. 550 str.
31. Sturm Jake: VB6 UML Design and development. Birmingham: Wrox Press, 1999. 550 str.
32. Taylor David A.: Object-Oriented Information Systems: Planning and Implementation. New York: John Wiley & Sons, Inc., 1992. 357 str.

33. Taylor David A.: Business Engineering with Object Technology. New York: John Wiley & Sons, Inc., 1995. 188 str.
34. Tomažič-Spiller Majda: Razvoj informacijskega sistema avtomatskih meteoroloških postaj z uporabo metode 'Tabelarni razvoj aplikacij'. Magistrsko delo. Ljubljana: EF, 2001. 91 str.
35. Watson Gregory H.: Business Systems Engineering. New York: John Wiley & Sons, Inc., 1994. 287 str.
36. Yeates Don, Cadle James: Project Management for Information Systems. London: Pitman Publishing, 1996, 340 str.
37. Živkovič Aleš, Heričko Marjan, Rozman Ivan: Razvoj programske opreme z uporabo jezika UML. Uporabna informatika, številka 4, 2000, str. 200-207.

8.2. Viri:

1. Ean International: Continuous Replenishment - How to use the EAN UCC Standards. B.k.: EAN International, 2000. 40 str.
2. Ean International: EAN-UCC Global User Manual. B.k.: EAN International, 2000. 69 str.
3. Interno gradivo Mercator: Organizacijski predpis logistike OP-15-001. Ljubljana: Mercator, 2000. 7 str.
4. Interno gradivo Mercator: Poslovniki kakovosti QM-4-01-01. Ljubljana: Mercator, 2000. 4 str.

PRILOGA

KAZALO PRILOGE

Tabela 1: Tabela nalog – obstoječe stanje.....	1
Tabela 2: Prenovljena tabela nalog	5
Slika 1: Procesni model 1 / 3.....	9
Slika 2: Procesni model 2 / 3.....	10
Slika 3: Procesni model 3 / 3.....	11
Tabela 3: Tabela operacij	12

Tabela 1: Tabela nalog – obstoječe stanje

Activity	Task Code	Description	Time	Condition	Input/Output
1 - vnos matičnih podatkov dobavitelja	K(1,2)	na osnovi sklenjene pogodbe vodja nabave posreduje podatke o dobavitelju nabavnemu komercialistu		sklenjena pogodba	podatki o dobavitelju
	K(1,3)	nabavni komercialist vnese matične podatke dobavitelja			podatki dobavitelja
2 - spremembe podatkov o dobavitelju	K(2,2)	Dobavitelj sporoči spremembo podatkov, npr. naslova. Vodja nabave posreduje podatke komercialistu			podatki o dobavitelju
	K(2,3)	komercialist vnese spremenjene podatke dobavitelja		dobavitelj v bazi	podatki o dobavitelju
3 - pridobitev ponudbe novih artiklov	K(3,2)	vodja sprejme predlog novih artiklov in ponudbo dobavitelja			predlog novih artiklov
	K(3,3)	nabavni komercialist prejme ponudbo novih artiklov, predlaga izbor in ga posreduje vodji nabave		dobavitelj v bazi	ponudba
4 - odobritev novih artiklov	K(4,2)	vodja nabave odobri nove artikle			odobren predlog
	K(4,3)	nabavni komercialist sprejme odobreni predlog novih artiklov			odobreni predlog
5 - vnos matičnih podatkov artikla	K(5,3)	nabavni komercialist vnese matične podatke artikla		odobren nov artikel	matični podatki artikla
6 - vnos osnovne kalkulacije	K(6,3)	nabavni komercialist vnese osnovno kalkulacijo		artikel v bazi	kalkulacija artikla
7 - pridobitev cenika dobavitelja	K(7,2)	vodja nabave sprejme cenik			cenik dobavitelja
	K(7,3)	dobavitelj najavi spremembo cen, nabavni komercialist najavo posreduje vodji nabave			cenik dobavitelja
8 - odobritev sprememb cen	K(8,2)	Vodja nabave odobri spremembo cen ali pa intervenira pri dobavitelju. Odločitev posreduje nabavnemu komercialistu		ali je napoved pravočasna (1 mesec)?	cenik, predvidena uveljavitev
	K(8,3)	nabavni komercialist sprejme odobreni/zavrjeni cenik dobavitelja			cenik dobavitelja
9 - vnos sprememb kalkulacije	K(9,3)	nabavni komercialist vnese spremenjeno kalkulacijo		odobrena sprememba	kalkulacija artikla
10 - pridobitev predloga ukinjanja artikla	K(10,3)	dobavitelj posreduje napoved ukinitve artikla			napoved ukinjenih artiklov
11 - sprememba statusa aktivnosti artikla	K(11,3)	nabavni komercialist spremeni status artikla		artikel v bazi	status artikla
12 - obvestilo o spremembi asortimana/cen	K(12,2)	vodja nabave sprejme obvestilo			obvestilo o spremembi asortimana/cen
	K(12,3)	nabavni komercialist obvesti vodjo nabave in prodaje o napovedanih spremembah		spremenbe	obvestilo o spremembah asortimana in cen, za prihodnje obdobje, evidenca predvidenih sprememb cen
12 - obvestilo o spremembi asortimana/cen	K(12,9)	vodja prodaje prejme obvestilo			obvestilo o spremembah asortimana/cen
13 - analiza pretekle prodaje	K(13,3)	nabavni komercialist izpiše in preuči gibanje prodaje skupin artiklov, za katere je zadožben			analiza prodaje in predlog nivojev signalnih in maksimalnih zalog
14 - nastavljanje signalnih zalog	K(14,3)	nabavni komercialist na osnovi predloga nastavi in ročno popravi signalne in maksimalne zaloge		predlog drugačen od trenutnih vrednosti	predlog nivojev zalog, signalna, maksimalna zaloga artikla na skladišču
15 - kreiranje predloga - signalne zaloge	K(15,3)	glede na trenutne zaloge nabavni komercialist kreira predlog naročila		zaloga <= signalni	predlog naročila
16 - korekcija naročila	K(16,3)	glede na zunanje dejavnike popravi predlog naročila			naročilo dobavitelju
17 - potrditev in posredovanje naročila	K(17,3)	komercialist pošlje prednaročilo dobavitelju in v skladišče		potrjeno prednaročilo	stiskano prednaročilo v skladišču, naročilo pri dobavitelju
	K(17,4)	vodja prevzema prevzame prednaročila in jih razporedi po datumih		stiskana prednaročila v skladišču	izpisano prednaročilo dobavitelju
18 - dodeljevanje prednaročila prevzemniku	K(18,4)	vodja prevzema na določen datum preda prednaročila prevzemniku		datum=datum predvidenega prevzema	
	K(18,5)	prevzemnik prevzame prednaročilo na podlagi katerega opravi prevzem blaga			prednaročilo dobavitelju
19 - sprejem dobaviteljevega tovornjaka	K(19,5)	prevzemnik sprejme dobaviteljev tovornjak, za katerega ima razpisano prednaročilo dobavitelju.		prednaročilo za ta prevzem	prednaročilo dobavitelju
20 - količinsko in kakovostno preverjanje blaga	K(20,5)	prešteje količino, preveri deklaracije, rok uporabnosti, fizične lastnosti blaga		blago na prevzemni rampi	prevzeta Q vpisana na prednaročilu
21 - vračanje praznih palet dobavitelju	K(21,5)	prevzemnik vrne enako število praznih palet kot je prevzel polnih			
22 - podpisovanje dokumentov	K(22,5)	prevzemnik podpiše dobaviteljevo dobavnico,		vse blago je prevzeto	dobavnica dobavitelja
23 - preverjanje in vnos manjkajčih log. podatkov	K(23,5)	Prevzemnik preveri log. podatke in vnese popravke oz. manjkajoče podatke		ali podatki manjkajo?	podatki vnešeni v IS
24 - označevanje palete za skladiščenje	K(24,5)	Prevzemnik na paletni označevalni listek napiše datum, kom. lokacijo		blago prevzeto	označevalni listek napisan
	K(24,7)	viličarist prevzame označevalni listek in blago			označevalni listek na paleti
25 - vskladiščenje palete na lokacijo	K(25,7)	Viličarist paleto odpelje na komisionirno lokacijo.		prazna kom. lok. če polna: rezervna lok+karton	evidenčni kartonček: rezervna lokacija, količina, rok

Activity	Task Code	Description	Time	Condition	Input/Output
26 - evidentiranje podatkov o blagu - skladiščniku	K(26,5)	prevzemnik vpiše matične št. delavcev na prednaročilo in ga podpiše ter preda skladiščniku. Ko skladiščnik podpiše, 1.izvod z dobaviteljevo dokumentacijo preda v prevzemno pisarno			podpisano prednaročilo (prevzemnik)
	K(26,6)	skladiščnik podpiše prednaročilo in 1. izvod vrne prevzemniku		blago v redu skladiščeno	izpolnjeno prednaročilo
	K(26,8)	administrator zbere prednaročilo in dobaviteljeve dokumente		prednaročilo podpisano(skladiščnik)	prednaročilo, dobavnica, CMR,
27 - kompletiranje, vnos in obdelava prevzema	K(27,8)	administrator pregleda podpise, kompletira do. dobavnico, cmr, vnese popravke količin, izpiše kalkulacijo in zaključi obdelavo prevzema		vsi dokumenti prevzema so zbrani	prevzem potrjen, zaloga povečana
28 - pošiljanje vhodne dokumentacije v likvidaturo	K(28,8)	Administrator vhodno dokumentacijo pošlje v likvidaturo		pregledana dokumentacija	dob. dobavnica, prevoznica/CMR, prednaročilo in prevzemna kalkulacija v likvidaturi
29 - pridobitev kupca	K(29,9)	vodja prodaje pridobi kupca, preveri njegovo boniteto in registracijo		boniteta in registracija v redu	podatki o kupcu
30 - vnos matičnih podatkov kupca	K(30,9)	vodja prodaje vnese matične podatke kupca		kupec sprejet	podatki o kupcu
31 - sprememba matičnih podatkov kupca	K(31,9)	kupec posreduje spremembe podatkov v pisni obliki, vodja prodaje spremembe vnese v računalnik		kupec je v bazi	sprememba matičnih podatkov kupca
32 - splošni pogoji kupca	K(32,9)	vodja prodaje dogovori splošne pogoje (rok plačila, rabat) in jih vnese v bazo		kupec v bazi	splošni pogoji kupca
33 - vnos skupine kupcev za cenik	K(33,9)	Vodja prodaje vnese skupino kupcev, v katero se združuje kupce, ki dobijo isti cenik		skupina ne obstaja	skupina za cenik
34 - uvrstitev kupca v skupino za cenik	K(34,9)	vodja prodaje uvrsti kupce v skupino za isti cenik		skupina, kupec obstaja	skupina za cenik prirejena kupcu
35 - poseben cenik kupca	K(35,9)	Vodja prodaje po dogovoru s kupcem izdela poseben cenik izbranih artiklov za kupca / skupino kupcev. Cene vnese v bazo.		kupec, artikli, skupina kupcev obstajajo	cenik za skupino kupcev, kupec uvrščen v skupino
	K(35,10)	cenik za kupca dostopen v bazi			cenik za kupca
36 - nadziranje sprememb cen-popravljanje cenikov	K(36,9)	vodja prodaje spremlja evidenco predvidenih sprememb cen in po potrebi v dogovoru s kupcem popravlja cenike za kupce, ko je marža prenizka zaradi povečanja vhodnih cen		marža < zelene	evidenca sprememb cen, cenik skupine kupcev
37 - ponudba kupcu	K(37,10)	prodajni referent izpiše in posreduje ponudbo kupcu		artikli,kupec,cenik obstajajo	ponudba kupcu
38 - sprejem naročil kupcev	K(38,10)	naročilo se prepíše iz kupčevega IS ali vnese s strani referenta		priselo naročilo	naročilo vnešeno
	K(38,11)	naročilo v sistemu			naročilo v sistemu
39 - pregled statusa naročila	K(39,10)	za izbrano naročilo v IS pregleda status, v katerem je naročilo - čaka, razporejeno, v komisioniranju, v kontroli, v odpremi, odpremljeno		naročilo v sistemu	podatki o statusu naročila in računa
40 - odgovor kupcu o predvideni izdobi	K(40,10)	kupec poizveduje o statusu naročila, referent preveri in mu odgovori			informacija o predvideni dobavi
41 - selekcija naročil za obdelavo	K(41,11)	preverjanje kupcev, če so aktivni in označevanje izbranih naročil		boniteta v redu, limit odobren, kupec aktiven	naročila izbranih kupcev izbrana
42 - združevanje naročil	K(42,11)	v primeru več naročil istega kupca, se preveri, če kupec dovoli združevanje. Naročila združimo po vrsti naročila		dovoljeno združevanje?	naročila iste vrste istega kupca združena
43 - rezervacija zalog	K(43,11)	preverjanje ali je v skladišču dovolj zaloge, rezerviranje zaloge		zaloge dovolj?	zaloga rezervirana
44 - razbijanje naročil	K(44,11)	izločevanje polnih palet in ustvarjanje novega naročila za polne palete		zahtevano izločanje polnih palet? količina?	novo naročilo za polne palete kreirano
45 - računalniško tehtanje blaga	K(45,11)	izračunajo se volumni in potrebno št. RLC za vsako naročilo napiše se pravo zaporedje za komisioniranje na RLC		znani volumni artiklov, naročila dokončna	volumensko ovrednoteno naročilo-znano število RLC
46 - izpisovanje liste za disponiranje	K(46,11)	Določimo odpremi box za vsako naročilo in določimo datum ter čas odpreme		znano število RLC, urniki	naročila so razporejena glede na urnike, izpisana lista
47 - dokončno razporejanje naročil na vožnje	K(47,11)	Disponent razdeli še nerazporejena naročila na vožnje. Razporedi jih tako, da se napolnijo tovornjaki, ki potujejo v isto smer, z naročili v tisto smer		ni stalnega urnika	vsa naročila razporejena na odpreme cone/vožnje
48 - obveščanje kupca	K(48,11)	potrdilo o predvideni dobavi se pošlje v kupčev IS		kupec ima predvideno avtomatsko potrjevanje	potrditev naročila
49 - izpis kom. listov, izdelava zapisa fakture	K(49,11)	Disponent kreira komisionirni list za vsak RLC in vsako paleto, rezervacija zaloge se zmanjša, zaloga se zmanjša, kreira se zapis računa		razporejeno naročilo	komisionirni listi za RLC in palete v print vrsti, zapis računa kreiran
	K(49,12)	izpiše komisionirne liste			komisionirni listi na papirju
50 - razporejanje komisionirnih listov med komisionarje	K(50,12)	Vodja komisionarjev razporedi kom. liste med komisionarje in položi kom. liste v predalčke		list še ni razporejen?	razporejeni kom. listi med komisionarje v računalniku (skenirano)
	K(50,13)	Komisionar prevzame kom. list, ko zaključi predhodni kom. list		skenirani kom. listi v računalniku	kom. list pri komisionarju, status 1

Activity	Task Code	Description	Time	Condition	Input/Output
51 - komisioniranje-priprava postavke	K(51,7)	komisionar pokliče viličarista		zaloga na komisionirni lokaciji ne zadošča	
	K(51,13)	blago je na kom. lokaciji. Komisionar vzame blago in na kom. list označi prevzeto količino. Če ni blaga na kom. lokaciji, in je na rezervi, pokliče skladiščnika, če ni rezerve popravi količino na kom. listu		zaloga na lokaciji zadošča?	podatki o izdani količini napisani na kom. listu
52 - zaključek komisioniranja	K(52,13)	podpisovanje kom. lista, predaja v pisarno		vse postavke pripravljene	podpisan kom.list
	K(52,15)	sprejem kom. lista			podpisan kom. list
53 - vnos popravkov kom.lista	K(53,15)	vnos popravkov, potrjen zaključek komisioniranja		kom. list še ni potrjen?	potrjen zaključek komisioniranja status 2
54 - predajanje kom. lista kontrolorju	K(54,8)	Administrator posreduje kom. list kontrolorju			koda administratorja v računalniku
	K(54,14)	kontrolor prevzame kom. list in posname svojo kodo		list je potrjen z zaključkom komisioniranja	koda kontrolorja v računalniku
	K(54,15)	administrator odloži obdelan kom. list na kup za kontrolo			komisionirni list - obdelan
55 - kontrola postavke, podpis. kom. lista	K(55,14)	Kontrolor preveri posamezne postavke na RLC dodaja ali odvzema artikle na RLC podpisuje kom. list in ga vrne v pisarno		vse postavke preverjene	podpisan komisionirni list
	K(55,15)	vrnjen kontroliran kom. list			kontroliran kom. list
56 - vnašanje popravljene Q v računalnik	K(56,15)	vnos popravkov, ki jih odkrije kontrolor potrditev kontrole		kontrola opravljena	označen status kom. lista (3-kontrolirano)
57 - izpisovanje nakladalne liste	K(57,15)	izpiše nakladalno listo		vsi rlc za vožnjo kontrolirani stat 3	izpisana nakladalna lista
58 - posredovanje nakladalne liste odpremniku	K(58,15)	administrator preda nakladalno listo		voznik čaka na nakladanje	nakladalna lista
	K(58,16)	odpremnik prevzame nakladalno listo			nakladalna lista
59 - predaja blaga vozniku	K(59,16)	izbira ustreznih RLC, predaja blaga vozniku			nakladalna lista
60 - vpisovanje RLC in palet na nak. listo, podpis	K(60,16)	odpremnik vpiše podatke o številu RLC ter palet			podatki vpisani na nakladalno listo
61 - vračanje nakladalne liste v odpremno pisarno	K(61,15)	odpremnik vrne nakladalno listo v pisarno		blago naloženo, lista izpolnjena	nakladalna lista
	K(61,16)	sprejem izpolnjene nakladalne liste			nakladalna lista
62 - vnašanje št. RLC in palet v računalnik	K(62,15)	administrator vnese število rlc in palet			vnešeni podatki za potni nalog
63 - izpisovanje računov	K(63,15)	izpisovanje računa		vsi RLC računa kontrolirani	izpisan račun, RLC status 4
64 - izpisovanje potnega naloga za eno nakladalno listo	K(64,15)	izpis potnega naloga		računi izpisani, vnešeni RLC, palete, voznik	potni nalog izpisan, status računa T
65 - predajanje potnega naloga, računov vozniku	K(65,15)	predajanje potnega naloga, računov vozniku			potni nalog, računi
66 - vračanje prazne embalaže	K(66,17)	prevzemnik embalaže prevzame prazno transportno embalažo in vračljivo embalažo za blago. Količine transportne embalaže vpiše na potni nalog, količine embalaže za blago primerja s količinami na sprejemnici embalaže, ki jo je oddal voznik.		voznik vrne prazno embalažo in dokumente	potni nalog, sprejemnica embalaže
	K(67,15)	administrator sprejme račun			račun
67 - kompletiranje odpremne dokumentacije	K(67,17)	prevzemnik embalaže potni nalog in sprejemnice embalaže odda embalažnemu referentu, podpisane račune pa v odpremno pisarno		embalaža prevzeta	potni nalog, sprejemnica embalaže, račun
	K(67,18)	referent za embalažo kompletira sprejemnice embalaže po strankah			sprejemnica embalaže
68 - arhiviranje računov, komisionirnih listov	K(68,15)	administrator posreduje račune skupaj s komisionirnimi listi v arhiv			račun, komisionirni list
69 - zaključevanje potnega naloga	K(69,18)	embalažni referent vnese prevožene kilometre in čas ter vrnjeno transportno embalažo v potni nalog.			potni nalog
70 - izdaja dobropisa za embalažo	K(70,18)	referent za embalažo izda dobropis kupcu za vrnjeno transportno (RLC, palete) in vračljivo embalažo blaga (zaboji, steklenice)		kupec je vrnil embalažo	dobropis
71 - kontrola salda transportne embalaže za vožnjo	K(71,18)	referent za embalažo izpiše saldo transportne embalaže za vožnjo, če ni (prevzeta + vrnjena od stranke - oddana stranki - vrnjena prevzemniku embalaže) = 0, terja voznika		usklajen saldo	potni nalog, dobropisi embalaže
72 - arhiviranje dobropisov za embalažo, potnih nalogov	K(72,18)	referent za embalažo posreduje dobropise in potne naloge v arhiv			dobropis za embalažo, potni nalog
73 - vračanje blaga od kupca	K(73,14)	kontrolor sprejme vrnjeno blago, obrazec o neskladnosti od voznika ali kupca		blago kupljeno pri nas	obrazec o neskladnosti, podatki o računu, postavki, ko je bilo blago prodano
74 - ločevanje vrnjenega blaga glede na uporabnost	K(74,14)	glede na rok uporabe in fizični izgled blaga loči blago na uporabno in neuporabno			rok uporabe

Activity	Task Code	Description	Time	Condition	Input/Output
75 - dopolnjevanje obrazca za neskladnost	K(75,14)	kontrolor dopolni obrazec s podatkom o uporabnosti blaga. Obrazec podpiše in preda referentu za reklamacije			obrazec o neskladnosti
	K(75,18)	referent za reklamacije sprejme obrazec o neskladnosti			obrazec o neskladnosti
76 - izdaja dobropisa kupcu	K(76,18)	referent za reklamacije izda dobropis kupcu		podpisan obrazec o neskladnosti	dobropis, obrazec o neskladnosti
77 - arhiviranje dobropisa, obrazca o neskladnosti	K(77,18)	referent za embalažo posreduje dobropis v arhiv			dobropis
78 - označevanje blaga za redno vskladiščenje	K(78,14)	kontrolor kompletira paleta istega blaga in jo označi s paletnim listkom. Napiše rok trajanja		blago ustrežno	paletni listek
79 - predaja vrnjenega blaga skladiščniku	K(79,6)	skladiščnik prejme vrnjeno blago od kontrolorja			blago, paletni listek
	K(79,14)	kontrolor preda blago skladiščniku			paletni listek
80 - določanje lokacije in vskladiščenje	K(80,6)	skladiščnik določi lokacijo in jo napiše na paletni listek. Paleta preda viličaristu			paletni listek
	K(80,7)	viličarist prejme vrnjeno blago od skladiščnika z navodili za vskladiščenje			paletni listek z lokacijo
81 - označevanje neuporabnega blaga	K(81,14)	kontrolor kompletira paleta neuporabnega blaga. Označi rok trajanja, lokacijo za neuporabno blago in razlog neustreznosti		blago neustrezno	paletni listek
82 - dopolnjevanje komisiorne lokacije	K(82,6)	skladiščnik identificira prazno komisiorno lokacijo, pokliče viličarista		komisiorna lokacija je prazna	kartonček na lokaciji
	K(82,7)	premik palete iz rezervne na komisiorno lokacijo, popravek kartončka evidence rezervnih lokacij		komisiorna lokacija je prazna	kartonček na lokaciji
83 - identifikacija neustreznega blaga	K(83,6)	skladiščnik glede na rok trajanja ali poškodbe blaga identificira neustrezno blago		rok blizu izteka	paletni listek
84 - označevanje neustreznega blaga	K(84,6)	skladiščnik označi blago z novim paletnim listkom, lokacija v posebnem delu skladišča			paletni listek
85 - predaja neustreznega blaga	K(85,6)	skladiščnik preda blago viličaristu, popravi evidenčni kartonček lokacije		lokacija na paletnem listu	kartonček na lokaciji
	K(85,7)	viličarist odpelje neustrezno blago			paletni listek
86 - vskladiščenje neuporabnega blaga	K(86,7)	viličarist začasno vskladišči neuporabno blago			paletni listek
	K(86,14)	kontrolor preda neustrezno blago viličaristu, izda navodilo za začasno vskladiščenje v posebnem delu skladišča			paletni listek
87 - obveščanje o neustreznem blagu	K(87,1)	sprejme informacijo			obrazec o neskladnosti
	K(87,2)	sprejme informacijo			obrazec o neskladnosti
88 - dogovor o vračilu blaga dobavitelju	K(87,6)	skladiščnik poroča o vrstah in količini neustreznega blaga na zalogi		zaloga neustreznega blaga	obrazec o neskladnosti
	K(88,1)	vodja nabave obvesti vodjo sdc o dogovoru z dobaviteljem			seznam artiklov, ki jih dobavitelj ne bo prevzel
89 - odločitev o odpisu blaga	K(88,2)	vodja nabave se dogovori o možnosti vračila blaga dobavitelju. V primeru dogovora se blago vrne, sicer predlaga odpis blaga			seznam artiklov, ki naj jih dobavitelj prevzame
	K(88,3)	vodja nabave obvesti nabavnega komercialista o dogovoru za vračilo blaga			seznam artiklov in količin za vračanje
90 - uničenje blaga	K(89,1)	vodja sdc sprejme odločitev o odpisu blaga			seznam artiklov za odpis
	K(89,6)	skladiščnik sprejme odobritev odpisa			seznam artiklov za odpis
91 - vnos odpisa blaga	K(90,6)	skladiščnik z delavci komisijko uniči blago. Podpisano poročilo o odpisu posreduje administratorju		podpisan seznam vodje sdc	seznam artiklov za odpis
	K(90,8)	administrator prejme izpolnjen zapisnik			seznam uničenih artiklov
92 - posredovanje dokumentacije o odpisu v likvidaturo	K(91,8)	administrator na podlagi komisijke podpisanega zapisnika o uničenju vnese in izpiše odpis blaga			odpis blaga, artikli, količine
93 - vnos predloga vračila blaga dobavitelju	K(92,8)	izpisan odpis skupaj z zapisnikom posreduje v likvidaturo		izpisan odpis	odpis, zapisnik o uničenju
94 - potrditev in izpis predloga vračila blaga	K(93,3)	nabavni komercialist vnese artikle, količine in dogovorjene cene za vrnjeno blago			predlog vračila blaga dobavitelju
	K(94,3)	nabavni komercialist potrdi in izpiše predlog vračila blaga			izpisan predlog vračila
95 - izdaja blaga iz skladišča na prevzemno rampo	K(94,6)	skladiščnik pripravi blago za izdajo			predlog vračila blaga dobavitelju
	K(94,7)	viličarist odpelje blago na prevzemno rampo			paletni listek
96 - kompletiranje blaga za vračilo dobavitelju	K(95,6)	skladiščnik označi palete za vračanje, naroči viličaristu, da odpelje blago na prevzemno rampo, popravi kartonček lokacije		dobaviteljev voznik na rampi	evidenčni kartonček lokacije, predlog vračila, paletni listek
	K(95,7)	viličarist odpelje blago na prevzemno rampo			paletni listek
97 - predaja blaga dobavitelju	K(96,6)	skladiščnik preveri količine za vračanje in po potrebi popravi količino na predlogu vračila blaga dobavitelju		količina je na voljo?	predlog vračila blaga dobavitelju
	K(96,7)	viličarist pripelje na rampo vse blago za vračilo enemu dobavitelju			paletni listki, označeni za vračilo
98 - posredovanje predloga vračila v prevzemno pisarno	K(97,6)	skladiščnik preda blago dobavitelju, podpišeta predlog vračila		blago kompletirano	predlog vračila
	K(98,6)	skladiščnik posreduje izpolnjen predlog vračila			izpolnjen predlog vračila
99 - vnos popravkov na predlog	K(98,8)	administrator v prevzemu prejme izpolnjen predlog vračila		podpisan dokument	izpolnjen predlog vračila
	K(99,8)	administrator vnese popravke		popravki obstojajo	predlog vračila dobavitelju
100 - potrditev in izpis vračila blaga dobavitelju	K(100,8)	administrator potrdi in izpiše vračilo dobavitelju		popravki vnešeni	vračilo dobavitelju
101 - posredovanje vračila blaga v likvidaturo	K(101,8)	administrator posreduje vračilo in podpisan predlog vračila v likvidaturo		izpisano vračilo	vračilo dobavitelju

Tabela 2: Prenovljena tabela nalog

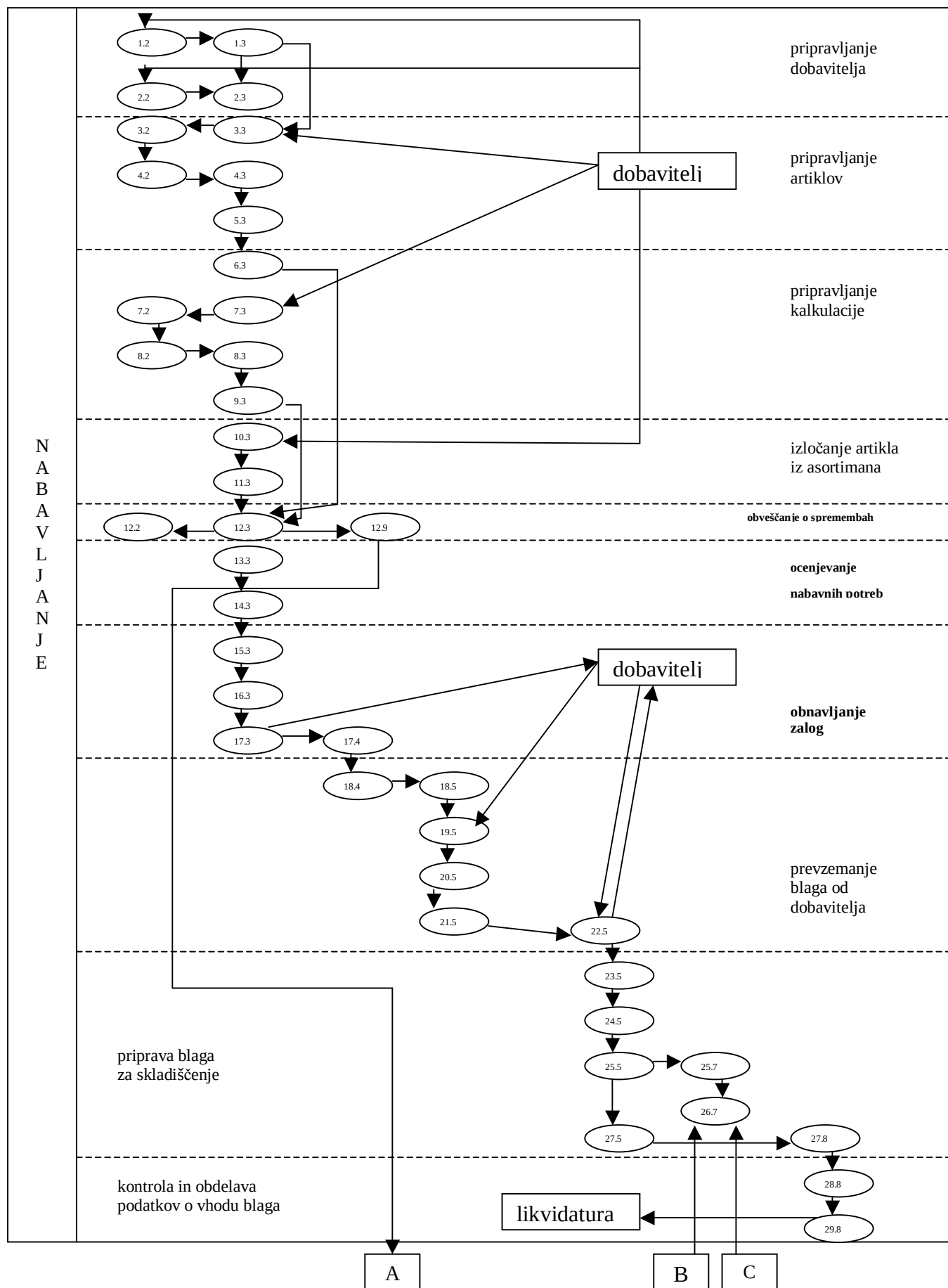
Activity	Task Code	Description	Time	Condition	Input/Output
1 - vnos matičnih podatkov dobavitelja	K(1,2)	na osnovi sklenjene pogodbe vodja nabave posreduje podatke o dobavitelju nabavnemu komercialistu		sklenjena pogodba	podatki o dobavitelju
	K(1,3)	nabavni komercialist vnese matične podatke dobavitelja			podatki dobavitelja
2 - spremembe podatkov o dobavitelju	K(2,2)	Dobavitelj sporoči spremembo podatkov, npr. naslova. Vodja nabave posreduje podatke komercialistu			podatki o dobavitelju
	K(2,3)	komercialist vnese spremenjene podatke dobavitelja		dobavitelj v bazi	podatki o dobavitelju
3 - pridobitev ponudbe novih artiklov	K(3,2)	vodja sprejme predlog novih artiklov in ponudbo dobavitelja			predlog novih artiklov
	K(3,3)	nabavni komercialist prejme ponudbo novih artiklov, predlaga izbor in ga posreduje vodji nabave		dobavitelj v bazi	ponudba
4 - odobritev novih artiklov	K(4,2)	vodja nabave odobri nove artikle			odobren predlog
	K(4,3)	nabavni komercialist sprejme odobreni predlog novih artiklov			odobreni predlog
5 - vnos matičnih podatkov artikla	K(5,3)	nabavni komercialist vnese matične podatke artikla		odobren nov artikel	matični podatki artikla
6 - vnos osnovne kalkulacije	K(6,3)	nabavni komercialist vnese osnovno kalkulacijo		artikel v bazi	kalkulacija artikla
7 - pridobitev cenika dobavitelja	K(7,2)	vodja nabave sprejme cenik			cenik dobavitelja
	K(7,3)	dobavitelj najavi spremembo cen, nabavni komercialist najavo posreduje vodji nabave			cenik dobavitelja
8 - odobritev sprememb cen	K(8,2)	Vodja nabave odobri spremembo cen ali pa intervenira pri dobavitelju. Odločitev posreduje nabavnemu komercialistu		ali je napoved pravočasna (1 mesec)?	cenik, predvidena uveljavitev
	K(8,3)	nabavni komercialist sprejme odobreni/zavrjeni cenik dobavitelja			cenik dobavitelja
9 - vnos sprememb kalkulacije	K(9,3)	nabavni komercialist vnese spremenjeno kalkulacijo		odobrena sprememba	kalkulacija artikla
10 - pridobitev predloga ukinjanja artikla	K(10,3)	dobavitelj posreduje napoved ukinjitve artikla			napoved ukinjenih artiklov
11 - sprememba statusa aktivnosti artikla	K(11,3)	nabavni komercialist spremeni status artikla		artikel v bazi	status artikla
12 - obvestilo o spremembi asortimana/cen	K(12,2)	vodja nabave sprejme obvestilo			obvestilo o spremembi asortimana/cen
	K(12,3)	nabavni komercialist obvesti vodjo nabave in prodaje o napovedanih spremembah		spremembe	obvestilo o spremembah asortimana in cen, za prihodnje obdobje, evidenca predvidenih sprememb cen
	K(12,9)	vodja prodaje prejme obvestilo			obvestilo o spremembah asortimana/cen
13 - analiza pretekle prodaje	K(13,3)	nabavni komercialist izpiše in preuči gibanje prodaje skupin artiklov, za katere je zadožben			analiza prodaje in predlog nivojev signalnih in maksimalnih zalog
14 - nastavljanje signalnih zalog	K(14,3)	nabavni komercialist na osnovi predloga nastavi in ročno popravi signalne in maksimalne zaloge		predlog drugačen od trenutnih vrednosti	predlog nivojev zalog, signalna, maksimalna zaloga artikla na skladišču
15 - kreiranje predloga - signalne zaloge	K(15,3)	glede na trenutne zaloge nabavni komercialist kreira predlog naročila		zaloga <= signalni	predlog naročila
16 - korekcija naročila	K(16,3)	glede na zunanje dejavnike popravi predlog naročila			naročilo dobavitelju
17 - potrditev in posredovanje naročila	K(17,3)	komercialist pošlje prednaročilo dobavitelju in v skladišče		potrjeno prednaročilo	potrjeno prednaročilo v bazi, naročilo pri dobavitelju
	K(17,4)	vodja prevzema pregleduje evidenco izdanih prednaročil			izpisano prednaročilo dobavitelju, prednaročilo v bazi
18 - dodeljevanje prednaročila prevzemniku	K(18,4)	vodja prevzema na določen datum dodeli prednaročilo prevzemnikom		datum=datum predvidenega prevzema	vnos prevzemnika na prednaročilo
	K(18,5)	prevzemnik z RF terminalom izbere prednaročilo prispelega dobavitelja			prednaročilo dobavitelju
19 - sprejem dobaviteljevega tovornjaka	K(19,5)	prevzemnik sprejme dobaviteljev tovornjak, za katerega ima razpisano prednaročilo dobavitelju.		prednaročilo za ta prevzem	prednaročilo dobavitelju
20 - količinsko in kakovostno preverjanje blaga	K(20,5)	prešteje količino, preveri deklaracije, rok uporabnosti, fizične lastnosti blaga		blago na prevzemni rampi, postavka obstaja	skupna prevzeta količina vnešena, postavka potrjena
21 - vračanje praznih palet dobavitelju	K(21,5)	prevzemnik vrne enako število praznih palet kot je prevzel polnih			
22 - podpisovanje dokumentov	K(22,5)	prevzemnik podpiše dobaviteljevo dobavnico,		vse blago je prevzeto	dobavnica dobavitelja
23 - preverjanje in vnos manjkajočih log. podatkov	K(23,5)	Prevzemnik preveri log. podatke in vnese popravke oz. manjkajoče podatke z RF		ali podatki manjkajo?	podatki vnešeni v IS
24 - kreiranje postavk delnih količin	K(24,5)	za vsako postavko prevzema kreiramo delne postavke za vsako paletu, dodelimo SSCC kode za palete		skupna količina = vsota delnih	delne postavke: SSCC, količina, rok trajanja, serija
25 - označevanje palete za skladiščenje	K(25,5)	prevzemnik izpiše označevalne SSCC nalepke in jih nalepi na palete		delne postavke kreirane	SSCC nalepke
	K(25,7)	viličarist skenira SSCC, odčita lokacijo, ki se določi glede na prostost komisionirne lokacije in rok trajanja			SSCC nalepka

Activity	Task Code	Description	Time	Condition	Input/Output
26 - vskladiščenje palete na lokacijo	K(26,7)	Viličarist paleto odpelje na določeno lokacijo. Skenira kodo lokacije, s tem je potrdil vskladiščenje			koda lokacije, povečana zaloga lokacije, potrjeno vskladiščenje
27 - Dopolnjevanje dobaviteljeve dokumentacije	K(27,5)	prevzemnik vpiše številko prednaročila na dobaviteljeve dokumente in jih preda v pisarno. Šifri prevzemnika in viličarista sta implicitni ob prijavi RF terminala			dobaviteljevi dokumenti
	K(27,8)	administrator zbere dobaviteljeve dokumente istega prednaročila		dokumenti označeni s št. prednaročila	prednaročilo, dobavnica, CMR
28 - kompletiranje, vnos in obdelava prevzema	K(28,8)	administrator pregleda podpise, kompletira do. dobavnico, cmr, izpiše kalkulacijo in zaključi obdelavo prevzema		vsi dokumenti prevzema so zbrani	prevzem potrjen, skupna zaloga povečana
29 - pošiljanje vhodne dokumentacije v likvidaturo	K(29,8)	Administrator vhodno dokumentacijo pošlje v likvidaturo		pregledana dokumentacija	dob. dobavnica, prevoznica/CMR, prednaročilo in prevzemna kalkulacija v likvidaturi
30 - pridobitev kupca	K(30,9)	vodja prodaje pridobi kupca, preveri njegovo boniteto in registracijo		boniteta in registracija v redu	podatki o kupcu
31 - vnos matičnih podatkov kupca	K(31,9)	vodja prodaje vnese matične podatke kupca		kupec sprejet	podatki o kupcu
32 - sprememba matičnih podatkov kupca	K(32,9)	kupec posreduje spremembe podatkov v pisni obliki, vodja prodaje spremembe vnese v računalnik		kupec je v bazi	sprememba matičnih podatkov kupca
33 - splošni pogoji kupca	K(33,9)	vodja prodaje dogovori splošne pogoje (rok plačila, rabat) in jih vnese v bazo		kupec v bazi	splošni pogoji kupca
34 - vnos skupine kupcev za cenik	K(34,9)	Vodja prodaje vnese skupino kupcev, v katero se združuje kupce, ki dobijo isti cenik		skupina ne obstaja	skupina za cenik
35 - uvrstitev kupca v skupino za cenik	K(35,9)	vodja prodaje uvrsti kupce v skupino za isti cenik		skupina, kupec obstaja	skupina za cenik prirejena kupcu
36 - poseben cenik kupca	K(36,9)	Vodja prodaje po dogovoru s kupcem izdela poseben cenik izbranih artiklov za kupca / skupino kupcev. Cene vnese v bazo.		kupec, artikli, skupina kupcev obstajajo	cenik za skupino kupcev, kupec uvrščen v skupino
	K(36,10)	cenik za kupca dostopen v bazi			cenik za kupca
37 - nadziranje sprememb cen-popravljanje cenikov	K(37,9)	vodja prodaje spremlja evidenco predvidenih sprememb cen in po potrebi v dogovoru s kupcem popravlja cenike za kupce, ko je marža prenizka zaradi povečanja vhodnih cen		marža < želene	evidenca sprememb cen, cenik skupine kupcev
38 - ponudba kupcu	K(38,10)	prodajni referent izpiše in posreduje ponudbo kupcu		artikli, kupec, cenik obstajajo	ponudba kupcu
39 - sprejem naročil kupcev	K(39,10)	naročilo se prepíše iz kupčevega IS ali vnese s strani referenta		prispelo naročilo	naročilo vnešeno
	K(39,11)	naročilo v sistemu			naročilo v sistemu
40 - pregled statusa naročila	K(40,10)	za izbrano naročilo v IS pregleda status, v katerem je naročilo - čaka, razporejeno, v komisijoniranju, v kontroli, v odpremi, odpremljeno		naročilo v sistemu	podatki o statusu naročila in računa
41 - odgovor kupcu o predvideni izdobi	K(41,10)	kupec poizveduje o statusu naročila, referent preveri in mu odgovori			informacija o predvideni dobavi
42 - selekcija naročil za obdelavo	K(42,11)	preverjanje kupcev, če so aktivni in označevanje izbranih naročil		boniteta v redu, limit odobren, kupec aktiven	naročila izbranih kupcev izbrana
43 - združevanje naročil	K(43,11)	v primeru več naročil istega kupca, se preveri, če kupec dovoli združevanje. Naročila združimo po vrsti naročila		dovoljeno združevanje?	naročila iste vrste istega kupca združena
44 - rezervacija zalog	K(44,11)	preverjanje, ali je v skladišču dovolj zaloge. rezerviranje zaloge		zaloge dovolj?	zaloga rezervirana
45 - razbijanje naročil	K(45,11)	izločevanje polnih palet in ustvarjanje novega naročila za polne palete		zahtevano izločanje polnih palet? količina?	ново naročilo za polne palete kreirano
46 - računalniško tehtanje blaga	K(46,11)	izračunajo se volumni in potrebno št. RLC za vsako naročilo napiše se pravo zaporedje za komisijoniranje na RLC		znani volumni artiklov, naročila dokončna	volumensko ovrednoteno naročilo-znano število RLC
47 - izpisovanje liste za disponiranje	K(47,11)	Določimo odpremi box za vsako naročilo in določimo datum ter čas odpreme		znano število RLC, urniki	naročila so razporejena glede na urnike, izpisana lista
48 - dokončno razporejanje naročil na vožnje	K(48,11)	Disponent razdeli še nerazporejena naročila na vožnje. Razporedi se tako, da se napolnijo tovornjaki, ki potujejo v isto smer, z naročili v tisto smer		ni stalnega urnika	vsa naročila razporejena na odpreme cone/vožnje
49 - obveščanje kupca	K(49,11)	potrdilo o predvideni dobavi se pošlje v kupčev IS		kupec ima predvideno avtomatsko potrjevanje	potrditev naročila
50 - izpis kom. listov, izdelava zapisa fakture	K(50,11)	Disponent kreira komisijonirni list za vsak RLC, rezervacija zaloge se zmanjša, zaloga se zmanjša, kreira se zapis računa in postavke		razporejeno naročilo	komisijonirni listi za RLC in palete v bazi, zapis računa kreiran
	K(50,12)	pregleduje evidenco razpisanih komisijonirnih listov			komisijonirni listi v sistemu
51 - razporejanje komisijonirnih listov med komisijonarje	K(51,12)	Vodja komisijonarjev razporedi kom. liste med komisijonarje - vnese njihove številke		list še ni razporejen?	komisijonirni listi dodeljeni komisijonarjem

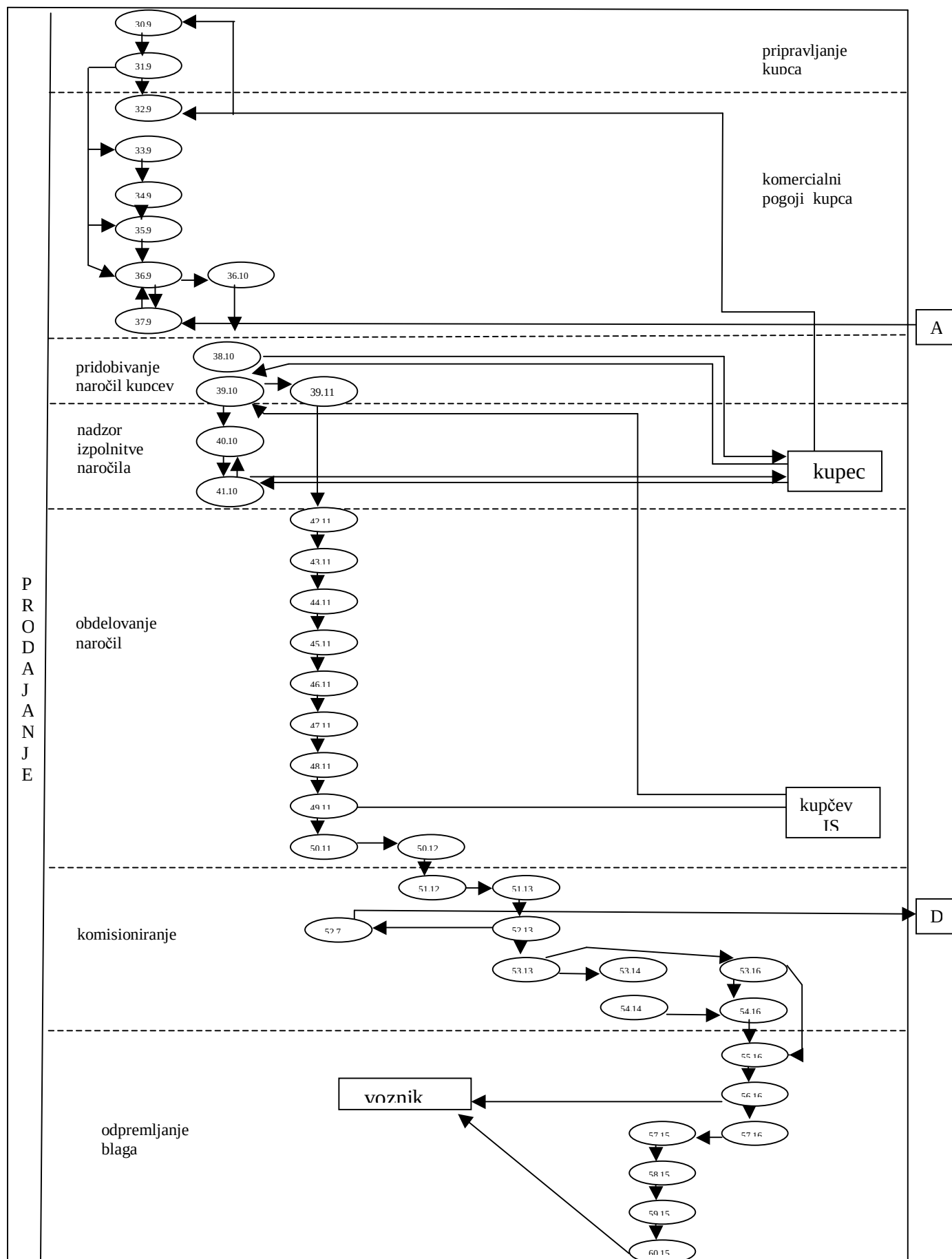
Activity	Task Code	Description	Time	Condition	Input/Output
51 - razporejanje komisijonirnih listov med komisijonarje	K(51,13)	Komisijonar aktivira komisijonirni list na RF terminalu, ko zaključi predhodnega		dodeljen komisijonirni list	kom. list v RF, status 1
52 - komisijoniranje-priprava postavke	K(52,7)	viličarist prejme signal za takojšnje dopolnjevanje		zaloga na komisijonirni lokaciji ne zadošča	zahtevek za dopolnjevanje
	K(52,13)	blago je na kom. lokaciji. Komisijonar vzame blago, skenira SSCC palete, potrdi količino. Če ni dovolj zaloge, sistem signalizira viličaristu dopolnjevanje		zaloga na lokaciji zadošča?	podatki o izdani količini, roku, seriji vnešeni na postavko računa, eventuelno kreiran zahtevek za dopolnjevanje
53 - kontrolno tehtanje, zaključek komisijoniranja	K(53,13)	tehtanje RLC, izpis označevanje nalepke za RLC, označitev RLC		teža v toleranci > status 3, sicer 2	status rlc 2 ali 3, označevalna nalepka, signal kontrolorju
	K(53,14)	glede na čas odpreme izbere naslednji RLC za kontrolo		teža RLC ni v toleranci, status 2	lokacija na odpremi coni, kjer je RLC, številka RLC
	K(53,16)	evidenca kontroliranih RLC		status RLC 3	številka RLC, status, lokacija
54 - kontrola RLC	K(54,14)	Kontrolor skenira oznako RLC, preveri vsebino, po potrebi dodaja in odvzema artikla, popravi količino, potrdi kontrolo status 3		status 2	potrjena kontrola, status 3
	K(54,16)	evidenca kontroliranih RLC		status 3	številka RLC, status, lokaci
55 - izbor paketa RLC za odpremo	K(55,16)	odpremnik izbere vožnjo, ki jo bo nakladal, vozniku glede na čas odpreme in prisotnost voznika		vsi RLC vožnje status 3, voznik prisoten	seznam RLC, lokacija na odpremi coni
56 - predaja blaga vozniku	K(56,16)	izbira in skeniranje ustreznih RLC, predaja blaga vozniku		RLC pripada izbrani vožnji	blago na kamionu, zabeleženo število RLC, naloženi RLC imajo status 4
57 - kontrola voznika za potni nalog	K(57,15)	administrator pregleduje vožnje, primerne za izpis dokumentacije		status potnega naloga 1	potni nalog v sistemu
	K(57,16)	odpremnik preveri šifro voznika, potrdi konec nalaganja vožnje		voznik v bazi	status potnega naloga 1-naloženo, znani naloženi RLC, voznik
58 - izpisovanje računov	K(58,15)	izpisovanje računov za celo vožnjo		vsi RLC računa naloženi, status 4, nalog	izpisan račun, RLC status 5
59 - izpisovanje potnega naloga za vožnjo	K(59,15)	izpis potnega naloga		računi izpisani, RLC stat 5, nalog stat 1	potni nalog izpisan, status 2, status računa T
60 - predajanje potnega naloga, računov vozniku	K(60,15)	predajanje potnega naloga, računov vozniku, voznik podpiše potni nalog in se zadoži za blago in dokumente		izpisana dokumentacija	potni nalog, računi
61 - vračanje prazne embalaže	K(61,17)	prevzemnik embalaže prevzame prazno transportno embalažo in vračljivo embalažo za blago. Količine transportne embalaže vpiše na potni nalog, količine embalaže za blago primerja z količinami na sprejemnici embalaže, ki jo je oddal voznik.		voznik vrne prazno embalažo in dokumente	potni nalog, sprejemnica embalaže
62 - kompletiranje odpreme dokumentacije	K(62,15)	administrator sprejme račun			račun
	K(62,17)	prevzemnik embalaže potni nalog in sprejemnice embalaže odda embalažnemu referentu, podpisane račune pa v odpremo pisarno		embalaža prevzeta	potni nalog, sprejemnica embalaže, račun
	K(62,18)	referent za embalažo kompletira sprejemnice embalaže po strankah			sprejemnica embalaže
63 - arhiviranje računov, komisijonirnih listov	K(63,15)	administrator posreduje račune skupaj s komisijonirnimi listi v arhiv			račun, komisijonirni list
64 - zaključevanje potnega naloga	K(64,18)	embalažni referent vnese prevožene kilometre, čas, in vrnjeno transportno embalažo v potni nalog.			potni nalog
65 - izdaja dobropisa za embalažo	K(65,18)	referent za embalažo izda dobropis kupcu za vrnjeno transportno (RLC, palete) in vračljivo embalažo blaga (zaboji, steklenice)		kupec je vrnil embalažo	dobropis
66 - kontrola salda transportne embalaže za vožnjo	K(66,18)	referent za embalažo izpiše saldo transportne embalaže za vožnjo, če ni (prevzeta + vrnjena od stranke - oddana stranki - vrnjena prevzemniku embalaže) = 0, terja voznika		usklajen saldo	potni nalog, dobropisi embalaže
67 - arhiviranje dobropisov za embalažo, potnih nalogov	K(67,18)	referent za embalažo posreduje dobropise in potne naloge v arhiv			dobropis za embalažo, potni nalog
68 - vračanje blaga od kupca	K(68,14)	kontrolor sprejme vrnjeno blago, obrazec o neskladnosti od voznika ali kupca		blago kupljeno pri nas	obrazec o neskladnosti, podatki o računu, postavki, ko je bilo blago prodano
69 - ločevanje vrnjenega blaga glede na uporabnost	K(69,14)	glede na rok uporabe in fizični izgled blaga loči blago na uporabno in neuporabno			rok uporabe
70 - dopolnjevanje obrazca za neskladnost	K(70,14)	kontrolor dopolni obrazec s podatkom o uporabnosti blaga. Obrazec podpiše in preda referentu za reklamacije			obrazec o neskladnosti
	K(70,18)	referent za reklamacije sprejme obrazec o neskladnosti			obrazec o neskladnosti
71 - izdaja dobropisa kupcu	K(71,18)	referent za reklamacije izda dobropis kupcu		podpisan obrazec o neskladnosti	dobropis, obrazec o neskladnosti

Activity	Task Code	Description	Time	Condition	Input/Output
72 - arhiviranje dobropisa, obrazca o neskladnosti	K(72,18)	referent za embalažo posreduje dobropis v arhiv			dobropis
73 - označevanje blaga za redno vskladiščenje	K(73,14)	kontrolor kompletira paleta istega blaga, kreira postavko vskladiščenja - premik		blago ustrezno	postavka vskladiščenja-premik
74 - predaja vrnjenega blaga skladiščniku	K(74,6)	skladiščnik prejme vrnjeno blago od kontrolorja			blago, SSCC palete
	K(74,14)	kontrolor preda blago skladiščniku			SSCC palete
75 - določanje lokacije in vskladiščenje	K(75,6)	skladiščnik skenira SSCC, določi lokacijo in jo vnese v RF terminal. Izpiše nalepko za paletu in jo nalepi. Blago preda viličaristu - kreira nalog za vskladiščenje			SSCC, nalog za vskladiščenje-premik
	K(75,7)	viličarist prejme vrnjeno blago od skladiščnika, skenira SSCC, izvrši postopek vskladiščenja			SSCC palete
76 - označevanje neuporabnega blaga	K(76,14)	kontrolor kompletira paletu neuporabnega blaga, kreira postavko vskladiščenja in SSCC kodo. Vnese rok trajanja, lokacijo za neuporabno blago, razlog neustreznosti. Izpiše nalepko za paletu SSCC in jo nalepi.		blago neustrezno	SSCC palete
77 - dopolnjevanje komisionirne lokacije	K(77,6)	skladiščnik identificira prazno komisionirno lokacijo, kreira zahtevek za dopolnjevanje. Lahko lokacijo vidi, ali pa izpiše listo praznih lokacij, ki še imajo rezervno zalogo		komisionirna lokacija je prazna	lista praznih lokacij
	K(77,7)	glede na vrstni red zahtevkov viličarist izbere lokacijo za dopolnjevanje, blago premakne iz določene rezervne lokacije. Skenira SSCC palete, blago premakne na kom. lokacijo, skenira kodo kom. lokacije, da potrdi premik		zahtevek za premik	SSCC, koda lokacije, zahtevek za premik
78 - identifikacija neustreznega blaga	K(78,6)	skladiščnik glede na rok trajanja ali poškodbe blaga identificira neustrezno blago. Pomaga si z pregledom rokov trajanja blaga na zalogi		rok blizu izteka	Lista blaga s kritičnim rokom trajanja
79 - označevanje neustreznega blaga	K(79,6)	skladiščnik označi blago z novo SSCC kodo, lokacija v posebnem delu skladišča			SSCC palete, nalog za premik v posebni del skladišča
80 - predaja neustreznega blaga	K(80,6)	skladiščnik preda blago viličaristu		premik kreiran	nalog za premik
	K(80,7)	viličarist prejme neustrezno blago			SSCC palete
81 - vskladiščenje neuporabnega blaga	K(81,7)	viličarist prejme neustrezno blago, skenira SSCC, izvrši postopek vskladiščenja v poseben del skladišča			SSCC palete
	K(81,14)	kontrolor preda neustrezno blago viličaristu,			SSCC palete, blago
82 - obveščanje o neustreznem blagu	K(82,1)	sprejme informacijo			obrazec o neskladnosti, seznam SSCC
	K(82,2)	sprejme informacijo			obrazec o neskladnosti, seznam SSCC
	K(82,6)	skladiščnik poroča o vrstah in količini neustreznega blaga na zalogi,		zaloga neustreznega blaga	seznam posameznih palet, s SSCC kodami, obrazec o neskladnosti
83 - dogovor o vračilu blaga dobavitelju	K(83,1)	vodja nabave obvesti vodjo sdc o dogovoru z dobaviteljem			seznam artiklov, ki jih dobavitelj ne bo prevzel, SSCC
	K(83,2)	vodja nabave se dogovori o možnosti vračila blaga dobavitelju. V primeru dogovora se blago vrne, sicer predlaga odpis blaga			seznam artiklov, SSCC, količin, ki naj jih dobavitelj prevzame
	K(83,3)	vodja nabave obvesti nabavnega komercialista o dogovoru za vračilo blaga			seznam artiklov in količin za vračanje, SSCC
84 - odločitev o odpisu blaga	K(84,1)	vodja sdc sprejme odločitev o odpisu blaga			seznam artiklov za odpis, SSCC
	K(84,6)	skladiščnik sprejme odobritev odpisa			seznam artiklov za odpis, SSCC
85 - uničenje blaga	K(85,6)	skladiščnik z delavci komisijko uniči blago. Z RF terminalom kreira odpis, skenira SSCC palet uničenega blaga. Izpiše zapisnik o uničenju, podpiše skupaj s člani komisije.		podpisan seznam vodje sdc	seznam artiklov za odpis, SSCC, odpis, zapisnik o uničenju
	K(85,8)	administrator vidi nepotrjen odpis, prejme zapisnik o uničenju			vnešen odpis, zapisnik o uničenju
86 - vnos odpisa blaga	K(86,8)	administrator na podlagi komisijke podpisanega zapisnika o uničenju potrdi in izpiše odpis			odpis blaga, artikli, količine, SSCC
87 - posredovanje dokumentacije o odpisu v	K(87,8)	izpisan odpis skupaj z zapisnikom posreduje v likvidaturo		izpisan odpis	odpis, zapisnik o uničenju
88 - vnos predloga vračila blaga dobavitelju	K(88,3)	nabavni komercialist vnese artikle, količine in dogovorjene cene za vrnjeno blago			predlog vračila blaga dobavitelju
89 - potrditev in izpis predloga vračila blaga	K(89,3)	nabavni komercialist potrdi in izpiše predlog vračila blaga			izpisan predlog vračila
	K(89,6)	skladiščnik pripravi blago za izdajo, skenira SSCC, s tem potrdi istovetnost blaga			predlog vračila blaga dobavitelju, SSCC
90 - izdaja blaga iz skladišča na prevzemno rampo	K(90,6)	skladiščnik naroči viličaristu, da odpelje blago na prevzemno rampo,		dobaviteljev voznik na rampi	SSCC palet za vračanje dobavitelju
90 - izdaja blaga iz skladišča na prevzemno rampo	K(90,7)	viličarist odpelje blago na prevzemno rampo, skenira SSCC in vpiše lokacijo prevzemne rampe			SSCC, potrjen premik
91 - predaja blaga dobavitelju	K(91,6)	skladiščnik preda blago dobavitelju, izpiše dokončni dokument vračila, oba podpišeta spisek SSCC, artiklov, količin iz vračila			podpisano vračilo
	K(91,7)	viličarist prepelje vse SSCC, predvidene za vračanje			SSCC palet za vračanje dobavitelju
92 - posredovanje predloga vračila v prevzemno pisarno	K(92,6)	skladiščnik posreduje podpisano vračilo v prevzemno pisarno			podpisano vračilo
	K(92,8)	administrator v prevzemu prejme podpisano vračilo		podpisan dokument	podpisano vračilo
93 - posredovanje vračila blaga v likvidaturo	K(93,8)	administrator posreduje podpisano vračilo v likvidaturo		podpisano vračilo	vračilo dobavitelju

Slika 1: Procesni model 1 / 3



Slika 2: Procesni model 2 / 3



Slika 3: Procesni model 3 / 3

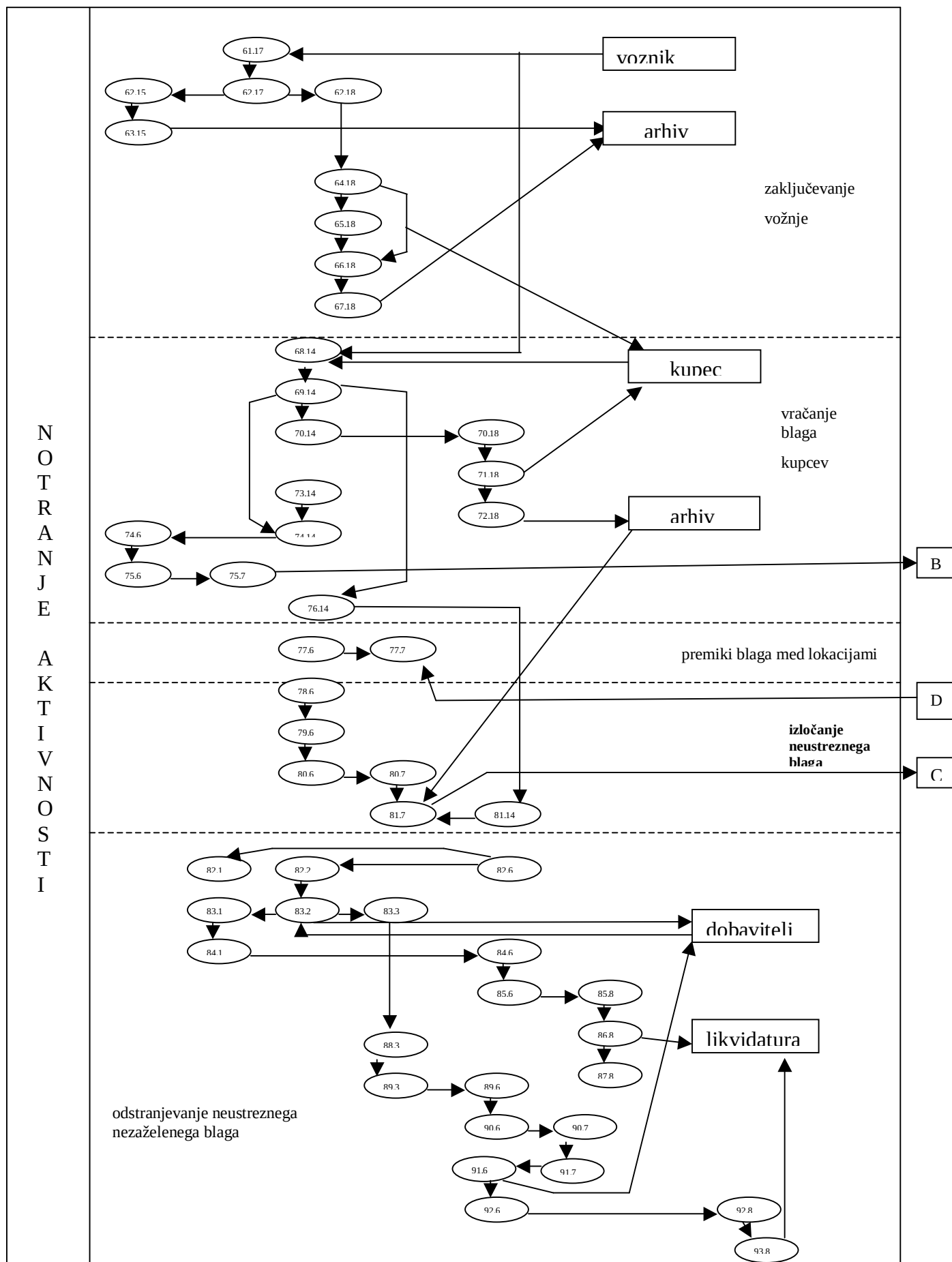


Tabela 3: Tabela operacij**1 / 4**

Naloga	Operacija	Razred
K1,3	KreirajDobavitelja	Dobavitelj
K2,3	SpremeniDobavitelja	Dobavitelj
K5,3	KreirajArtikel	Artikel
K6,3	KreirajKalkulacijo	KalkulacijaOsnovna
K9,3	SpremeniKalkulacijo	KalkulacijaOsnovna
K11,3	DeaktivirajArtikel	Artikel
K12,2	PreglejObvestilo	KomercialnoObvestilo
K12,3	KreirajObvestilo	KomercialnoObvestilo,
K12,3	PreglejArtikel	Artikel
K12,3	PreglejKalkulacijo	KalkulacijaOsnovna
K12,9	PreglejObvestilo	KomercialnoObvestilo
K13,3	PreglejProdajo	Artikel, PostRačuna
K14,3	SpremeniZalogo	Zaloga
K15,3	KreirajPrednaročilo	Prednaročilo, PostPrednaročila
K15,3	PredlagajKoličino	Zaloga
K16,3	SpremeniPrednaročilo	Prednaročilo, PostPrednaročila
K17,3	PotrdiPrednaročilo	Prednaročilo, PostPrednaročila
K17,4	PreglejNovaPrednaročila	Prednaročilo
K18,4	DoločiPrevzemnika	Prednaročilo, Prevzemnik
K18,5	IzberiPrednaročilo	Prednaročilo
K19,5	PreglejPrednaročilo	Prednaročilo
K20,5	VnesiKoličino	PostPrednaročila
K22,5	VnesiDobavnico	Prednaročilo
K23,5	SpremeniLogPodatke	Artikel
K24,5	KreirajSSCC	SSCCPremik
K24,5	DoločiLokacijo	SSCCPremik, Lokacija, ZalogaSSCC
K25,5	IzpišiSSCC	SSCCPremik
K25,7	IzberiSSCC	SSCCPremik, Viličarist
K26,7	PotrdiPremikSSCC	SSCCPremik, ZalogaSSCC
K28,8	ZaključiPrevzem	Prednaročilo, PostPrednaročila
K28,8	SpremeniZalogo	Zaloga
K31,9	KreirajKupca	Kupec
K32,9	SpremeniKupca	Kupec
K33,9	KreirajPogojeKupca	PogojiKupca, Kupec,
K34,9	KreirajSkupinoKupcev	SkupinaKupcev
K35,9	DoločiSkupinoKupcev	Kupec
K36,9	KreirajCenikSkupineKupcev	CenikSkupineKupcev,
K37,9	SpremeniCenikSkupineKupcev	CenikSkupineKupcev, KalkulacijaOsnovna
K38,10	UgotoviCenoKupca	Artikel, CenikSkupineKupcev

Naloga	Operacija	Razred
K39,10	KreirajNaročilo	Naročilo, PostNaročila
K39,11	PreglejNaročilo	Naročilo
K40,10	VrniStatus	Naročilo
K40,10	VrniStatus	RLC
K40,10	VrniStatus	Račun
K41,10	PreglejNaročilo	Naročilo
K42,11	PreglejTerjatve	Kupec, Terjatve
K42,11	PreglejLimit	Kupec
K42,11	IzberiNaročilo	Naročilo
K43,11	IzberiNaročilozaZdruževanje	Naročilo
K43,11	ZdružiNaročilo	Naročilo
K44,11	RezervirajZalogo	PostNaročila, Zaloga
K45,11	IzberiPolnoPalet	PostNaročila, Artikel
K45,11	KreirajNaročilo	Naročilo
K45,11	PrepišiPostavko	PostNaročila
K46,11	DoločiVolumen	PostNaročila
K46,11	KreirajRLC	RLC
K46,11	DoločiRLC	PostNaročila
K47,11	DoločiVožnjoRLC	Kupec, Urnik, Vožnja, RLC
K48,11	SpremeniVožnjoRLC	RLC
K49,11	PreglejNaročilo	Naročilo, PostNaročila
K50,11	SpremeniZalogo	Zaloga
K50,11	InitStatusRLC	RLC
K50,11	KreirajRačun	Račun, PostRačuna
K50,11	KreirajPotniNalog	PotniNalog
K50,12	PreglejRLC	RLC
K51,12	DoločiKomisionarjaRLC	RLC
K51,13	IzberiRLC	RLC
K52,13	KreirajSSCC	SSCCPremik
K52,13	PotrdiPostavkoRačuna	PostRačuna
K52,13	PotrdiSSCC	SSCCPremik, ZalogaSSCC
K53,13	PreveriTežoRLC	RLC
K53,13	VrniTežoPostavke	PostRačuna
K53,14	PreglejRLCstat	RLC
K53,16	PreglejRLCstat	RLC
K54,14	IzberiRLC	RLC
K54,14	DoločiKontrolorRLC	RLC
K54,14	SpremeniPostavkoRačuna	PostRačuna, Zaloga
K54,14	PotrdiPostavkoRačuna	PostRačuna
K54,14	PotrdiKontroloRLC	RLC

Naloga	Operacija	Razred
K54,16	PreglejRLCstat	RLC
K56,16	OdpremiRLC	RLC
K57,15	PreglejStatusRLCVožnje	Vožnja, RLC
K57,16	DoločiOdpremnik	PotniNalog
K57,16	PotrdiOdpremoNaloga	Vožnja, PotniNalog
K58,15	IzpišiRačuneNaloga	Vožnja, Račun
K59,15	IzpišiPotniNalog	PotniNalog
K64,18	VnesiKmEmbalaža	PotniNalog
K65,18	KreirajDobropis	Dobropis, PostDobropisa
K66,18	PreglejEmbaVožnje	Dobropis, PotniNalog
K71,18	KreirajDobropis	Dobropis, PostDobropisa
K73,14	KreirajSSCC	SSCCPremik
K74,6	IzberiSSCC	SSCCPremik
K74,14	IzpišiSSCC	SSCCPremik
K75,6	DoločiLokacijo	SSCCPremik, Lokacija, ZalogaSSCC
K75,6	IzpišiSSCC	SSCCPremik
K75,7	IzberiSSCC	SSCCPremik
K75,7	PotrdiPremikSSCC	SSCCPremik, ZalogaSSCC
K76,14	KreirajSSCC	SSCCPremik
K76,14	DoločiLokacijo	SSCCPremik, Lokacija, ZalogaSSCC
K76,14	IzpišiSSCC	SSCCPremik
K77,6	VrniZalogoLokacije	ZalogaSSCC
K77,6	KreirajSSCC	SSCCPremik
K77,7	IzberiSSCC	SSCCPremik
K77,7	PotrdiPremikSSCC	SSCCPremik, Lokacija, ZalogaSSCC
K78,6	VrniRokZalogeLokacije	ZalogaSSCC, Lokacija
K79,6	KreirajSSCC	SSCCPremik
K79,6	DoločiLokacijo	SSCCPremik, Lokacija, ZalogaSSCC
K79,6	IzpišiSSCC	SSCCPremik
K80,6	KreirajSSCC	SSCCPremik
K81,7	IzberiSSCC	SSCCPremik
K81,7	PotrdiPremikSSCC	SSCCPremik, Lokacija, ZalogaSSCC
K81,14	KreirajSSCC	SSCCPremik, PostSezSSCC
K82,1	PreglejSeznamSSCC	SeznamSSCC
K82,2	PreglejSeznamSSCC	SeznamSSCC
K82,6	VrniZalogoLokacije	ZalogaSSCC
K82,6	KreirajSeznamSSCC	SeznamSSCC
K83,2	VnesiDogovor	PostSeznamaSSCC

Naloga	Operacija	Razred
K84,1	VnesiOdpis	PostSeznamSSCC
K84,6	PreglejSeznamSSCC	SeznamSSCC
K85,6	KreirajOdpis	Odpis, PostOdpisa
K85,6	IzberiSSCC	SSCCPremik
K85,6	PotrdiPremikSSCC	SSCCPremik, ZalogaSSCC
K85,6	IzpišiZapisnikUničenja	Odpis
K85,8	PreglejOdpis	Odpis
K85,8	PotrdiOdpis	Odpis,
K85,8	SpremeniZalogo	Zaloga
K85,8	IzpišiOdpis	Odpis,
K88,3	KreirajPrednaročilo	Prednaročilo, PostPrednaročila
K89,3	IzpišiPredlogVračiloDobavitelju	Prednaročilo
K89,6	IzberiSSCC	SSCCPremik
K90,6	KreirajSSCC	SSCCPremik
K91,7	PotrdiPremikSSCC	SSCCPremik, ZalogaSSCC
K91,6	ZaključiVarčiloDobavitelju	Prednaročilo
K91,6	SpremeniZalogo	Zaloga