

Metodologija razvoja informacijskih sustava

Lerga, Rebeka

Undergraduate thesis / Završni rad

2014

Degree Grantor / Ustanova koja je dodijelila akademski / stručni stupanj: **University of Rijeka, Faculty of Humanities and Social Sciences / Sveučilište u Rijeci, Filozofski fakultet**

Permanent link / Trajna poveznica: <https://um.nsk.hr/um:nbn:hr:186:816245>

Rights / Prava: [In copyright](#)/[Zaštićeno autorskim pravom.](#)

Download date / Datum preuzimanja: **2025-01-26**



Repository / Repozitorij:

[Repository of the University of Rijeka, Faculty of Humanities and Social Sciences - FHSSRI Repository](#)



SVEUČILIŠTE U RIJECI

FILOZOFSKI FAKULTET U RIJECI

Dvopredmetni studij engleskog jezika i informatike

Metodologija razvoja informacijskog sustava

ZAVRŠNI RAD

STUDENT: Rebeka Lerga

MENTOR: Sanja Čandrić, dr. sc.

Rijeka, rujan 2014.

Sadržaj

1. Uvod	1
1.1. Ukratko o informacijskim sustavima.....	1
1.2.1. Problemi informacijskih sustava.....	2
1.2.2. Načini razvoja informacijskih sustava	3
1.2.3. Modeli informacijskih sustava	3
2. Metodologije	5
2.1. Vrste metodologija	6
2.1.1. Zadaci specijalizirane metodologije	7
2.1.2. CASE alati	8
2.2. Metodologija SSADM.....	8
2.2.1. Životni ciklus metodologije SSADM	9
2.2.1.1. Modul izvodivosti	10
2.2.1.2. Modul analize potreba.....	10
2.2.1.3. Modul specifikacije potreba	12
2.2.1.4. Modul logičke specifikacije sustava	13
2.2.1.5. Modul fizičkog dizajna.....	13
2.3. Metodologija MIRIS.....	14
2.3.1. Životni ciklus metodologije MIRIS	14
2.3.1.1. Logičko oblikovanje (projektiranje).....	15
2.3.1.2. Fizičko oblikovanje (izgradnja IS).....	15
3. Praktični rad	22
4. Zaključak	36
5. Popis slika	37
6. Literatura	38

SAŽETAK

Završni rad obrađuje metodologiju razvoja informacijskog sustava. U uvodu navodi načine i modele razvoja informacijskog sustava, probleme pri uvođenju informacijskih sustava, općenito definira metodologije, navodi vrste metodologija te definira zadatke specijaliziranih metodologija. U razradi detaljnije opisuje i uspoređuje metodologiju MIRIS i metodologiju SSADM. Predstavlja životne cikluse razvoja informacijskog sustava kako su definirani prema tim metodologijama. Završni rad također ima i praktični dio unutar kojeg predstavlja modele procesa i podataka poduzeća KG d.o.o.

KLJUČNE RIJEČI:

informacijski sustav, metodologija, metoda, MIRIS, SSADM, model procesa, model podataka

1. Uvod

Tema završnog rada je metodologija razvoja informacijskih sustava. Uz općenita poglavlja o metodologijama, rad detaljnije objašnjava i uspoređuje pristupe razvoju informacijskog sustava metodologija MIRIS i SSADM.

Uz teorijski dio završni rad također ima i praktičan dio. Za praktičan rad korišteni su poslovni procesi i dokumenti poduzeća KG d.o.o.

1.1. Ukratko o informacijskim sustavima

Sukladno danas najčešće primjenjivanom sustavnom pristupu, pojave u stvarnom pa i u zamišljenom svijetu shvaćaju se i promatraju kao cjeline (sustavi). Jedna od takvih pojava su i informacijski sustavi. (Panian, i dr., 2010) Informacijski sustav je sustav u kojem po određenoj proceduri zajednički djeluju ljudi, sredstva informacijske tehnologije s ciljem da onima kojima je to potrebno pravovremeno učini dostupnima određene podatke odnosno informacije. (Čerić, i dr., 1998)

Ljudi su danas neprestano u doticaju sa informacijskim sustavima. Današnje je društvo informacijsko, informacija je proizvod i roba za tržište, glavnina financijskih tokova odvija se kroz informacijske sustave, nabava i prodaja robe na svjetskom tržištu nezamislive su bez informacijskih sustava. Svijetom dominiraju informacije i znanje. Nemoguće je klasičnim metodama proizvoditi ili pratiti nastanak novih informacija. Bez informacijskog sustava život je suvremenog čovjeka nezamisliv. Bilo kakav zapis na papiru, kamenu, magnetskom ili drugom mediju dio je informacijskog sustava. Informacijski sustavi su svagdje. Svaki čovjek ima u svojem umu vlastiti informacijski sustav. Svaka tvrtka, organizacija, udruga, institucija ima informacijski sustav. Za uobičajeno odvijanje funkcija sustava i da bi se moglo upravljati procesima sustava, treba prikupiti i obraditi velik broj informacija o stanju poslovnog sustava, što premašuje čovjekove mogućnosti. Stoga se ljudi služe pomoćnim sredstvima među kojima središnju ulogu ima računalo kao mjesto za čuvanje podataka i kao alat za obradbu i prezentaciju informacija. (Pavlič, 2011)

Pod pojmom informacijskog sustava podrazumijeva se onaj dio stvarnog (organizacijskog) sustava koji služi pretvorbi ulaznih u izlazne informacije. Svako poduzeće, odnosno poslovni sustav ima svoj informacijski sustav koji prikuplja, pohranjuje, čuva, obrađuje i isporučuje informacije. Informacije se u informacijskom sustavu pohranjuju u obliku podataka. Proces razvoja informacijskog sustava složen je postupak, a njegova složenost proizlazi iz složenosti poslovnog sustava koji podržava. Razvojem informacijskog sustava obuhvaćen je niz

aktivnosti poput planiranja, analize, oblikovanja, izgradnje, uvođenja u rad i održavanja. Navedene aktivnosti treba organizirati kao projekt u kojem sudjeluju korisnici informacijskog sustava i informatičari koji izgrađuju informacijski sustav. Pri provedbi projekta razvoja informacijskog sustava koriste se brojne metode i tehnike. (Panian, i dr., 2010)

Središnji dio informacijskog sustava su same informacije zajedno s tehnologijom pomoću koje se izvode procesi nad informacijama. Najvažnija tehnologija obradbe podataka jest informacijska tehnologija. (Pavlič, 2011) Informacijska tehnologija u posljednjem je desetljeću doživjela buran razvoj i ostvarila golem utjecaj na sva područja rada i života razvijenih društava. Njen se razvoj i dalje odvija nesmanjenom brzinom, uz ogromna izdvajanja za istraživanja i razvoj koja dovode do velikog broja novih postupaka, metoda i uređaja. Informacijska tehnologija jedna je od ključnih generičkih tehnologija koja prodire u sve sfere gospodarstva, znanosti, društvenog i privatnog života i unosi u sva ta područja radikalne promjene. (Čerić, i dr., 1998)

Informacijski sustav gradi se za potrebe organizacijskog sustava. Struktura organizacijskog sustava osnova je za modeliranje strukture informacijskog sustava. Kreiran informacijski sustav slika je organizacijskog sustava, a na apstraktnoj razini postoji model organizacijskog sustava koji je sličan modelu informacijskog sustava. Organizacijski sustav postoji u realnosti, neovisno o našoj spoznaji o njemu. Stvarni sustavi predmet su našeg zanimanja i za njih razvijamo informacijske sustave: na osnovi njihove strukture gradi se model strukture informacijskog sustava. (Pavlič, 2011)

Uvođenje informacijske tehnologije vrlo je složen posao, kako tehnološki tako i organizacijski i psihološki. Mnoge organizacije i menadžeri nisu dorasli tim problemima, što dovodi do velikog kašnjenja, premašaja predviđenih troškova i problema u radu. Dobar informacijski sustav, iako se čini nevidljivim, nezaobilazan je dio poslovnog sustava. Poslovni sustav, da bi preživio, mora imati svoj informacijski sustav i u njemu razrađene postupke informacijskih aktivnosti. (Čerić, i dr., 1998)

1.2.1. Problemi informacijskih sustava

U projektiranju i izgradnji poslovnih informacijskih sustava postoje dvije metode (Srića, i dr., 1999):

- Metoda improvizacije - izgradnja sustava izgradnja sustava temelji se na postupnosti, polazi se od rješavanja problema i razvijanja pojedinih aplikacija onako kako ih nametnu potrebe, slučaj ili odluke voditelja. Informacijski sustav se gradi od dijelova prema cjelini bez plana cjelovitog informacijskog sustava.

- Sustavni pristup - izgradnja sustava započinje jasnim definiranjem bitnih obilježja cjeline, a tek zatim počinje realizacija njezinih pojedinačnih dijelova.

Cilj je svake metode razvoja informacijskog sustava načiniti kvalitetan i uspješan sustav koji treba zadovoljiti utvrđene zahtjeve te biti razvijen na vrijeme i unutar planiranih troškova.

Problemi s informacijskim sustavima su sljedeći (Pavlič, 2011):

1. Model informacijskog sustava ne odgovara organizacijskom sustavu
2. Razvoj informacijskog sustava kasni
3. Stvarni troškovi razvoja informacijskog sustava dvostruko prelaze planirane troškove
4. Velik dio poslovnih sustava nije obuhvaćen automatskom obradom podataka
5. Nedostaju kadrovi za razvoj softvera
6. Odjeli za razvoj softvera uglavnom se bave održavanjem postojećih informacijskih sustava
7. Nerazumijevanje korisnika ili rješavanje krivog problema
8. Mijenjanje zahtjeva
9. Nepostojanje strategijskog plana informatizacije
10. Nekorištenje metoda za dizajn informacijskog sustava
11. Slaba kontrola projekta
12. Slaba kvaliteta projekta i programskoga proizvoda

1.2.2. Načini razvoja informacijskih sustava

Pri razvoju informacijskog sustava, poduzeća mogu koristiti jedan ili više načina razvoja (Panian, i dr., 2010):

- Vanjski razvoj (Outsourcing)
- Vlastiti razvoj (Insourcing)
- Kupnju gotovog rješenja
- Unajmljivanje gotovog rješenja (Hosting)

1.2.3. Modeli informacijskih sustava

Proces razvoja informacijskog sustava sastoji se od mnogobrojnih aktivnosti koje se mogu grupirati u faze. Prepoznamo tri osnovne faze: projektiranje, izgradnja i održavanje informacijskih sustava.

Izgradnju informacijskog sustava treba organizirati kao projekt, tj. kao skup organiziranih poslova koje valja obaviti u određenom roku da bi se izgradio određeni proizvod. Praksa

pokazuje da se izgradnji informacijskog sustava različito pristupa, kako organizacijski tako i metodološki, te da je uspjeh projekata razvoja informacijskog sustava vrlo različit. (Čerić, i dr., 1998)

Modeli razvoja opće su ideje o mogućim putovima razvoja informacijskog sustava. Metodologije su intelektualni proizvodi koji uključuju ideje iz više različitih modela i sadržavaju vlastite ideje o redosljedu modeliranja te imaju detaljno razrađen i opisan postupak modeliranja informacijskog sustava. (Pavlič, 2011)

Linearan model razvoja informacijskog sustava još se naziva i vodopadnim jer se poslovi obavljaju po fazama, onim redom kako faze prirodno slijede u razvoju sustava. U svakoj se fazi proizvode određeni rezultati (specifikacije sustava, dijagrami sustava, programi, dokumentacija i sl.) koji predstavljaju ulaz u iduću fazu. To omogućuje da nakon svake faze razvoja poslovodstvo može ocijeniti napredovanje projekta i odlučiti o njegovoj daljnjoj realizaciji. (Čerić, i dr., 1998)

Pseudostrukturni model razvoja informacijskog sustava kaskadni je model s ugrađenim mehanizmom povratne veze na ranije faze. To omogućuje primjenu rezultat prethodnih faza i daljnji nastavak u prihvaćenim izmjenama.

Strukturni model i dalje definira strogo odvojene faze razvoja s mehanizmom povratne veze, uz napomenu da pronalazi faze koje mogu teći usporedno jer su relativno neovisne. U osnovnoj se ideji predlaže paralelno izvođenje faza logičkog oblikovanja i paralelno izvođenje faza fizičke izvedbe sustava. Taj je model najčešći u praktičnoj primjeni jer ima najveću slobodu a zahtjeva projekt prije izradbe bilo kakvog proizvoda. (Pavlič, 2011)

V-model nastao je na osnovi činjenice da se sustav i proizvod sastoje od dijelova koji se dalje mogu sastojati od dijelova i da je tako moguće više razina hijerarhije. Smisao je tog modela da se najprije logički razrađuje sustav od cjeline prema detaljima sve do logičkog modula koji će kodiranjem postati programski modul, a potom se od najsitnijih programskih modula grade sve veće cjeline, i to tako do cijelog sustava. (Pavlič, 2011)

Prototipski model zasnovan je na zamisli da se izgradi prototip proizvoda koji nije gotov proizvod i on se provjerava i poboljšava dok ne zadovolji potrebe. Prototipski je pristup prigodan za softverske proizvode ili dijelove informacijskog sustava. Prototipski model podržava iterativan pristup prema kojemu se konačnom cilju približavamo mehanizmom

povratne veze. Taj je pristup popraćen uporabom naprednih softverskih alata za brz razvoj programskog proizvoda, u kojem se brzo izrađuje ciljani softverski proizvod radni model. Radni model daje se korisniku na uvid, a tada korisnik daje primjedbe za popravak, poboljšanja i izradbu novoga radnoga modela. Prije izradbe prvog prototipa potrebno je najprije analizirati korisničke zahtjeve i dizajnirati radni model. (Pavlič, 2011)

Evolucijski model razvoja temelji se na modernim informatičkim sredstvima kojima se brže i jednostavnije može razviti početna verzija informacijskog sustava, korisnicima dati sustav odmah na upotrebu, zatim ga postupno doradivati poboljšanjem prethodne verzije dok se ne dobije zadovoljavajući informacijski sustav. Dijelovi sustava razvijaju se u početku bez preciznih specifikacija, pa se tako mogu izgrađivati manji dijelovi informacijskog sustava odnosno pojedine aplikacije. (Čerić, i dr., 1998)

Spiralni model oblik je evolucijskog modela u kojem se razvoj odvija u koracima koji se kao u spirali ponavljaju do konačnog rješenja. Jedan korak može odgovarati jednoj fazi razvoja. (Čerić, i dr., 1998) Svaka faza ima sve apstraktne aktivnosti za jednu vrstu cilja koji želimo postići u projektu tom fazom. Tako prvi krug može označavati izradbu studije izvodivosti, drugi krug može biti definicija korisničkih zahtjeva, treći krug može biti dizajn sustava, četvrti implementacija, peti uvođenje i integracija, a šesti održavanje. (Pavlič, 2011)

2. Metodologije

Metodologija propisuje postupak analize sustava, modeliranja i implementacije računalom podržanog informacijskog sustava. Ona sadrži zadaće, tehnike i alate koji se primjenjuju u životnom ciklusu razvoja informacijskih sustava.

Metodologija detaljnije opisuje faze razvoja i aktivnosti pojedine faze na najnižoj potrebnoj razini detalja. Ona opisuje veze između pojedinih aktivnosti i propisuje slijed izvođenja aktivnosti, određuje kada započeti i kada završiti aktivnosti te koje su informacije ulazi u aktivnosti, a koji su rezultat rada aktivnosti. Za svaku je aktivnost određena metoda kojom se aktivnost izvodi i opis ulazno-izlazne dokumentacije. (Pavlič, 2011)

Metodologija osigurava (Varga & Ćurko, 2014):

- komunikaciju između sudionika uključenih u razvoj informacijskog sustava

- skup metoda i tehnika koji će omogućiti da se zadaci izvršavaju na standardni način i provjerenim načinima rada
- učinkovit nadzor sa ciljem uočavanja pogrešaka u ranim fazama
- elastične promjene poslovanja i tehnologije (npr. odvajanjem analize i oblikovanja)
- razvojnu strategiju kojom će se ukloniti ad hoc rješavanje problema
- da se dovoljno pažnje posveti analizi poslovanja

Metodologije su napravljene sa jasnih ciljem da se poslovni zahtjevi postepeno razrađuju kroz niz razina apstrakcije, da se funkcionalnosti informacijskog sustava dodaju, razrađuju, detaljiziraju i programiraju u nizu faza, zadataka, aktivnosti, iteracija, no metodologije ne propisuju kako se informacijski sustav modelira korištenjem vizualnog jezika kao UML. (Krljeza, 2014)

Metodologija odabire metode, prilagođuje ih konačnom cilju, propisuje redoslijed upotrebe metoda, propisuje proces modeliranja od početka do kraja životnog ciklusa informacijskog sustava. Metoda služi kao pojam za intelektualni alat za jednu ili više aktivnosti u okviru metodologije. Metoda je definirani postupak djelovanja za postizanje određenog cilja na polju projektiranja i izgradnje informacijskog sustava. Neke metodologije imaju i više od deset različitih metoda. Jedna metoda može imati podmetode. Metode dovode do modela. Metoda ima mnogo više nego metodologija. Modela ima gotovo onoliko koliko ima informacijskih sustava, odnosno svaki je informacijski sustav originalan. Projektiranje (dizajn) informacijskih sustava jest nalaženje modela informacijskog sustava, koji se sastoji od najmanje triju modela: modela procesa, modela podataka, modela resursa. S vremenom se povećava broj postojećih metodologija. Nekoliko je uzroka povećanju broja metodologija: ne postoji jedan jedinstven put izgradnje informacijskog sustava, postoji više dobrih rješenja, metodologije su slične, ali imaju više marketinških imena, nastaju kao dobra praksa unutar pojedine organizacije, ovise o raznim elementima (tipu organizacijskog sustava, ciljnom proizvodu, uporabljenim alatima i IT-u, namjeni i fazi koju prekrivaju). Zato nije dovoljna jedna metodologija za sve slučajeve. (Pavlić, 2011)

2.1. Vrste metodologija

Izgradnju informacijskog sustava treba organizirati kroz projekt u kojem zajednički sudjeluju korisnici i informatičari. Skupovi metoda koji se koriste pri izgradnji IS-a su inženjerstvo sustava, informacijsko inženjerstvo i programsko (softversko) inženjerstvo. (e-obrazovanje, 2014)

Cilj razvoja informacijskog sustava možemo obuhvatiti sljedećim zahtjevima (Panian, i dr., 2010):

- informacijski sustav treba odgovarati zahtjevima korisnika i zadovoljavati poslovne ciljeve,
- informacijski sustav treba biti pouzdan unutar zadanih granica,
- informacijski sustav treba biti cijenom prihvatljiv.

Postoji više različitih klasifikacija metodologija. Neki autori razlikuju tri osnovne skupine metodologija: tradicionalna (posvećena konceptu procesa), informacijsko inženjerstvo (posvećeno analizi podataka) i objektno orijentirana (posvećena objektima i njihovim operacijama). (Pavlič, 2011)

Opća metodologija razvoja informacijskih sustava znanstvena je disciplina i grana metodologije koja svoja istraživanja koncentrira na polje metoda i metodologije u užem smislu vezano uz informacijske sustave. (Pavlič, 2011)

Specijalizirana metodologija razvoja informacijskih sustava cjelokupnost je načela, pravila, preporučeni niz koraka i procedura, metoda i tehnika koje se primjenjuju u postizanju cilja projektiranja, izgradnje i održavanja informacijskog sustava. Specijalizirana metodologija je skup svih načina kako se svrhovito može obaviti neki posao. Danas su takve metodologije „komercijalni“ proizvodi. Njihovi su autori tvrtke i pojedinci koji su svojim radom unaprijedili proces izvedbe informacijskog sustava. Kupnjom metodologije kupac dobiva priručnike u kojima se propisuje postupak gradnje informacijskog sustava. Školovanje ljudi za primjenu metodologije je nužno. Usvajanjem jedne metodologije podiže se kvaliteta izgradnje informacijskog sustava. Uvođenje specijalizirane metodologije dugoročan je i složen projekt. Uvodi se timskim radom, školovanjem i nadzorom primjene na odabranim projektima. Intelktualni je alat i ne može se kupiti kao uobičajeni proizvod. (Pavlič, 2011)

2.1.1. Zadaci specijalizirane metodologije

Specijalizirana metodologija pokriva sve faze razvoja projekta i osigurava kvalitetu procesa razvoja u svim fazama. Zapravo, metodologija najprije dijeli proces razvoja informacijskog sustava u faze. Specijalizirane metodologije detaljnije opisuju faze razvoja navodeći aktivnosti pojedine faze na najnižoj potrebnoj razini detalja. Prema tome, faza je agregacija aktivnosti. Veća se aktivnost može sastojati od manjih i ta se raščlamba provodi na više razina, sve dok se ne ocijeni da će biti moguće korištenje metodologijom i da je definiran

dovoljan fond informacija za praktičnu uporabu metodologije. Specijalizirana metodologija opisuje veze između pojedinih aktivnosti i propisuje slijed izvođenja aktivnosti.

Osnovni je zadatak specijalizirane metodologije projektiranja informacijskog sustava postaviti recept koji će posao projektiranja informacijskog sustava učiniti što je moguće više formaliziranim. Metodologija je stoga generalni model vođenja projekta koji se u svakom konkretnom projektu uzima kao početni uzor i mijenja prema konkretnim mogućnostima i potrebama projekta. (Pavlič, 2011)

Cilj specijalizirane metodologije i metode razvoja informacijskog sustava jest izgraditi i uvesti uspješan sustav prezentacije činjenica o sustavu čiji se uspjeh može pripisati izvršenju ciljeva sustava i produktivnosti, odnosno sustav treba zadovoljiti utvrđene zahtjeve, biti razvijen na vrijeme i unutar planiranih financijskih sredstava. Da bi se otkrio nedostatak jedne ili više metoda promatrane specijalizirane metodologije u postizanju navedenih ciljeva, treba istraživati zapreke postizanja kvalitete i produktivnosti. (Pavlič, 2011)

2.1.2. CASE alati

Pri projektiranju informacijskog sustava na raspolaganju su nam neki alati koji softverski podržavaju cijeli životni ciklus ili njegove dijelove. Poznati su kao CASE (Computer-Aided Software Engineering – računalom podržano softversko inženjerstvo). Njihovom se primjenom automatiziraju i pojednostavnjuju neke aktivnosti izrade aplikacijskog softvera. Postoje dvije razine alata za automatizaciju izrade softvera za koje su uobičajeni nazivi gornji (upper) i donji (lower) CASE alati. (Srića, i dr., 1999)

Gornji CASE alati podržavaju ranije faze i korake životnog ciklusa izgradnje informacijskog sustava, a bave se analizom postojećeg sustava i izradom dijagrama toka podataka (DFD – Data Flow Diagram).

Donji CASE alati softverski podržavaju kasnije faze izgradnje sustava, a glavni im je cilj pokušati automatski pretvoriti specifikacije novog sustava u računalni program (tzv. generator aplikacije ili generator programa). (Srića, i dr., 1999)

2.2. Metodologija SSADM

SSADM (Structured Systems Analysis Design Method Version) je jedna od poznatijih komercijalno dostupnih specijaliziranih metodologija. SSADM pruža sustavni pristup analizi i dizajnu IT aplikacija. Razvoj metodologije je započeo 1981. godine a razvili su je britanski konzultanti Learmouth and Burchett Management Systems (LBMS) i Central Computing and Telecommunications Agency (CCTA). U lipnju 1990. razvijena je verzija 4 metodologije

SSADM. (Pavlič, Informacijski sustavi, 2011) Nakon prve verzije u metodologiju SSADM unesene su mnoge promjene, a 1996. izdana je zadnja verzija SSADM 4 + verzija 4.3.

SSADM je metodologija razvoja informacijskih sustava preporučena za korištenje za potrebe britanske vlade. Tijekom vremena se proširila i upotrebljava se u 40% slučajeva.

Cilj ove metodologije je da omogući timu koji razvija informacijski sustav da precizno analizira potrebe informacijskog sustava kako bi podržavao informacijsku strategiju organizacije te dizajnirao i specificirao informacijski sustav koji bi najjeftinije ispunio potrebe organizacije. (Tudor & Tudor, 1997)

Osnovna ideja SSADM metodologije je da sustav pripada korisniku, stoga je uloga korisnika u razvoju informacijskog sustava esencijalna. Konzultiran je prilikom prikupljanja činjenica, potvrde potpunosti, ispravnosti i razumljivosti produkata u raznim stadijima razvoja SSADM. Neke je produkte moguće razvijati samo uz aktivnu suradnju korisnika, a u ostalim su zaključci i odluke menadžmenta korisnika osnova za daljnji razvoj informacijskog sustava, jer sljedeća faza može nastupiti tek nakon što je korisnik potvrdio prethodnu fazu. (Pavlič, 2011) Visoko je strukturirana metodologija s detaljnim pravilima i smjernicama za primjenu. Uspjeh metodologije očituje se i u standardima koje omogućuje. Dokumentacija pokriva sva područja projekta razvoja informacijskog sustava. (Pavlič, 2011)

Metodologija usvaja tri perspektive koje se međusobno provjeravaju kako bi se greške primijetile u ranom stadiju razvoja sustava.

Tri perspektive su usmjerene na: funkcije oblikovanjem toka procesa, na događaje oblikovanjem događaja/entiteta kroz povijest života entiteta (Entity Life Histories) i dijagrame podudarnosti učinka (Effect Correspondance Diagrams) te na podatke logičkim oblikovanjem podataka. (Tudor & Tudor, 1997)

SSADM verzija 4 sadrži pet modula koji čine glavnu srž metodologije. (Tudor & Tudor, 1997) Modularnost je uvedena kako bi omogućila projektnom menadžmentu jasno definiranje produkta na kraju svakog modula. (Pavlič, 2011) Moduli su podijeljeni u faze, a faze u daljnje aktivnosti koje se detaljnije razrađuju.

2.2.1. Životni ciklus metodologije SSADM

Faze životnog ciklusa verzije 4 metodologije SSADM su (Pavlič, 2011):

1. Modul studije izvodivosti

- 1.1. Izvodljivost

2. Modul analize potreba

- 2.1. Ispitivanje trenutačne okoline

- 2.2.Prijedlog poslovnih opcija
- 3. Modul specifikacije potreba
 - 3.1.Definiranje potreba
- 4. Modul logičke specifikacije sustava
 - 4.1.Prijedlog tehničkih opcija sustava
 - 4.2.Logički dizajn
- 5. Modul fizičkog dizajna
 - 5.1.Fizički dizajn

Iako okvir metodologije slijedi linearan tijek od postojećeg fizičkog sustava, kroz postojeći i zahtijevani logički model do zahtijevanog fizičkog sustava, nije neophodno da aktivnosti slijede navedeni linearni slijed. (Tudor & Tudor, 1997)

2.2.1.1. Modul izvodivosti

Kao odgovor na dokument o inicijaciji projekta (Project Initiation Document) može se pokrenuti studija izvodivosti. Dokument o inicijaciji projekta sastavljen je od strane projektnog menadžmenta na osnovu informacija sakupljenih izvan projekta. (Tudor & Tudor, 1997)

S obzirom na prirodu projekta odlučuje se o poduzimanju studije izvodivosti. Za jeftine i nisko rizične projekte ne zahtjeva se provođenje studije izvodivosti. Ipak, visoko rizični ili politički osjetljivi projekti zahtijevaju provođenje studije izvodivosti. Ako se ne provede studija izvodivosti, dokument o inicijaciji projekta prosljeđuje se modulu analize potreba.

Izješće o izvodivosti projekta sadrži: dijagram konteksta, fizički model procesa postojećeg sustava, model podataka, katalog potreba, katalog korisnika, opis postojeće okoline, opis zahtijevane okoline, izvješće s definiranim problemima, prijedlog poslovnih opcija, prijedlog tehničkih opcija, plan akcija. (Tudor & Tudor, 1997)

2.2.1.2. Modul analize potreba

Rezultat analize potreba je prijedlog poslovnih opcija (Business System Options), odabrane poslovne opcije (Selected Business System Options), domena projekta i analize.

Analiza potreba koncentrira se na prikupljanje informacija kako bi se u potpunosti shvatilo poslovno područje koje je predmet analize. Poslovni modeli postojećeg sustava (ručnog i računalnog) se produciraju i identificiraju se budući poslovni zahtjevi. Prijedlog poslovnih opcija predstavlja se korisnicima i sponzorima sustava, s detaljima razmjera predloženog

sustava. Na kraju modula većina odlučuje o opcijama koje će biti daljnje razrađene. (Tudor & Tudor, 1997)

Na početku modula razmatra se dokument o inicijaciji projekta i dokumenti vezani za modul izvodivosti. Sve moguće teškoće i problemi vezano za projekt razmatraju se i rješavaju s timom koji vodi projekt. Razvija se plan projekta i koristi se lista potreba sustava prema predloženim dokumentima koji služe kao ulaz u ovaj modul kako bi se razvio katalog potreba koji služi kao glavni skup potreba za novi sustav. Katalog potreba se dopunjuje novim potrebama koje se otkrivaju u daljnjim procesima. (Tudor & Tudor, 1997)

Katalog korisnika se koristi kako bi se definirali krajnji korisnici, njihove uloge i radna mjesta. Domena i granice sustava su utvrđeni dokumentom o inicijaciji projekta i izvještajem studije izvodivosti. (Tudor & Tudor, 1997)

Logički model podataka (Logical Data Model) sastoji se od modela podataka (Logical Data Structure) i popratne dokumentacije o entitetima, atributima i vezama.

Model procesa (Data Flow Model) se sastoji od više modela procesa i popratne dokumentacije o komponentama modela procesa.

Model procesa zajedno s dijagramom konteksta (ukoliko je izrađen) ažuriraju se u dogovoru s korisnicima i u razmatranju dokumenata postojećeg sustava. Unutar modula analize potreba postojeći fizički modeli procesa razvijaju se u postojeće logičke modele procesa uklanjajući fizička, organizacijska, geografska i povijesna ograničenja predstavljena fizičkim modelima procesa i mijenjajući spremišta podataka entitetima ili grupama entiteta. (Tudor & Tudor, 1997)

Definiranje potreba

Koraci vezani za izgradnju modela procesa i modela podataka odvijaju se paralelno s razmatranjem i definicijom potreba. Ciljevi tehnika za definiranje potreba su: (Tudor & Tudor, 1997)

1. Identificirati potrebe kako bi odgovorile na potrebe korisnika i poslovnog sustava u cjelini
2. Opisati potrebe u mjerljivim pojmovima
3. Osigurati osnovu za buduće odluke vezane za novi sustav
4. Doprinijeti donošenju točne specifikacije potreba
5. Usredotočiti analizu na potrebe budućeg sustava

Prijedlog poslovnih opcija

Druga se faza unutar modula specifikacije potreba bavi donošenjem (do šest) mogućih poslovnih opcija sustava. Opcije opisuju što sustav radi, njegova ograničenja, ulaze i izlaze. Opcije mogu biti opisane riječima, u obliku dijagrama ili u kombinaciji, ovisno o tome što se smatra prigodnim za određenu situaciju. U dogovoru s korisnicima lista opcija je sažeta do dvije ili tri mogućnosti za daljnji detaljan opis i analizu troškova i koristi. Odabrane opcije moraju odgovarati ciljevima i ograničenjima navedenim u dokumentu o inicijaciji projekta. Razlozi za odabir određenih opcija i odbacivanje ostalih se dokumentiraju unutar odabranih poslovnih opcija. (Tudor & Tudor, 1997)

2.2.1.3. Modul specifikacije potreba

Specifikacija potreba predstavlja najsloženiju fazu razvoja informacijskog sustava. U ovom modulu detaljno se definiraju korisnički zahtjevi. Cilj modula specifikacije potreba je izvršiti pregled analize potreba prema odabranim poslovnim opcijama sustava kako bi se dodali detalji o zahtjevima određenih podataka, funkcija i događaja koji trebaju biti predstavljeni u budućem sustavu. Krajnji proizvod je specifikacija zahtjeva (Requirements Specification) koja je dovoljno detaljno sastavljena i služi kao osnova za ugovoreni sporazum s dobavljačem softvera ili omogućava timu za razvoj softvera da nastavi dizajn softvera prema preciznom opisu zahtjeva. (Tudor & Tudor, 1997)

Logički model procesa

Prema odabranim opcijama poslovnog sustava mijenjaju se logički dijagrami modela procesa. Dodaju se poslovni procesi koji su identificirani a uklanjaju oni koji nisu dio odabranih poslovnih opcija. Dodaju se i opisi elementarnih procesa za sve nove procese najniže razine uz popratnu dokumentaciju. (Tudor & Tudor, 1997)

Logički model podataka

Logički model podataka također je izmijenjen prema odabranim poslovnim opcijama u logički model podataka prema zahtjevima sustava uklaňanjem ili dodavanjem entiteta, veza i atributa kako je definirano prema odabranim poslovnim opcijama. Katalog potreba se ažurira tako što se određene potrebe predaju logičkom modelu podataka dok se nove potrebe dopisuju u katalog. (Tudor & Tudor, 1997)

Definicija funkcija

Funkcije su središnje značajke metodologije SSADM (v4). Funkcije predstavljaju različita područja za obradu, definirane od strane korisnika, koje podržavaju zahtjeve sustava. Definicija funkcija nije precizna tehnika, nego rezultat dogovora s korisnicima.

Izrađuju se dijagrami povijest života entiteta (Entity Life History) i dijagrami podudarnosti učinka (Effect Correspondence Diagram) označujući događaje povezane sa svakom funkcijom. Za svaku definiranu funkciju stvara se dijagram i opis strukture ulaza/izlaza.

Identificirani događaji se koriste za modeliranje entiteta/događaja preko matrice u kojoj jednu os prate entiteti a događaji drugu. Matrica je potpuna ukoliko svaki događaj utječe na jedan ili više entiteta i svaki entitet ima događaj koji ga stvara i događaj koji ga uništava. Matrica služi kao ključ razvoja povijesti života entiteta i dijagrama podudarnosti učinka.

Funkcije upita su identificirane u dogovoru s korisnicima i uključene u katalog potreba. Za svaki se upit gradi pristupni put za upit (Enquiry Access Paths). (Tudor & Tudor, 1997)

2.2.1.4. Modul logičke specifikacije sustava

Modul logičke specifikacije sustava se sastoji od dvije paralelne faze: prijedloga tehničkih opcija sustava (Technical System options) i logičkog dizajna.

Prijedlog tehničkih opcija sustava proces je sličan prijedlogu poslovnih opcija sustava, ali je usredotočen na to kako je sustav implementiran, a ne na to što će sustav raditi. Identificira se do šest tehničkih opcija sustava. Opcije su vezane za hardver, softver i način na koji tehničke opcije odgovaraju specifikaciji zahtjeva te na njihov utjecaj na obuku korisnika, osoblje i organizaciju kao i na analizu koristi i troškova. Potencijalne tehničke opcije predstavljaju se korisnicima kako bi se među njima odabrale opcije koje će dalje biti obrađene.

Prvi korak unutar faze logičkog dizajna je definiranje korisničkih dijaloga. Funkcije ažuriranja i funkcije upita se odvojeno razmatraju i izrađuje se model procesa ažuriranja (Update Process Model) i model procesa upita (Enquiry Process Model). Modeli se dalje razmatraju tijekom modula fizičkog dizajna. (Tudor & Tudor, 1997)

2.2.1.5. Modul fizičkog dizajna

Svrha modula fizičkog dizajna je odrediti fizičke podatke, procese, ulaze i izlaze koristeći jezik i značajke odabrane fizičke okoline, uključujući standarde za instalaciju. Na kraju ovog modula sve bi trebalo biti točno na mjestu određujući kako će se aplikacija konstruirati. Integritet logičkog dizajna je sačuvan koliko je to moguće.

Prve aktivnosti unutar modula fizičkog dizajna bave se planiranjem pristupa fizičkom dizajnu, istraživanjem i dokumentiranjem značajki i zahtjeva ciljanog hardvera i popratnog softvera. Strategija fizičkog dizajna uvelike je određena tijekom opisa tehničke okoline (Technical Environment Description) koja je sastavljena tijekom faze odabira tehničkih opcija sustava. Na osnovu odabranih opcija se određuje stil za korisničko sučelje (The Application Style Guide). (Tudor & Tudor, 1997)

Tijekom ovog modula model logičkih podataka izmjenjuje se u prvi dizajn fizičkih podataka. Detaljno se razrađuje plan implementacije funkcija i stvara plan implementacije komponenti funkcija (Function Component Implementation Map) koji određuje kako će se svi logički procesi fizički implementirati. Početni dizajn se optimizira dok se ne postigne zadovoljavajući dizajn. Kompletira se specifikacija funkcija. Produkti modula fizičkog dizajna konačno se skupljaju za odluku menadžmenta, podređuju se dogovoru s korisnicima i predaju se timu za izgradnju informacijskog sustava. (Tudor & Tudor, 1997)

2.3. Metodologija MIRIS

Specijalizirana metodologija MIRIS (Metodologija za razvoj informacijskog sustava) skup je metoda i uputa čiji je ukupni cilj projektirati i izgraditi informacijski sustav. Objavljena je 1995. godine. Propisuje faze razvoja i aktivnosti pojedine faze do potrebne razine detalja informacijskog sustava.

Osnovna hipoteza oko koje je MIRIS oblikovan glasi: Životni ciklus projektiranja podijeliti u tri faze. U prvoj fazi apstraktno modelirati cijeli sustav i u tom modelu sustav podijeliti u podsustave. U drugoj fazi modelirati podsustav i propisati odgovarajuću metodu za modeliranje procesa. U trećoj fazi modelirati podatke relevantnih procesa i definirati arhitekturu aplikacije prema modelu procesa i modelu podataka. (Pavlič, 2011)

2.3.1. Životni ciklus metodologije MIRIS

Predloženi životni ciklus metodologije MIRIS je „V“ model i to tako da su na lijevoj strani faze kojima se dekomponira poslovni sustav (dok se kod uobičajenog „V“ modela dekomponira proces modeliranja sustava), a na desnoj strani predlaže linearan razvoj i vođenje aplikacija neovisno o složenosti (dok se kod uobičajenog „V“ modela izvodi testiranje i okrupnjavanje modula u veće proizvode).

Faze životnog ciklusa grupirane su u dvije skupine: logičko oblikovanje (projektiranje informacijskog sustava) i fizičko oblikovanje (izgradnja informacijskog sustava). Svaka skupina ima tri faze koje se dalje dijele u aktivnosti. (Pavlič, 2011)

2.3.1.1. Logičko oblikovanje (projektiranje)

1. Strateško planiranje informacijskog sustava (SP)

1.1. Analiza: definiranje i poduka tima, dekompozicija procesa, popis dokumentacije i kretanje kroz sustav

1.2. Podsustavi: određivanje podsustava i veza

1.3. Prioriteti: određivanje prioriteta

1.4. Resursi: definicija cjelovite infrastrukture

1.5. Plan: planiranje glavnih projekata i aktivnosti

2. Glavni projekt (GP)

2.1. PZ: izradba projektnog zadatka

2.2. Intervjuiranje, raščlanjene i modeliranje procesa (DTP)

2.3. Procesi GP: analiza procesa, problema i prijedloga poboljšanja

2.4. Podaci GP: opisivanje podataka

2.5. Plan GP: planiranje izvedbenih projekata

2.6. Resursi GP: definiranje modela resursa glavnog projekta

3. Izvedbeni projekt (IP)

3.1. DEV: Intervjuiranje, apstrakcija i modeliranje podataka (EV)

3.2. Prevođenje: prevođenje modela podataka u shemu BP (RM)

3.3. Arhitektura IP: definiranje arhitekture programskog proizvoda (APP)

3.4. Operacije IP: projektiranje operacija nad shemom BP

2.3.1.2. Fizičko oblikovanje (izgradnja IS)

4. Proizvodnja softvera (PS)

4.1. Planiranje proizvodnje

4.1.1. Planiranje aktivnosti proizvodnje SW

4.1.2. Određivanje izvršitelja za pojedine zadatke i određivanje rokova

4.1.3. Određivanje i kreiranje produkcijske, testne i razvojne okoline

4.2. Oblikovanje baze podataka

4.2.1. Prevođenje logičkog modela podataka u fizički model sheme baze podataka

4.2.2. Kreiranje razvojne okoline za svakog pojedinog programera

4.2.3. Punjenje sheme baze podataka u razvojnoj okolini iz postojeće produkcijske BP

4.2.4. Dodavanje novih koncepata iz modela podataka u razvojnu shemu BP (tip entiteta i dr.)

4.2.5. Kreacija razvojne baze podataka

4.2.6. Inicijalno punjenje testne baze podataka

4.3. Razvoj programskih proizvoda

4.3.1. Izvedba glavnog izbornika (aplikacijskog stabla) ili dorada već postojećeg

4.3.2. Izradba ekrana za pregled redaka svake tablice po jednom ili više ključeva

4.3.3. Izradba ekrana za operacije nad jednim retkom tablice (unos, izmjena, brisanje i pregled)

4.3.4. Izradba programskih modula različitih vrsta i namjena: obračuna, procedura, funkcija, kontrola, look-upova nad tablicama (s prvim resetiranjem modula)

4.3.5. Izradba izvještaja (s prvim resetiranjem modula)

4.4. Testiranje u testnoj okolini

4.4.1. Prijenos razvijenih programskih modula u testno okruženje

4.4.2. Spajanje novih modula s postojećim

4.4.3. Back up verzija softvera

4.4.4. Testiranje prototipa softvera nad testnom bazom podataka

4.4.5. Ažuriranje planova proizvodnje softvera

4.4.6. Izvođenje prema potrebi aktivnosti iz ranijih skupina aktivnosti i ponovno testiranje

4.5. Testiranje i ispravljanje u radnoj okolini

4.5.1. Prijenos razvijenih programskih modula u radno okruženje

4.5.2. Spajanje novih modula s postojećima

4.5.3. Back up verzija softvera

4.5.4. Punjenje baze podataka

4.5.5. Testiranje prototipa softvera nad produkcijskom bazom podataka koje provodi programer

4.5.6. Ažuriranje planova proizvodnje softvera

4.5.7. Izvođenje prema potrebi aktivnosti iz ranijih skupina aktivnosti i ponovno testiranje

4.6. Testiranje koje provodi korisnik

4.6.1. Prezentacija softvera korisniku

- 4.6.2. Testiranje koje provodi korisnik
- 4.6.3. Izvedba popisa primjedbi korisnika
- 4.6.4. Ažuriranje planova proizvodnje softvera
- 4.6.5. Izvođenje prema potrebi aktivnosti iz ranijih skupina aktivnosti i ponovno testiranje
- 4.6.6. Izradba zapisnika o testiranju i prihvaćanju faze uvođenja

5. Uvođenje (UVO)

- 5.1. Instalacija gotovog softvera na produkcijsko okruženje (kod korisnika)
- 5.2. Izradba uputa
- 5.3. Prezentacija gotovog softvera
- 5.4. Poduka
- 5.5. Završne konverzije
- 5.6. Završno testiranje
- 5.7. Početak primjene nove aplikacije
- 5.8. Uspostava novog sustava i potpisivanje primopredajnog zapisnika

6. Primjena i održavanje (ODR)

- 6.1. Podešavanje novog aplikacijskog sustava
- 6.2. Izvješće o procjeni novog projekta
- 6.3. Raspodjela odgovornosti korisnika i programera
- 6.4. Korištenje aplikacijskim sustavom
- 6.5. Postavljanje korisnikovih zahtjeva za izmjenama

Cjelokupni posao započinje izradom strateškog plana, potom se u fazi Glavnog projekta analizira poslovanje i kreira model procesa.

U fazi Izvedbenog projekta modeliraju se podaci i definira logička arhitektura programskog proizvoda.

Faza Proizvodnja softvera započinje planiranjem proizvodnje. Uzimaju se u obzir postojeći programski proizvodi i baza podataka u sustavu. Cilj je reorganizirati postojeću bazu podataka (stvoriti novu ako ništa ne postoji) i kreirati potreban novi programski proizvod, odnosno izmijenjen ili zamijeniti postojeći programski proizvod.

Gotov programski proizvod uvodi se u sustav, a po uvođenju primjenjuje se u svakodnevnom radu. Po potrebi se dijelovi informacijskog sustava održavaju kako bi on zadovoljio potrebe korisnika.

MIRIS se koristi trima osnovnim metodama: metodom za modeliranje podataka, metodom za modeliranje procesa i metodom za modeliranje aplikacija. Metode su slične mnogobrojnim metodama u drugim metodologijama. (Pavlič, 2011)

Modeliranje procesa

Model procesa definira kako se obrađuju, prikupljaju i distribuiraju podaci informacijskog sustava. Model procesa opisuje dinamiku podataka informacijskog sustava. On opisuje skup procesa, odnosno funkcija kojima se mijenjaju podaci informacijskog sustava. U implementiranom informacijskom sustavu modeli procesa se realiziraju kroz odgovarajuću programsku podršku (aplikacije). (Tudor M. , 2014)

Proces je skup povezanih aktivnosti i odluka, preko kojih objekti sustava ostvaruju dijelove cilja svog postojanja, a za njihovo izvršenje su potrebni određeni resursi i određeno vrijeme. (Pavlič, 1996)

Poslovna funkcija, odnosno proces, posao je koji treba obaviti u vezi s poslovnim objektom. Funkcije su npr. naručivanje proizvoda, izdavanje računa, otvaranje projekta, ocjenjivanje studenata itd. Informatički gledajući funkcijama se mijenjaju podaci objekata. (Čerić, i dr., 1998)

Cilj analize sustava je modeliranje procesa sustava. Model sustava je pojednostavljena reprezentacija o relevantnim karakteristikama sustava, preko skupa dijelova, veza među dijelovima i atributa dijelova i veza. Od modela procesa se zahtjeva da bude vjerna slika sustava, odnosno, jedan od temeljnih ciljeva je ugradnja što više znanja sa aspekta procesa u model procesa. (Pavlič, 1996)

Dijagram toka podataka tehnika je koja koristi kod modeliranja procesa kako bi se prepoznali koraci u procesu više i niže razine i tokovi podataka u njima. (Panian, i dr., 2010)

Dijagram toka podataka je grafičko sredstvo za prezentaciju procesa sustava. Uveo ga je DeMarco 1978. godine. Nastao je proširenjem tehnike „Ulaz – Proces - Izlaz“ i tehnike HIPO tvrtke IBM. Ideja HIPO metode je promatrati sustav uočivši jedan ili nekoliko ulaza koje koristi jedan određen proces ili skup procesa iste klase te uočiti jedan ili nekoliko izlaza iz tog procesa. (Pavlič, 1996)

Pomoću dijagrama toka podataka korisnik i analitičar precizno definiraju zahtjeve korisnika. Dijagram toka podataka predstavlja jezik za komunikaciju korisnika i analitičara. Dijagram toka podataka sastoji se od (Pavlič, 1996):



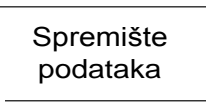

1. Ulaznih i izlaznih tokova podataka (zaslona, dokumenata) koje sustav dobiva ili daje u okružje

2. Vanjskih objekata (organizacija, ljudi, drugih sustava) koji šalju prema ili primaju tokove podataka od sustava
3. Procesu sustava (programa, aktivnosti, radova, podsustava) koji transformiraju ulazne tokove u izlazne tokove podataka i
4. Spremišta podataka (baze podataka, kartoteke) u kojima se čuvaju podaci potrebni za izvršenje procesa ili dobiveni kao rezultat rada procesa

Dijagram toka podataka specificira što sustav radi, odnosno što će sustav raditi na logičkoj razini. Pored toga moguće je dijagram toka podataka koristiti za opis fizičke razine sustava (postojećeg i budućeg). Osnovni cilj procesne analize sustava je utvrđivanje logičkog toka podataka kroz sustav. Procesi na dijagramu toka podataka se paralelno izvršavaju u stvarnom sustavu. Ne sadrži opis programske logike niti iz njega može proizaći programska potpora. Prije nego što se dođe do razine opisa logike programa zaustavlja se proces analize i gradnje dijagrama toka podataka. Pomoću njega se jasno i detaljno u grafičkom obliku predstavlja sustav te se on može lako, brzo i u potpunosti shvatiti.

Dijagram toka podataka koristi čovjeku bliske koncepte (elemente grafičkih simbola) za opis sustava. Koncepti su prikazani u tablici 1.

Tablica 1: Koncepti dijagrama toka podataka

KONCEPT	SIMBOL
Tok podataka je skup podataka koji se kreću od jednog dijela sustava k drugom. Tok podatka je vod kojim teku informacije poznate strukture i sadržaja.	
Proces je transformacija strukture i sadržaja ulaznih u izlazne tokove podataka.	
Spremište je memorijsko mjesto gdje se čuvaju podaci za kasnije procese obrade	
Vanjski sustav predstavlja sustave koji su u vezi sa promatranim sustavom bilo da su izvori ili odredište podataka.	

Modeliranje podataka

Model podataka je simbolički prikaz logičkih, vremensko-logičkih i hijerarhijskih odnosa među ljudima, stvarima i događajima u nekome sustavu, iskazanih podacima kao njihovim manifestacijama. Modeliranje podataka je postupak izrade stanovitog modela podataka i obavlja se tijekom procesa razvoja informacijskog sustava. Model podataka je apstraktna reprezentacija podataka. On opisuje skupove podataka i njihovih međusobnih veza. To je strukturirani skup informacija koji opisuje stanja objektnog sustava, odnosno njegovog informacijskog sustava. Modeliranje podataka je proces koji počinje utvrđivanjem i analiziranjem zahtjeva korisnika, a završava izgradnjom stabilne, ali prilagodljive baze podataka. (Vukmirović & Čapko, 2014)

Primjenom procesa modeliranja podataka zasnovanog na nekoj metodi kao rezultat dobiva se model podataka. Model nastaje procesom apstrakcije u kojem se najprije biraju relevantni koncepti koje reprezentacija treba sadržavati, a zatim se svakom konceptu pridružuju relevantna obilježja koja se žele prikazati u okviru modela. Model podataka konstruiraju projektanti informacijskog sustava, a potvrdu njegove ispravnosti daju krajnji korisnici. Model podataka je osnova za razvoj baze podataka i definiranje oblika arhitekture programskog proizvoda. Baza podataka je organizirani skup podataka. (Pavlić, 2011a)

Metoda entiteti-veze (EV) je grafički prikaz međusobno povezanih grupa podataka promatranog sustava. EV je semantički bogata metoda za modeliranje podataka jer raspoloživo ljudima bliskim konceptima. EV se odlikuje prirodnošću opisa, a njezini koncepti su bliski korisniku pa je shema modela podataka laka za razumijevanje i komunikaciju korisnika i projekatana. (Pavlić, 2011a)

Dijagram strukture modela naziva se dijagram entiteta i veza (DEV). DEV je grafički prikaz modela podataka sustava, metodom EV. (Pavlić, 2011a)

Koncepti metode entiteti-veze od kojih se gradi struktura modela entiteti-veze jesu:

- 1 Entiteti i tip entiteta
- 2 Slab tip entiteta
- 3 Veza i tip veze
- 4 Atribut tipa entiteta
- 5 Agregirani tip entiteta


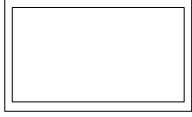
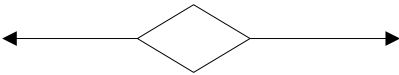

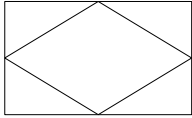
Navedeni koncepti prikazani su u tablici 2.

Entitet je materijalni ili ne materijalni element poslovnog sustava o kojem informacijski sustav treba bilježiti informacije (npr. proizvod, narudžba, račun, radnik, student ili projekt). (Čerić, i dr., 1998)

Ključ tipa entiteta je takav skup atributa koji neovisno o vremenu zadovoljava sljedeće uvijete (Pavlić, 2011a):

- Uvjet jedinstvenosti - ne postoje dva pojedinačna pojavljivanja entiteta u tipu entiteta takva da imaju istu vrijednost atributa koji čine ključ i ne postoje dva tipa entiteta koji imaju isti skup atributa za ključ
- Uvjet neredundantnosti (minimalnosti) – ne postoji nijedan atribut kao dio ključa koji se može izostaviti iz ključa a da se pritom uvjet jedinstvenosti ne gubi.

Tablica 2: Koncepti E-V dijagrama po Chenovoj notaciji

KONCEPT	SIMBOL
Tip entiteta je skup entiteta iste vrste dobiven procesom apstrakcije.	
Slab tip entiteta tip entiteta koji je na neki način ovisan o nekim drugim tipovima entiteta.	
Veza predstavlja odnos među entitetima. Tip veze je skup veza iste vrste.	
Atribut je neko kvalitativno ili kvantitativno svojstvo entiteta	
Agregacija je koncept koji omogućuje promatranje više tipova entiteta i njihovih međusobnih veza kao jedan novi tip entiteta	

Model podataka u kome se podaci o stvarnom sustavu čuvaju u tablicama naziva se relacijski model podataka. Osnovni koncepti za gradnju strukture relacijskog modela podataka su relacije. Relacija u relacijskom modelu podatak je isto što i relacija u matematici s tom razlikom da su relacije u relacijskom modelu podataka vremenski promjenjive. S motrišta korisnika vrijedi definicija relacije: relacija je imenovana tablica, atribut je stupac tablice, n-torka je redak tablice. (Pavlić, 2011a)

3. Praktični rad

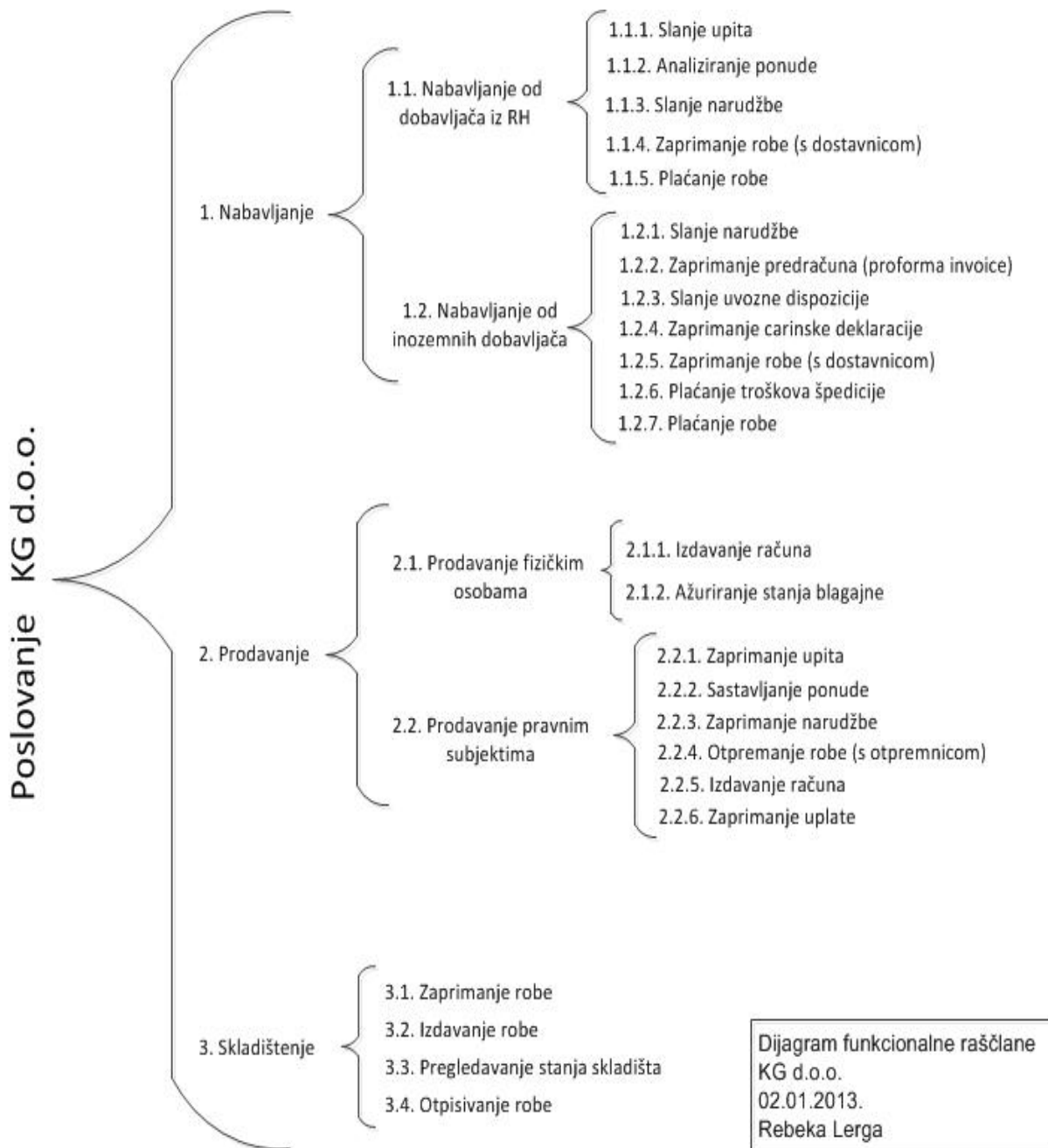
Poduzeće KG d.o.o. je trgovina s opremom za rasvjetu i ostalim proizvodima za kućanstvo. Osnovna djelatnost je nabavljanje robe unutar Republike Hrvatske kao i iz inozemstva, prodaja robe fizičkim osobama i pravnim subjektima unutar Hrvatske te skladištenje robe.

Nabavljanje robe, prodaja i skladištenje prikazani su dijagramom funkcionalne raščlane, dijagramom konteksta te dijagramima tokova podataka.

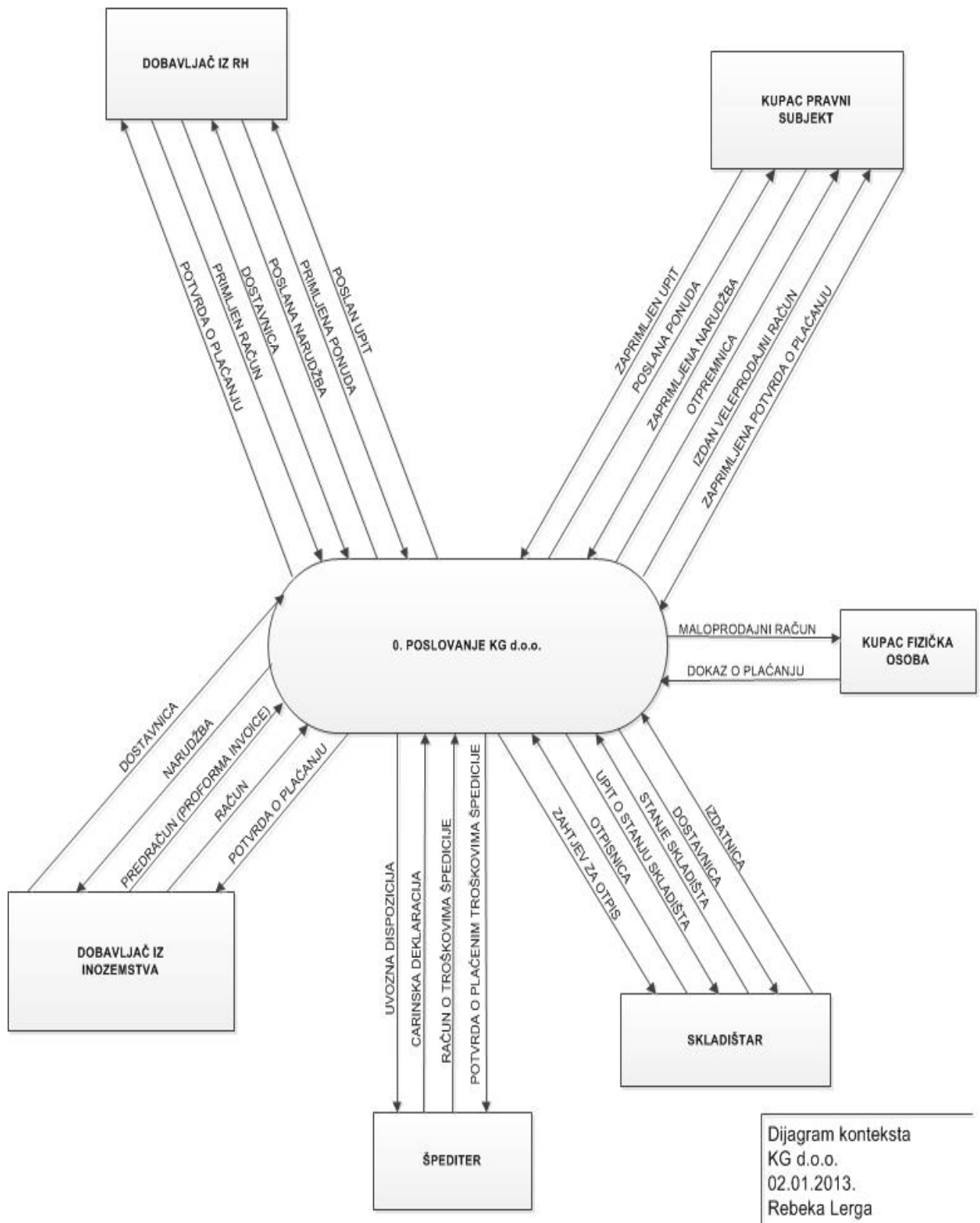
Na slici 1 prikazano je poslovanje poduzeća KG d.o.o. dijagramom funkcionalne raščlane.

Dijagram konteksta prikazan je na slici 2, a dijagram toka podataka na slici 3. Raščlana nižih

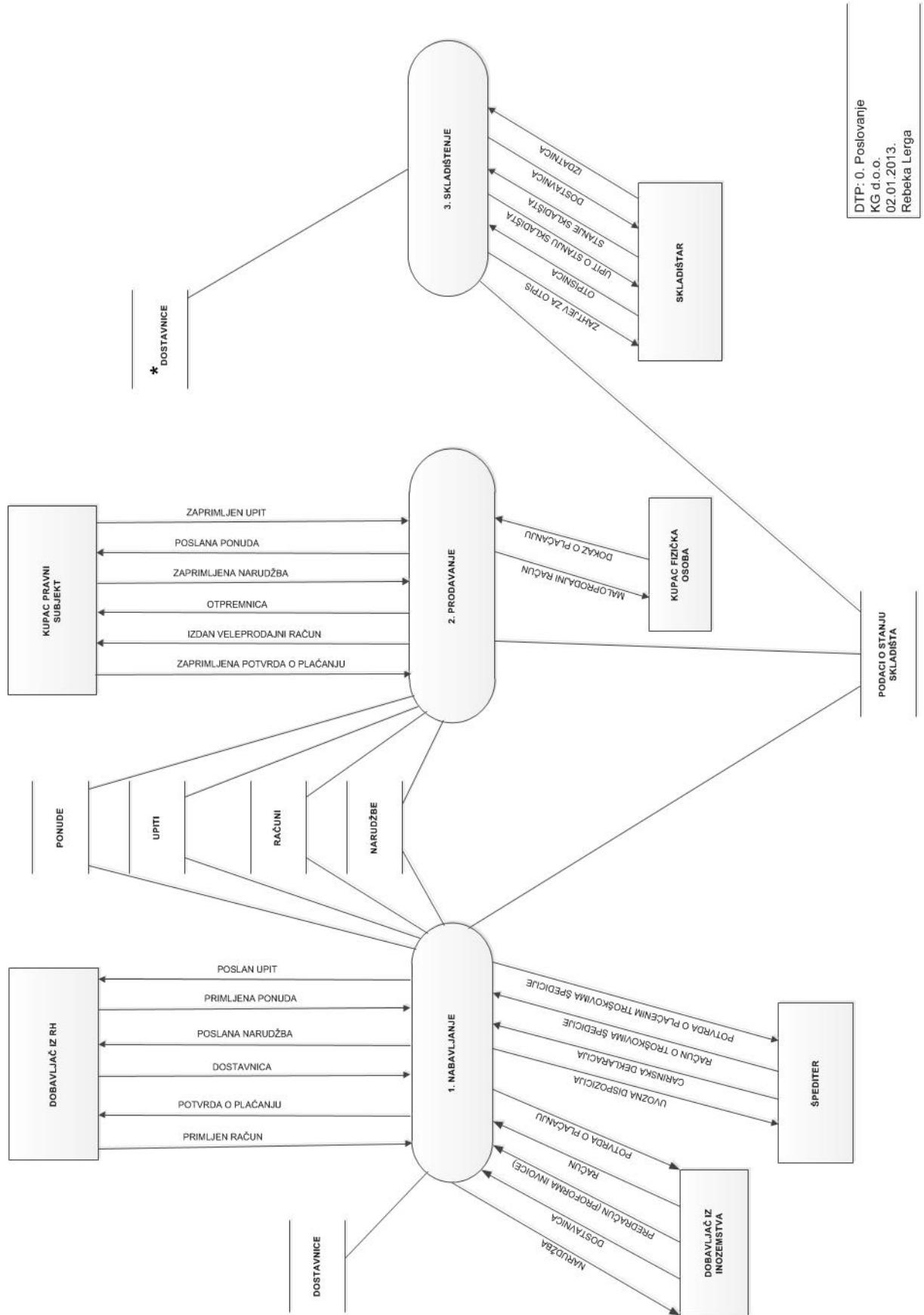
razina prikazana je dijagramima tokova podataka od slike 4 do slike 10.



Slika 1 Dijagram funkcionalne raščlane KG d.o.o.

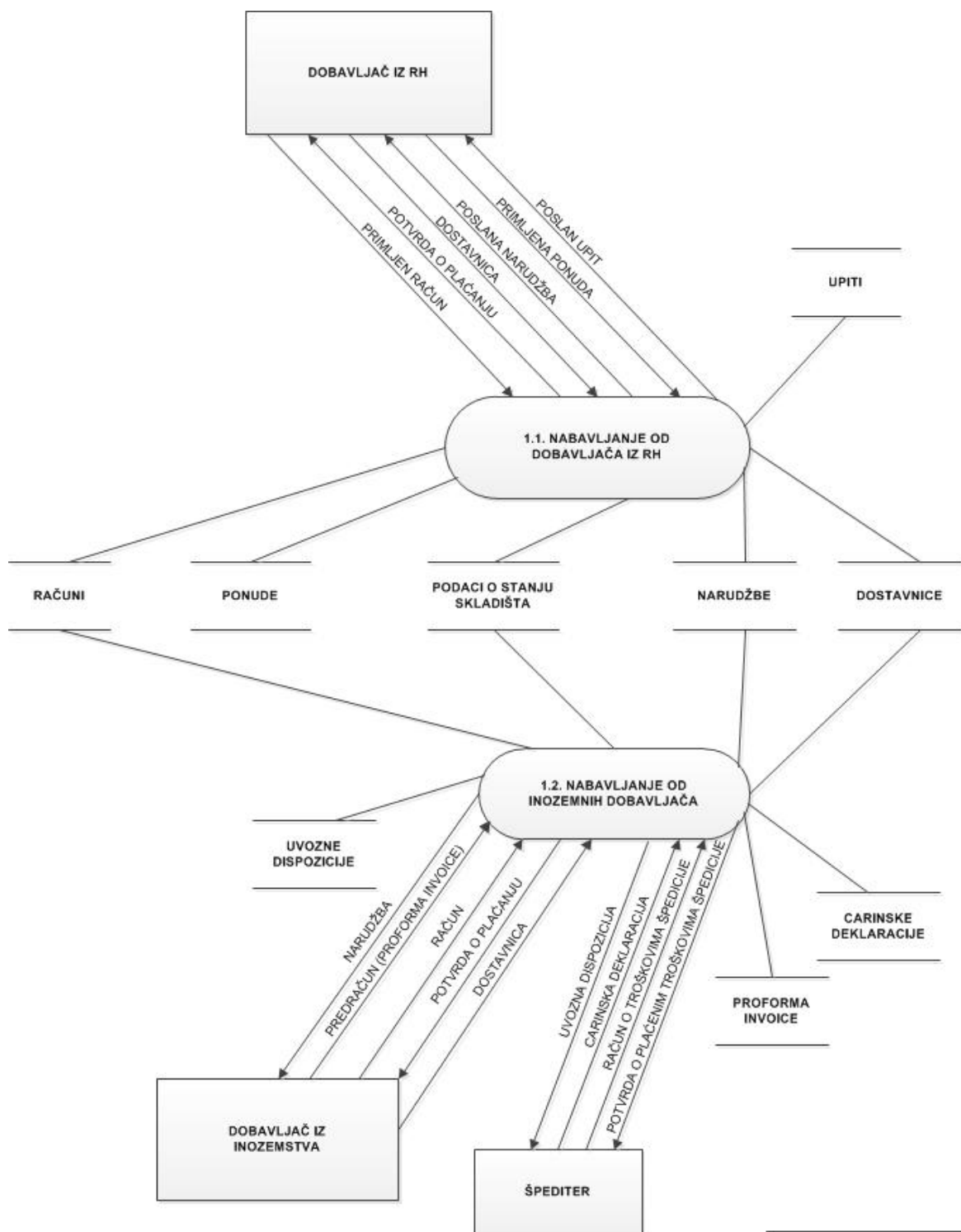


Slika 2 Dijagram konteksta KG d.o.o.



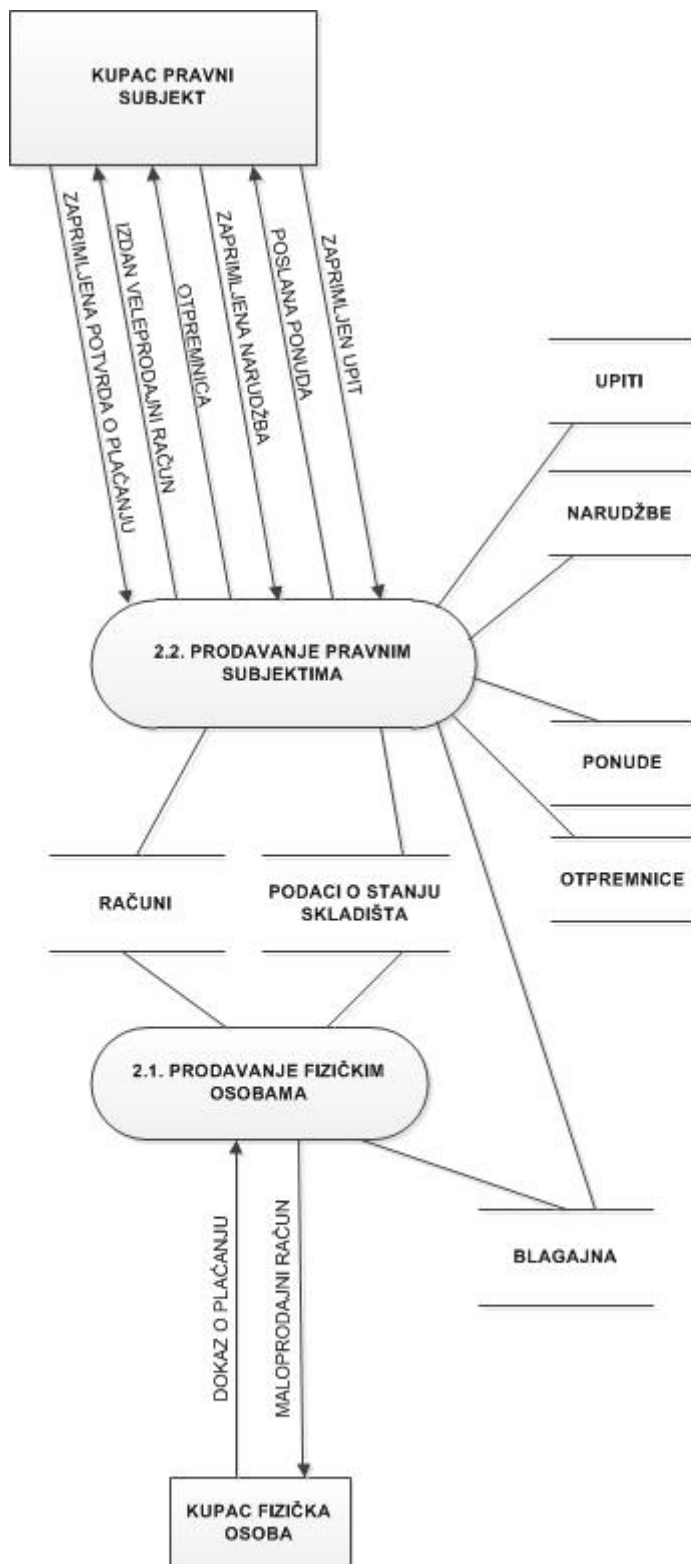
DTP: 0. Poslovanje
 KG d.o.o.
 02.01.2013.
 Rebeka Lerga

Slika 3 DTP: 0. Poslovanje KG d.o.o.



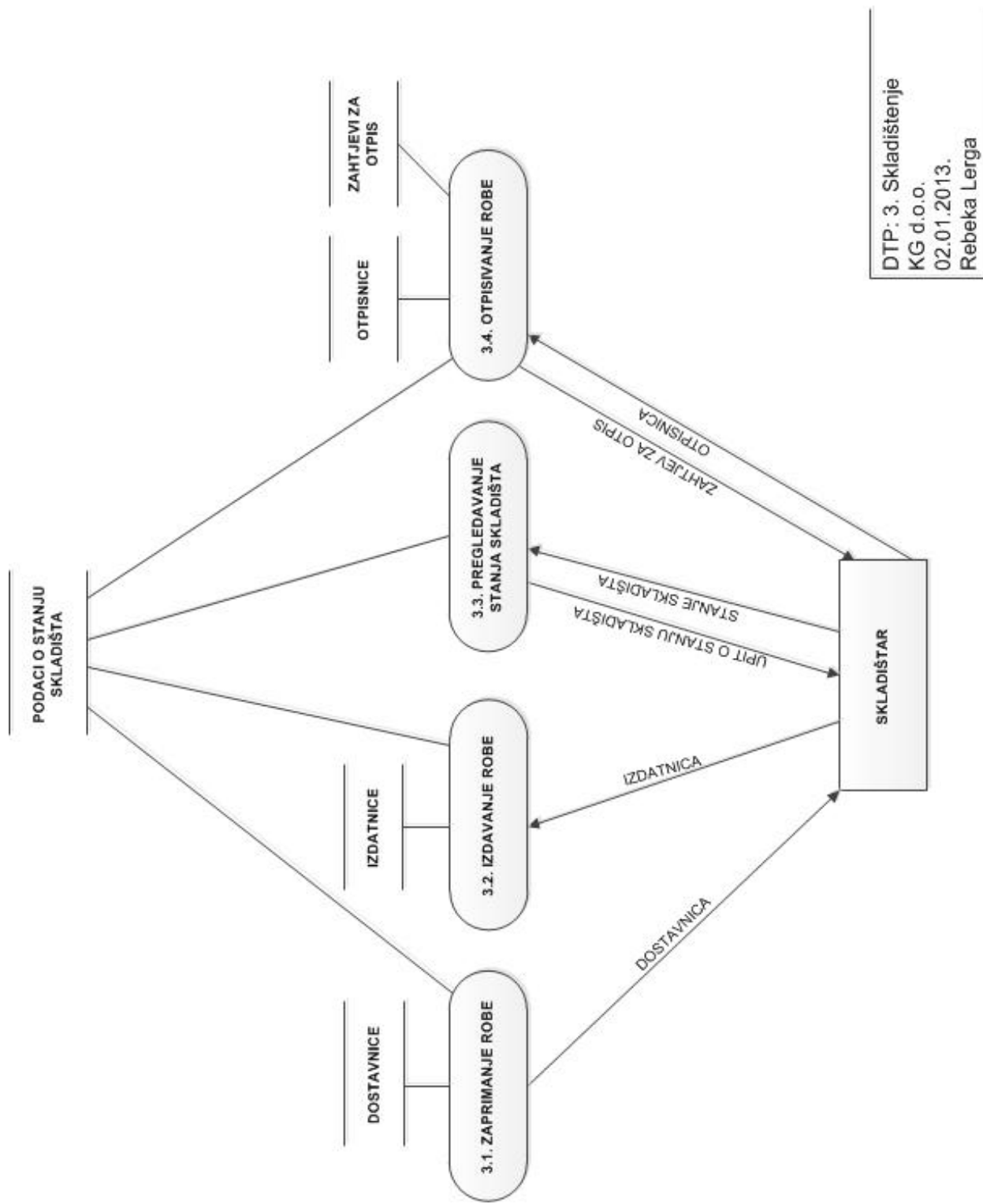
DTP: 1. Nabavljanje
 KG d.o.o.
 02.01.2013.
 Rebeka Lerga

Slika 4 DTP: 1. Nabavljanje KG d.o.o.



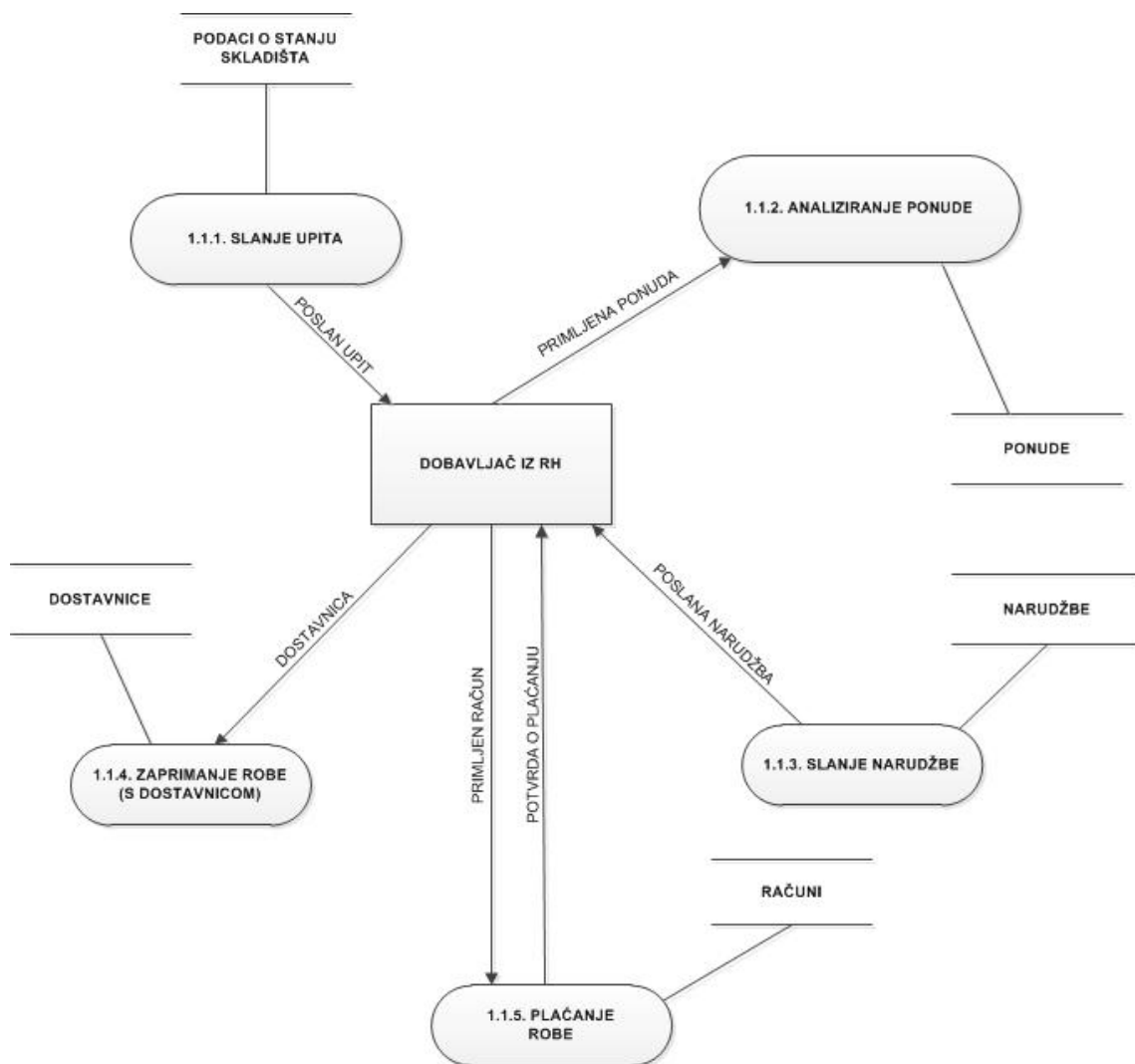
DTP: 2. Prodavanje
 KG d.o.o.
 02.01.2013.
 Rebeka Lerga

Slika 5 DTP: 2. Prodavanje KG d.o.o.



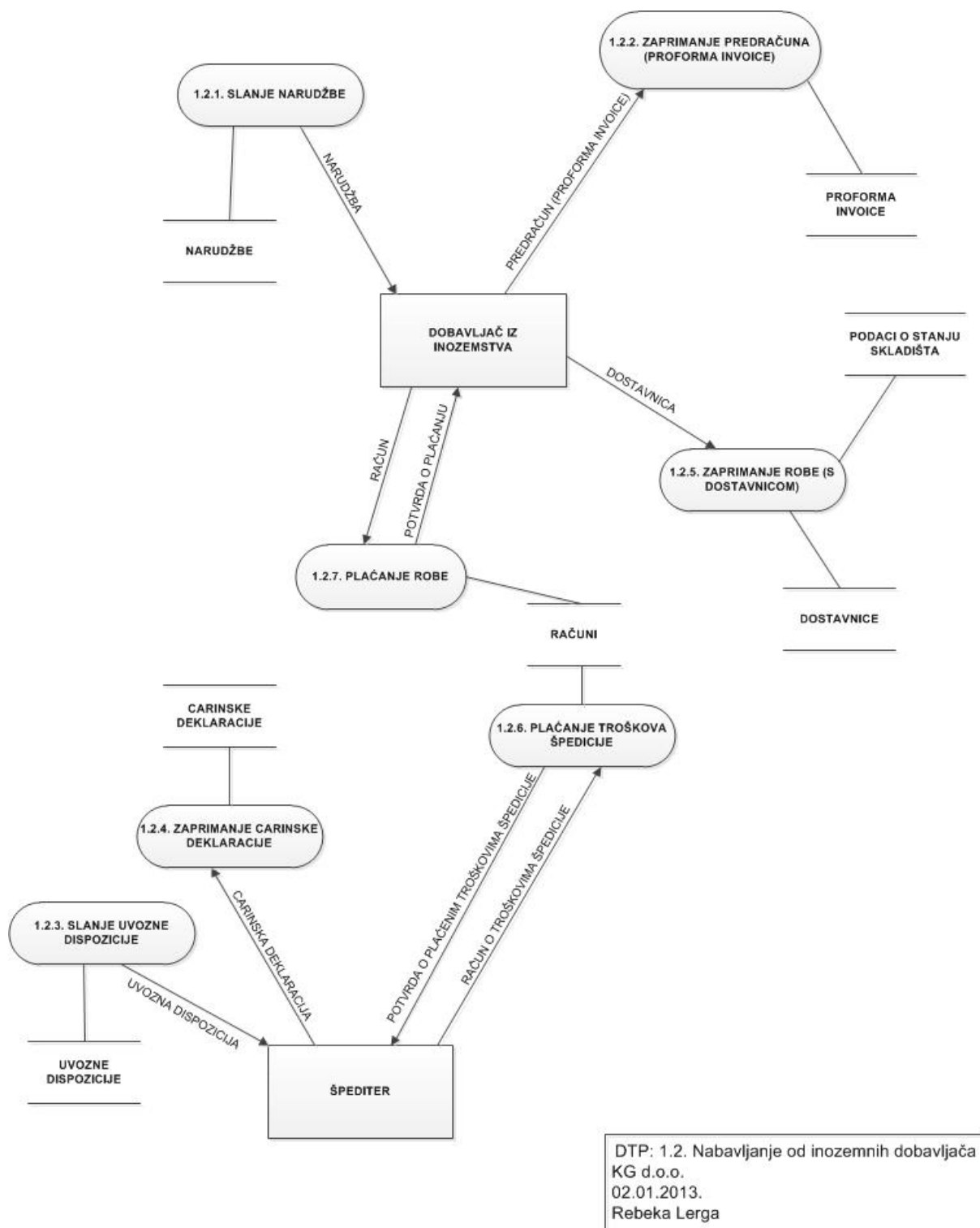
DTP: 3. Skladištenje
 KG d.o.o.
 02.01.2013.
 Rebeka Lerga

Slika 6 DTP: 3. Skladištenje KG d.o.o.

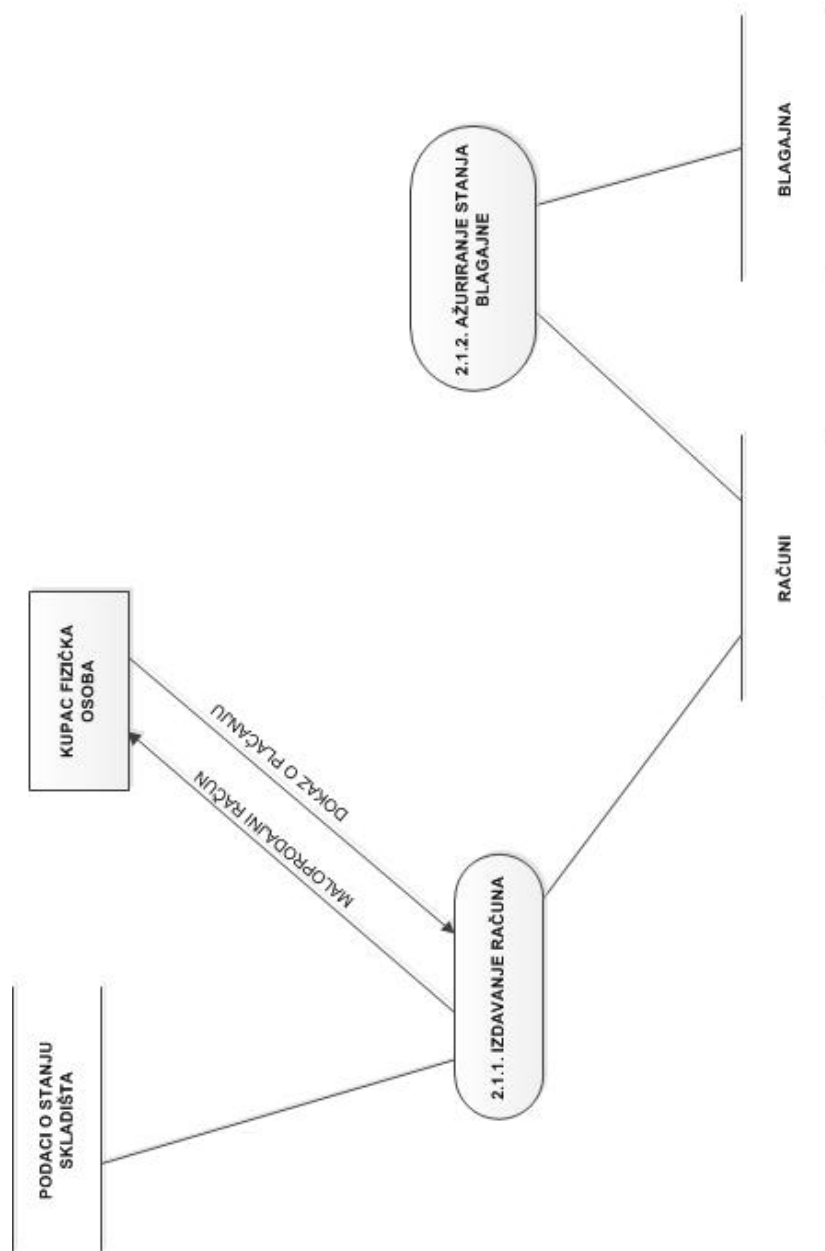


DTP: 1.1. Nabavljanje od dobavljača iz RH
 KG d.o.o.
 02.01.2013.
 Rebeka Lerga

Slika 7 DTP: 1.1 Nabavljanje od dobavljača iz RH KG d.o.o.

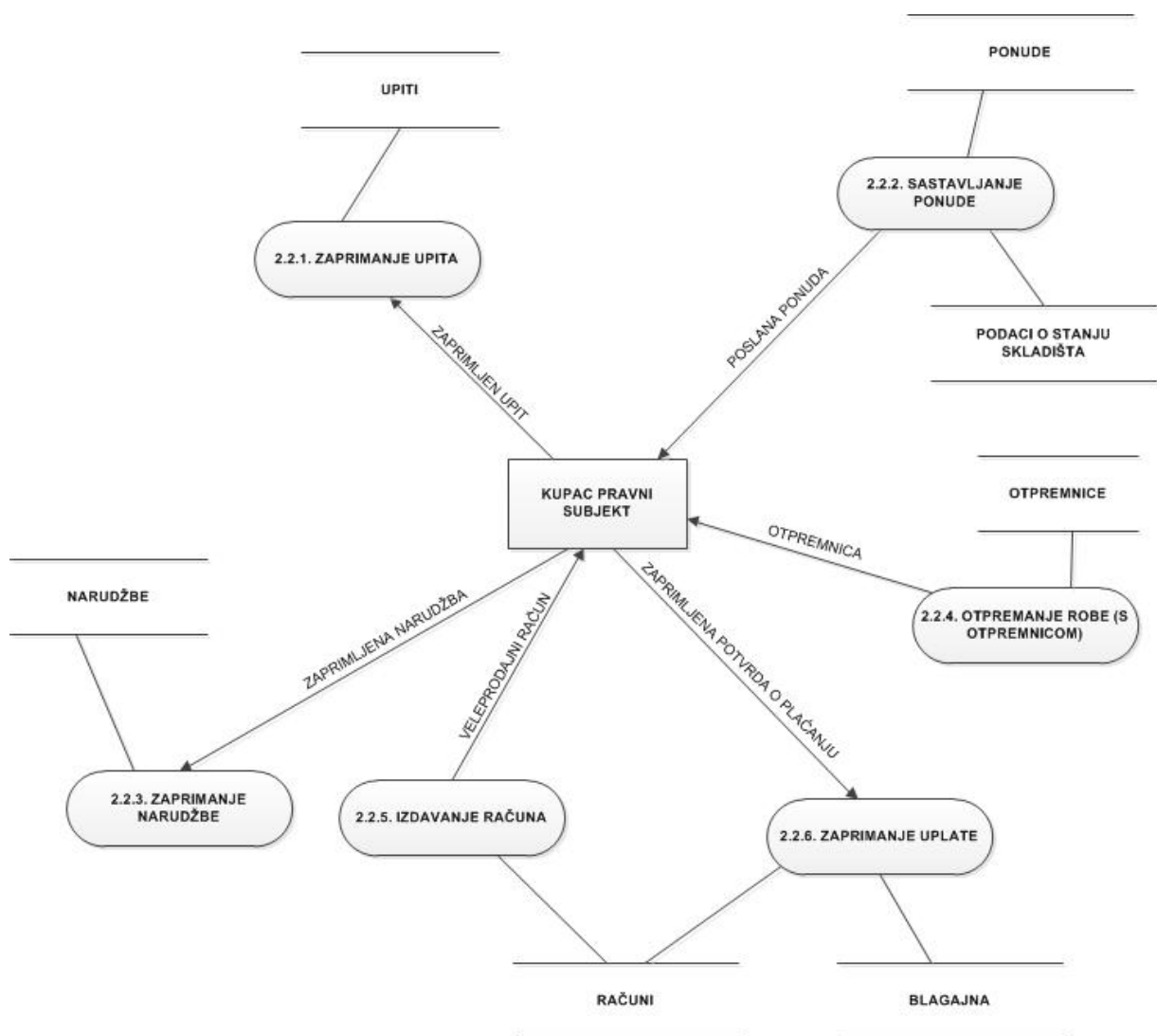


Slika 8 DTP: 1.2. Nabavljanje od inozemnih dobavljača KG d.o.o.



Slika 9 DTP: 2.1. Prodavanje fizičkim osobama KG d.o.o.


DTP: 2.1. Prodavanje fizičkim osobama
 KG d.o.o.
 02.01.2013.
 Rebeka Lerga



DTP: 2.2. Prodavanje pravnim subjektima
 KG d.o.o.
 02.01.2013.
 Rebeka Lerga

Slika 10 DTP: 2.2. Prodavanje pravnim subjektima KG d.o.o.

Dokument na slici 11 je otpremnica poduzeća KG d.o.o. Dokument je opisan u tablici 3, a na slici 12 prikazan je dijagramom entiteta i veza.



ELEKTROMAX d.o.o.
 51 000 Rijeka, Nikole Cara 2
 MB: 0443581, OIB: 91634021778
 Ž.R. 2340009-1117017063 - PRIVREDNA BANKA ZAGREB d.d.
 Prodajna mjesta:
 01 Prodavaonica Rijeka, Nikole Cara 3, tel. 051/322-960
 03 Skladište Marinići, Marinići bb, Viškovo, tel. 051/269-490

792/150

R - 1

Mjesto izdavanja: Marinići
 Datum računa: 6.12.2012
 Datum dospijea: 4.2.2013
 Odgovorna osoba: P. L. L.
 Narudžba kupca broj:

MB Kupca: 023 [redacted]
 OIB Kupca: 34 [redacted]
 Strana: 1

K [redacted] G [redacted] D.O.O.
 [redacted]

Račun / Otpremnica: 12-300-005555

Vezni dokument: 12-010-008024/6.12.2012

Rbr.	Šifra	Naziv	Količina	MJ	Cijena	R %	PDV%	Vrij. bez PDV
1	11654	C48109-1 BLOKADA UTIČNOG MJESTA/10KOM	4,00	PAK	9,29	18,00	25	37,16
2	01874	KAB PP/L 3X0,75 (H03VV-F) BIJELI	200,00	M	2,04	20,00	25	408,00
3	07472	C56112 BRODSKA SVJ. 60W BIJELA	6,00	KOM	15,99	18,00	25	95,94
4	01533	GW60009 IND.UTIKAČ 5P 16A IP44 400V	3,00	KOM	20,03	29,00	25	60,09
Ukupno:								601,19
Popust:								122,99
PDV:								119,55
Za platiti:								HRK 597,75

Slovima: petstodevedesetsedam HRK 75/100

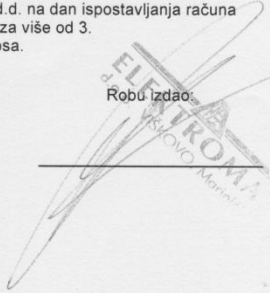
Porezne stope	Osnova	PDV	Vrijednost
Roba.PDV-stopa 25%	478,20	119,55	597,75

-Prilikom plaćanja upišite poziv na broj 12-300-005555
 U slučaju prekoračenja roka plaćanja zaračunavam zatezne kamate.
 Cijena je definirana po prodajnom tečaju PRIVREDNE BANKE ZAGREB d.d. na dan ispostavljanja računa
 Pridržavam pravo promjene vrijednosti računa ukoliko se tečaj promijeni za više od 3.
 Roba ostaje u vlasništvu ELEKTROMAX d.o.o. do uplate cjelokupnog iznosa.

Robu primio: _____

Robu izdao: _____

Broj vozila: _____



Registriрани korisnik: ELEKTROMAX D.O.O. | Obrasc: A00

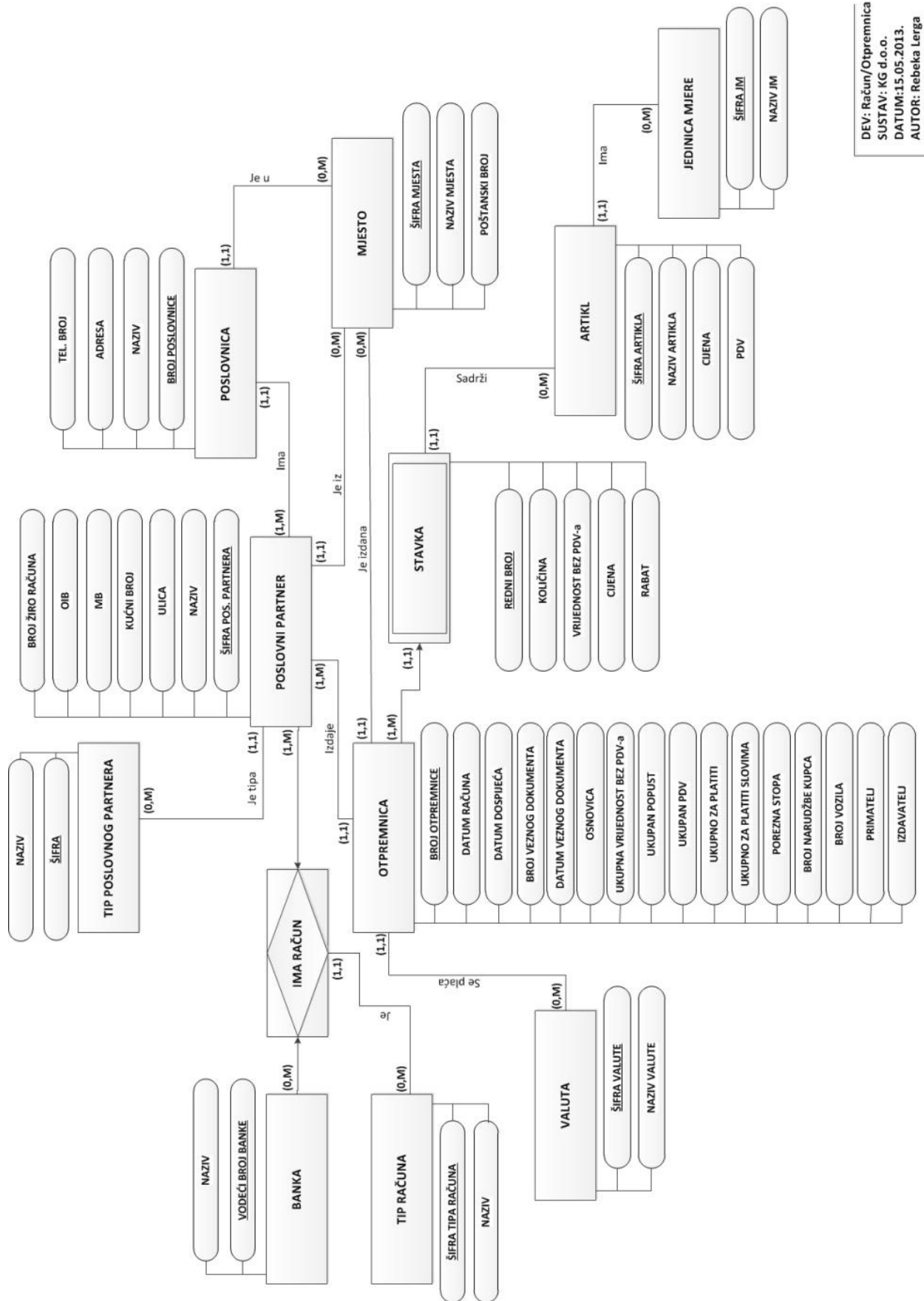
Trgovački sud u Rijeci Tt-95/9784-2, MBS 040036138

Slika 11 Račun/Otpremnica

Tablica 3: Opis Otpremnice

NAZIV DOKUMENTA		Račun/Otpremnica		
OPIS DOKUMENTA		Služi kao potvrda da je roba otpremljena kupcu. Sadrži informacije o količini, vrsti, nazivu i cijeni otpremljene robe.		
Rbr.	Naziv podatka	Opis podatka	Primjer	Tip i duljina
1.	Dobavljač	Naziv dobavljača	"Elektromax d.o.o."	S30
2.		Adresa dobavljača	N. Cara 2	S30
3.		Poštanski broj dobavljača	51000	N5
4.		Naziv mjesta dobavljača	Rijeka	S30
5.	Mjesto izdavanja	Naziv mjesta	Marinići	S20
6.	Datum računa	Datum izdavanja računa	6.12.2012.	D
7.	Datum dospijeća	Datum dospijeća računa	4.2.2013.	D
8.	Odgovorna osoba	Ime djelatnika	Milan M.	S30
9.	Narudžba kupca broj	Šifra narudžbe	13-12-54854	N10
10.	MB Kupca	Matični broj kupca	54847854	N8
11.	OIB Kupca	Osnovni identifikacijski broj kupca	87845985781	N11
12.	Nabavljač	Naziv nabavljača	"KG d.o.o."	S30
13.		Adresa nabavljača	Ludvetov Breg 20	S50
14.		Poštanski broj nabavljača	51000	N5
15.		Naziv mjesta	Rijeka	S20

		nabavljača		
16.	Vezni dokument	Broj veznog dokumenta	12-010-008024	N11
17.		Datum veznog dokumenta	6.12.2012.	D
18.	Rbr.	Redni broj robe	1.	N5
19.	Šifra	Šifra robe	11654	N10
20.	Naziv	Naziv robe	BRODSKA SVJ. 60W BIJELA	S50
21.		Oznaka robe	C56112	S10
22.	Količina	Količina otpremljene robe	200	N5
23.	Cijena	Cijena jedne jedinice mjere robe	9,29	N 10,2
24.	R%	Postotak rabata	18,00%	N4,2
25.	PDV%	Postotak PDV-a	25%	N4
26.	Vrij. bez PDV	Iznos u koji nije uračunat PDV	37,16	N10,2
27.	Za platiti	Iznos koji plaća kupac	597,75	N10,2
28.	Porezne stope	Postotak PDV-a	25%	N10,2
29.	Osnova	Cijena robe bez uračunatog PDV-a	478,20	N10,2
30.	Vrijednost	Cijena robe sa uračunatim PDV-om	597,75	N10,2
31.	Robu primio	Potpis odgovorne osobe koja je primila robu	Ivan P.	S20
32.	Robu izdao	Potpis odgovorne osobe kaj je robu izdala i pečat	Milan M.	S20
33.	Broj vozila	Broj vozila koje je dostavilo robu	RI-542-FG	S5



DEV: Račun/Otpremnica
 SUSTAV: KG d.o.o.
 DATUM: 15.05.2013.
 AUTOR: Rebecka Lerga

Slika 12 DEV: Račun/Otpremnica KG d.o.o.

4. Zaključak

Uvođenje informacijske tehnologije vrlo je složen posao. Procesu razvoja i izgradnji informacijskog sustava različito se pristupa i postoje mnogobrojne metodologije koje definiraju različite pristupe. U ovom radu detaljnije su opisani pristupi izgradnji informacijskog sustava prema metodologijama MIRIS i SSADM.

Životni ciklus metodologije MIRIS je „V“ model. Faze razvoja su podijeljene u dvije skupine: logičko oblikovanje (projektiranje informacijskog sustava) i fizičko oblikovanje (izgradnja informacijskog sustava). Svaka skupina ima tri faze koje se dalje dijele u aktivnosti. Izgradnja započinje izradom strateškog plana, potom se u fazi Glavnog projekta analizira poslovanje i kreira model procesa. U fazi Izvedbenog projekta modeliraju se podaci i definira logička arhitektura programskog proizvoda. Faza Proizvodnja softvera započinje planiranjem proizvodnje. Cilj je reorganizirati bazu podataka i kreirati novi programski proizvod ili zamijeniti postojeći programski proizvod. Gotov programski proizvod uvodi se u sustav. Po potrebi se dijelovi informacijskog sustava održavaju kako bi on zadovoljio potrebe korisnika. Metodologija SSADM visoko je strukturirana metodologija s detaljnim pravilima i smjernicama za primjenu. Usvaja tri perspektive usmjerene na: funkcije - oblikovanjem toka procesa, na događaje - oblikovanjem događaja/entiteta kroz povijest života entiteta (Entity Life Histories) i dijagrame podudarnosti učinka (Effect Correspondance Diagrams) te na podatke - logičkim oblikovanjem podataka. SSADM verzija 4 sadrži pet modula podijeljenih u faze, a faze u daljnje aktivnosti koje se detaljnije razrađuju. SSADM je metodologija razvoja informacijskih sustava preporučena za korištenje za potrebe britanske vlade, a tijekom vremena primjenjuje se i na drugim projektima.

Važno je napomenuti da ne postoji jedan jedinstven put izgradnje informacijskog sustava. Postoji više dobrih rješenja pa nije dovoljna jedna metodologija za sve slučajeve. Zato se broj postojećih metodologija i pristupa izgradnji informacijskog sustava povećava.

5. Popis slika

Slika 1 Dijagram funkcionalne raščlane KG d.o.o.	22
Slika 2 Dijagram konteksta KG d.o.o.	23
Slika 3 DTP: 0. Poslovanje KG d.o.o.	24
Slika 4 DTP: 1. Nabavljanje KG d.o.o.	25
Slika 5 DTP: 2. Prodavanje KG d.o.o.	26
Slika 6 DTP: 3. Skladištenje KG d.o.o.	27
Slika 7 DTP: 1.1 Nabavljanje od dobavljača iz RH KG d.o.o.	28
Slika 8 DTP: 1.2. Nabavljanje od inozemnih dobavljača KG d.o.o.	29
Slika 9 DTP: 2.1. Prodavanje fizičkim osobama KG d.o.o.	30
Slika 10 DTP: 2.2. Prodavanje pravnim subjektima KG d.o.o.	31
Slika 11 Račun/Otpremnica	32
Slika 12 DEV: Račun/Otpremnica KG d.o.o.	35

6. Literatura

- Čerić, V., Varga, M., Birolla, H., Budin, L., Budin Posavec, A., Ribarić, S., . . . Baranović, M. (1998). *Poslovno računalstvo*. Zagreb: Znak d.o.o.
- e-obrazovanje*. (04.. 09. 2014). Dohvaćeno iz <http://e-obrazovanje.wikispaces.com/Pristup+izgradnji+informacijskog+sustava>
- Krleza, D. (02.. 09. 2014). *www.fer.unizg.hr*. Dohvaćeno iz Fakultet elektrotehnike i računarstva: https://www.fer.unizg.hr/_download/repository/KDI,_Dalibor_Krleza.pdf
- Panian, Ž., Čurko, K., Bosilj Vukšić, V., Čerić, V., Pejić Bach, M., Požgaj, Ž., . . . Varga, M. (2010). *Poslovni informacijski sustavi*. Zagreb: Element.
- Pavlić, M. (1996). *Razvoj informacijskih sustava*. Zagreb: ZNAK.
- Pavlić, M. (2011). *Informacijski sustavi*. Zagreb: Školska knjiga.
- Pavlić, M. (2011a). *Oblikovanje baze podataka*. Rijeka: Odjel za informatiku Sveučilišta u Rijeci.
- Srića, V., Bosilj Vukšić, M., Čurko, K., Pejić Bach, M., Strugar, I., & Škoro, I. (1999). *Menedžerska informatika*. Zagreb: M.E.P. Consult.
- Tudor, D. J., & Tudor, I. J. (1997). *System analysis and Design - A Comparison of Structured Methods*. Wiltshire: MacMillan.
- Tudor, M. (02.. 09. 2014). *Pomorski fakultet Rijeka*. Dohvaćeno iz <http://www.pfri.uniri.hr/~tudor/materijali/Informacijski%20sustavi,%20baze%20podataka.htm>
- Varga, M., & Čurko, K. (02.. 09. 2014). *web.efzg.hr*. Dohvaćeno iz Ekonomski fakultet Zagreb: <http://web.efzg.hr/dok/inf/uis/PIS-%204.pdf>
- Vukmirović, S., & Čapko, Z. (02.. 09. 2014). Dohvaćeno iz infosustav: <http://www.infosustav.com/oap/modeliranje2.pdf>