

UNIVERZA V LJUBLJANI
EKONOMSKA FAKULTETA

MAGISTRSKO DELO

**OBLIKOVANJE MODELA ZA IZBIRO PONUDNIKA
RAČUNALNIŠTVA V OBLAKU ZA SREDNJA IN MAJHNA
PODJETJA**

Ljubljana, maj 2016

DAVID PAVLOVIČ

IZJAVA O AVTORSTVU

Podpisani David Pavlovič, študent Ekonomske fakultete Univerze v Ljubljani, avtor predloženega dela z naslovom Oblikovanje modela za izbiro ponudnika računalništva v oblaku za srednja in majhna podjetja, pripravljenega v sodelovanju s svetovalcem doc. dr. Jako Lindičem

IZJAVLJAM

1. da sem predloženo delo pripravil samostojno;
2. da je tiskana oblika predloženega dela istovetna njegovi elektronski obliki;
3. da je besedilo predloženega dela jezikovno korektno in tehnično pripravljeno v skladu z Navodili za izdelavo zaključnih nalog Ekonomske fakultete Univerze v Ljubljani, kar pomeni, da sem poskrbel, da so dela in mnenja drugih avtorjev oziroma avtoric, ki jih uporabljam oziroma navajam v besedilu, citirana oziroma povzeta v skladu z Navodili za izdelavo zaključnih nalog Ekonomske fakultete Univerze v Ljubljani;
4. da se zavedam, da je plagiatorstvo – predstavljanje tujih del (v pisni ali grafični obliki) kot mojih lastnih – kaznivo po Kazenskem zakoniku Republike Slovenije;
5. da se zavedam posledic, ki bi jih na osnovi predloženega dela dokazano plagiatorstvo lahko predstavljalo za moj status na Ekonomski fakulteti Univerze v Ljubljani v skladu z relevantnim pravilnikom;
6. da sem pridobil vsa potrebna dovoljenja za uporabo podatkov in avtorskih del v predloženem delu in jih v njem jasno označil;
7. da sem pri pripravi predloženega dela ravnal v skladu z etičnimi načeli in, kjer je to potrebno, za raziskavo pridobil soglasje etične komisije;
8. da soglašam, da se elektronska oblika predloženega dela uporabi za preverjanje podobnosti vsebine z drugimi deli s programsko opremo za preverjanje podobnosti vsebine, ki je povezana s študijskim informacijskim sistemom članice;
9. da na Univerzo v Ljubljani neodplačno, neizključno, prostorsko in časovno neomejeno prenašam pravico shranitve predloženega dela v elektronski obliki, pravico reproduciranja ter pravico dajanja predloženega dela na voljo javnosti na svetovnem spletu preko Repozitorija Univerze v Ljubljani;
10. da hkrati z objavo predloženega dela dovoljujem objavo svojih osebnih podatkov, ki so navedeni v njem in v tej izjavi.

V Ljubljani, dne 6.6.2016

Podpis študenta: _____

KAZALO

UVOD	1
1 RAČUNALNIŠTVO V OBLAKU.....	4
1.1 Zgodovina računalništva v oblaku.....	4
1.2 Opredelitev računalništva v oblaku	5
1.3 Storitveni modeli računalništva v oblaku	9
1.3.1 Infrastruktura kot storitev – IaaS	10
1.3.2 Računalniško okolje kot storitev – PaaS	12
1.3.3 Aplikacija kot storitev – SaaS	14
1.3.4 Ostale različice storitvenega modela računalništva v oblaku	15
1.4 Izvedbeni modeli računalništva v oblaku	16
1.4.1 Javni oblak.....	16
1.4.2 Privatni oblak.....	17
1.4.3 Hibridni oblak.....	18
1.4.4 Oblak skupnosti	18
1.5 Predpogoj za izvedbo računalništva v oblakih	19
1.5.3 Virtualizacija kot eden najpomembnejših elementov računalništva v oblakih ..	19
1.5.2 Ostali elementi za zagotavljanje obstoja računalništva v oblaku	20
1.6 Prihodnost računalništva v oblaku.....	21
2 ANALIZA MODELOV ZA IZBIRO PONUDNIKA OBLAKA.....	21
2.1 Predstavitev obstoječih modelov	22
2.2 Pomanjkljivosti obstoječih modelov	27
3 OPREDELITEV KRITERIJEV	28
3.1 Finančni kriteriji	29
3.1.1 Stroški.....	29
3.1.2 Modeli obračuna storitve	30
3.2 Kriteriji o kakovosti storitve.....	32
3.2.1 Zmogljivost.....	33
3.2.1.1 Odzivnost.....	33
3.2.1.2 Razpoložljivost	34
3.2.2 Zanesljivost.....	35
3.2.3 Varnost.....	36
3.2.4 Skalabilnost	37
3.2.5 Elastičnost.....	38
3.3 Funkcionalni kriteriji	38
3.3.1 Uporabnost.....	38
3.3.2 Prenosljivost	39
3.4 Pravni kriteriji.....	39
3.5 Ekološki kriteriji.....	39
4 ŠTUDIJA PRIMERA: PROGRAMSKA REŠITEV ZA MARINE	40

4.1 Opis programske rešitve	41
4.2 Funkcionalna področja programske rešitve	42
4.3 Arhitektura programske rešitve	43
4.4 Motivacija za uvedbo oblaka	44
4.5 Analiza zahtev	46
4.5.1 Strojna oprema	46
4.5.2 Licenciranje	47
4.5.3 Delovanje v nepovezanem načinu	47
4.5.4 Varnost podatkov	48
4.6 Opredelitev kriterijev	48
4.7 Analiza ponudnikov računalništva v oblaku	50
4.7.1 Slovenskih trg ponudnikov Iaas	51
4.7.2 Evropski trg ponudnikov Iaas	53
4.7.3 Vodilni ponudniki IaaS	53
5 IZDELAVA MODELA ZA IZBIRO PONUDNIKA RAČUNALNIŠTVA V OBLAKU S POMOČJO METODOLOGIJE ANALITIČNO HIERARHIČNEGA PROCESA (AHP)	56
5.1 Opis metodologije analitično hierarhičnega procesa – AHP	57
5.1.1 Večparametrsko odločanje	58
5.1.2 Hierarhični večparametrski model	59
5.1.3 Analitično hierarhični proces – AHP	61
5.1.3.1 Korak 1 – opredelitev problema in cilja	63
5.1.3.2 Korak 2 – sestava hierarhične strukture	64
5.1.3.3 Korak 3 – parna primerjava	64
5.1.3.4 Korak 4 – opredelitev uteži	65
5.1.3.5 Korak 5 – preverjanje skladnosti	66
5.1.3.6 Korak 6 – določitev ocene alternative	67
5.1.3.7 Korak 7 – odločitev in analiza spremembe parametrov	68
5.2 Opredelitev kriterijev in strukture modela	68
5.2.1 Zasnova modela	68
5.2.2 Struktura modela	69
5.2.2.1 Finančni kriteriji	70
5.2.2.2 Kakovostni kriteriji	71
5.2.2.3 Funkcionalni kriteriji	73
5.2.2.4 Kriteriji zanesljivosti	73
5.3 Opredelitev pomembnosti kriterijev in njihova uskladitev	74
5.4 Opredelitev alternativ modela	78
5.5 Sprejetje odločitve	80
SKLEP	80
LITERATURA IN VIRI	83

KAZALO TABEL

Tabela 1: Seznam obstoječih modelov in uporabljena metodologija	23
Tabela 2: Modeli obračunov za oblak	32
Tabela 3: Seznam oznak razpoložljivosti z maksimalnimi časi izpada.....	34
Tabela 4: Seznam slovenskih ponudnikov IaaS	52
Tabela 5: Parna primerjava elementov	65
Tabela 6: Temeljna skala za parno primerjanje elementov	65
Tabela 7: Slučajni indeksi v odvisnosti od dimenzije matrike	67
Tabela 8: Kriteriji kakovosti.....	72
Tabela 9: Funkcionalni kriteriji	73
Tabela 10: Parna primerjava kriterijev prve ravni.....	75
Tabela 11: Normalizirana matrika in izračunan prioritetni vektor	76
Tabela 12: Pomembnost kriterija FINANCE	76
Tabela 13: Model za izbiro ponudnika v oblaku	77
Tabela 14: Parna primerjava alternativ za kriterij "Ugled na trgu"	79
Tabela 15: Prioritetna vektorja za alternative po kriteriju Ugled	79

KAZALO SLIK

Slika 1: Model delitve storitev po NIST.....	9
Slika 2: Ravni delitve oblaka po UCSB-IBM	10
Slika 3: Struktura hibridnega oblaka	18
Slika 4: Struktura virtualizacije	20
Slika 5: Struktura kriterijev pridobljenih iz literature	29
Slika 6: Tradicionalna IT zasnova	43
Slika 7: Zasnova z uporabo oblaka.....	44
Slika 8: Vodilni ponudniki storitev IaaS	54
Slika 9: Hierarhični prikaz modela.....	63
Slika 10: Osnova struktura modela.....	70
Slika 11: Finančni kriteriji.....	71
Slika 12: Kriteriji kakovosti	72
Slika 13: Kriteriji zanesljivosti	74

UVOD

Osnovna ideja o tem, da bi računalniške vire uporabili kot storitev, t.i. storitveno računalništvo (angl. *utility computing*), se je porodila že v zgodnjih letih računalništva in je bila prvič javno omenjena že v leta 1955, ko je znanstvenik in pionir računalništva John McCarthy na konferenci v Dartmouthu v Kanadi podal zasnovo o tem, kako naj bi računalništvo v prihodnosti ponudili kot javno dostopno storitev, kot je npr. elektrika, voda, telefon. Ob tem je dodal, da bo to zelo pomembna doba razvoja računalniške industrije (Filipe, 2008, str. 18). Tudi računalniški znanstvenik in profesor na University of California Los Angeles, Leonard Kleinrock, je leta 1969 izjavil, da bo razvoj računalniških omrežij omrežja tako izpopolnil in se bodo razširila v takšni meri, da bo storitveno računalništvo na voljo v vsakem gospodinjstvu, tako kot je oskrba z električno energijo (Antonopoulos & Gillam, 2010, str. 63).

Razvoj računalniških omrežij je res dosegel, da je v današnjem razvitem delu sveta poveztljivost z internetom enako dosegljiva kot oskrba s pitno vodo ali električno energijo. Ravno to pa nam poleg ostalih ugodnosti omogoča tudi uporabo računalniških virov na daljavo. Podobno kot uporabo električne energije lahko tudi oskrbo z računalniškimi viri, kot so omrežje, procesorski čas, pomnilniški prostor, aplikacije ipd., uporabimo in s tem tudi plačamo le toliko, kolikor smo porabili. S tem računalniške vire najamemo, namesto, da bi jih kupili. Resda uporabniki vseeno potrebujejo minimalno računalniško opremo, vendar je tu govora le o lahkih odjemalcih, kot so npr. pametni telefoni, tablice in prenosniki, ostala strojna arhitektura pa je uporabniku skrita in zanj nepomembna.

In ravno ideja o najemu računalniških virov kot storitev nas pripelje do računalništva v oblaku (angl. *cloud computing*) ali krajše oblak (angl. *cloud*). Oblak je model, ki omogoča dostop do računalniških virov. Računalniški viri so nastavljivi in se prilagajajo zahtevam uporabnika. Definicija navaja pet glavnih lastnosti (Mell & Grance, 2011, str. 2–3):

- samopostrežna storitev na zahtevo,
- širok dostop do omrežja,
- akumulacija virov,
- hitra elastičnost,
- merljiva storitev;

in tri ravni oblaka:

- aplikacija kot storitev (angl. *Software as a Service*, v nadaljevanju SaaS),
- računalniško okolje kot storitev (angl. *Platform as a Service*, v nadaljevanju PaaS),
- infrastruktura kot storitev (angl. *Infrastructure as a Service*, v nadaljevanju IaaS);

ter štiri modele glede na dostopnost oblaka:

- privatni oblak,
- oblak skupnosti,
- javni oblak,
- hibridni oblak.

Kot že opisano zgoraj, oblaki podjetjem omogočajo najem računalniških virov, namesto da bi jih kupili. Če se osredotočim predvsem na področje delovanja IaaS z nekaterimi elementi PaaS, to vključuje celotno strojno in programsko infrastrukturo, ki je potrebna za delovanje podjetju ključnega informacijskega sistema. Infrastruktura vključuje strežnike in delovne postaje (procesorski čas, pomnilniški prostor, ...), mrežne komponente, varnostne komponente ter podatkovne baze in operacijske sisteme. Nakup opreme ne vključuje le začetnega stroške za strojno opremo, programsko opremo in licence, temveč tudi vzdrževanje in posodabljanje tako programske kot tudi strojne opreme. Začetni strošek nakupa je seveda odvisen od zahtev informacijskega sistema in navadno predstavlja resen poseg v finančno stanje podjetja. Majhna in srednje velika podjetja navadno nimajo oddelka za informacijsko tehnologijo (angl. *Information Technology*, v nadaljevanju IT), kar pomeni, da morajo celotno vzdrževanje infrastrukture oddati v zunanje izvajanje. V primeru uporabe oblaka podjetju z najemom računalniških virov ni treba skrbeti za vzdrževanje in posodabljanje le teh in se to izvaja s strani ponudnika oblaka. Po drugi strani pa podjetje z najemom izgubi nadzor nad infrastrukturo, pojavijo se dvomi glede varnosti podatkov, skrb glede dostopnosti ipd. Posledično si podjetja poleg analize stroškov, zastavijo marsikatero vprašanje v zvezi s prednostmi in slabostmi realizacije infrastrukture s pomočjo lastne strojne opreme (lahko tudi lastna izvedba oblaka) ali z uporabo oblakov.

Namen naloge je izdelati model, ki bo majhnim in srednje velikim podjetjem pomagal pri odločitvi, na kakšen način in pri katerem ponudniku implementirati oblak z upoštevanjem različnih kriterijev. Kriteriji zajemajo ekonomsko, tehnično, pravno in ekološko področje. Pri tem se dotaknem tudi vprašanja lastne izvedbe oblaka v primerjavi z obstoječimi ponudniki oblakov. Kot uvedba lastnega oblaka opisujem celotno strojno infrastrukturo v lasti podjetja. Izziv pri tem je, da bo model tako transparenten, da bo uporaben za širšo množico malih in srednje velikih podjetij, brez večjih modifikacij v strukturi modela.

Cilji dela, ki sem si jih zastavil, so:

- preučiti področje oblakov in analizirati trg tujih in domačih ponudnikov.
- Kot glavni cilj poskušam razviti uporaben model, ki bo podjetjem v pomoč pri sprejemanju odločitev ali strojno infrastrukturo kupijo ali jo najamejo v oblaku ter katera implementacija oblaka je primerna ob določenih kriterijih. Pri izbiri množice

kriterijev bom upošteval, da bodo na eni strani dovolj dobro opredelili zahteve in potrebe ter na drugi strani ohranili enostavnost in s tem razumevanje modela. Pri tem želim doseči, da bo struktura modela toliko dinamična, da bo model še vedno uporaben tudi v primeru manjših sprememb v strukturi. Na ta način bo model mogoče uporabiti tudi za podjetja, ki tržijo svojo programsko opremo majhnim in srednje velikim podjetjem. Z modelom lahko svetujejo svoji stranki, na kakšen način je najbolj optimalno implementirati infrastrukturo glede na njihove zahteve in potrebe.

- Izdelani model bom preveril in ovrednotil na realnem primeru, ter izbral najprimernejšega kandidata za izbran primer. Pri tem si bom pomagal z uveljavljenimi metodologijami.

S pomočjo domače in tuje strokovne literature sem raziskal področje oblakov in se dotaknil same definicije oblakov. Pri tem želim definicijo primerjati s ponudbo vodilnih svetovnih in nekaterih domačih ponudnikov oblakov.

S pomočjo sekundarnih virov poiščem strokovne članke in znanstvene prispevke, kjer bom skušal najti že obstoječe modele, ki se navezujejo na vprašanje uvedbe oblakov. Modele bom analiziral, poiskal sinergijo s svojim modelom in jih kritično primerjal.

Izdelan model želim preveriti tudi na praktičnem primeru. Kot navaja Eisenhardt (1989, str. 537) je treba izbiri študije primerov nameniti posebno pozornost in odsvetuje naključno izbiro le teh. Eden od razlogov, ki jih navaja Yin (2009, str. 40–42) je izbira reprezentativnega ali tipičnega primera. Na podlagi tega sem izbral malo slovensko podjetje iz Ljubljane, ki ponuja programsko rešitev namenjeno upravljanju marin.

Programska rešitev je namenjena vsem zaposlenim v marini, tako vodstvu kot mornarjem in gostom marine. Za uporabo rešitve mora marina poskrbeti za zmogljivo strojno opremo. Vzpostavitev lastne tradicionalne strojne infrastrukture marinam ne pomeni le začetne finančne skrbi, temveč navadno tudi resno tehnično in kadrovsko oviro, v skrajnih primerih celo preprečitev informatizacije. Izbrano podjetje želi z razvojem in svetovanjem marinam pomagati pri čim lažji in transparentni informatizaciji. Svojim strankam mora biti sposobno svetovati in pomagati pri izbiri različnih možnosti za vzpostavitev strojne infrastrukture, potrebne za delovanje programske rešitve. Ravno nakup strojne opreme, kot tudi njeno vzdrževanje in nadgradnja, je skrb, katero želi ponudnik v sodelovanju s svojimi strankami, torej marinami, prenesti na oblak.

S pomočjo primarnih virov (literature podjetja in poglobljenih intervjujev z zaposlenimi v podjetju) bom opredelil zahtevano infrastrukturo, ki bo temelj za izbiro IaaS alternativ ponudnikov na domačem in tujem trgu. Kritično bom ocenil njihovo primernost glede na vstopne kriterije in izločil najmanj primerne alternative. Ožjo množico alternativ bom podrobneje ovrednotil s pomočjo razvitega modela. Z uporabo metodologije analitično hierarhičnega procesa (angl. *Analytics hierarchy process*, v nadaljevanju AHP) bom

določil pomembnost kriterijev ter izvedel primerjavo alternativ in določil najprimernejšo med njimi. Metodologija AHP omogoča logično in znanstveno podlago za odločanje, kjer paroma primerjamo kriterije za doseganje končnega cilja (Saaty 1977; Harker 1988). Izbrana metodologija je po trditvah Lee, Tang in Sugumaran (2012, str. 1) zelo uspešna in pogosto uporabljena odločitvena metoda.

Strukturo magistrskega dela v prvem delu sestavlja teoretično poglobljen opis oblaka in njegova opredelitev, delitev storitev oblaka, vrste oblaka ter zgodovina razvoja oblaka. Sledi analiza obstoječih modelov v literaturi in podrobna opredelitev kriterijev po področjih. Nadaljujem s študijo primera, kjer opišem funkcionalnosti programske rešitve, njene strojne zahteve ter nefunkcionalne zahteve in pričakovanja strank v zvezi z rešitvijo. V sklopu študije primera izvedem analizo trga ponudnikov za IaaS. Sledi empirični del naloge, kjer poiščem sintezo med kriteriji in njihovimi pomembnostmi iz literature in zahtevami iz študije primera. Na osnovi dobljenih rezultatov s pomočjo metodologije AHP izdelam model. Model preverim na primeru alternativ in na koncu v sklepu povzamem ugotovitve.

1 RAČUNALNIŠTVO V OBLAKU

Če pogledamo nazaj v industrijsko revolucijo in njen vpliv na svetovno ekonomijo ta ni povzročila sprememb tekom noči, temveč skozi čas. Če za primer pogledamo razvoj interneta, se je ta vsekakor razvijal skozi obdobja in v več fazah. In vsekakor ima oblak in razvoj oblaka velik potencial za revolucionarne spremembe (Mather, Kumaraswamy, & Latif, 2009, str. 7). Sama tehnologija oblaka sicer ni revolucionarna, temveč evolucijska. Revolucionarna je uporaba te tehnologije, ki uporabnikovo samoumevnost, da so zahtevani podatki in računalniški viri na voljo, enači s samoumevnostjo elektrike v vtičnici (Höllwarth, 2012, str. 30). Prav to podjetjem omogoča, da lahko celotno skrb za informacijsko tehnologijo, ki je potrebna za upravljanje podjetja, v celoti nadomesti z oblakom. To je še toliko bolj pomembno za manjša in srednje velika podjetja, ki imajo dodatno omejene vire in navadno nimajo IT oddelka. To podjetjem omogoča boljše osredotočanje na primarno dejavnost in posledično kakovostnejše izdelke in storitve.

1.1 Zgodovina računalništva v oblaku

Zgodovina oblaka se je začela pisati z zgodovino predhodnih tehnologij, ki so potrebne za obstoj oblaka. Sam se bom v nadaljevanju osredotočil le na zgodovino, ki je določeno tehnologijo, model ali storitev že prepoznala kot oblak.

Termin oblakov prvotno prihaja iz telekomunikacije, pri povezovanju dveh točk, npr. navidezno zasebno omrežje (angl. *virtual private network* – VPN), kot grafična

predstavitev vmesne javne komunikacijske infrastrukture. Pogosto se ga uporablja tudi za grafični prikaz interneta.

Za namen oblakov ga je prvič uporabil prof. R.K. Chellappa leta 1997 na sestanku v Dallasu v Združenih državah Amerike (v nadaljevanju ZDA), kjer navaja oblake kot dinamično računalniško paradigmo, kjer bodo meje določene kvečjemu iz ekonomske ali organizacijske in ne iz tehnične plati. Struktura računalniškega sistema bo prešla iz osnovnega centralnega računalnika (angl. *mainframe*) na omrežno strukturo (Chellappa & Gupta, 2002, str. 118). V intervjuju za Fujitsu IT Compass (Fujitsu, 2010) je povedal, da je besedno zvezo uporabil kot nejasno sliko o tem, kako se bo model o storitvenem računalništvu širil v prihodnosti in do kam bo segal.

Na trgu so se pojavili ponudniki, ki so nudili storitve, katere so sestavljene iz raznih kombinacij različnih področij računalništva, npr. omrežno in storitveno računalništvo. Ker je bila takrat definicija oblakov še zelo nedefinirana, je bilo pri trženju njihovih storitev moč zaslediti pojav termina oblakov. Vendar o pravih oblakih, kakršni so definirani danes, še ne moremo govoriti, saj niso zadoščali vsem potrebnim pogojem. Takrat morda posamezni kriterij še niti ni obstajal (Marks & Lozano, 2010, str. 9).

Marks in Lozano (2010, str. 14) še navajata, da bi prvo prepoznavo oblaka lahko podelili podjetju Google, ko je izdelal svoj iskalnik po internetu. Razlogi, zakaj te lovorike niso osvojili drugi predhodniki (npr. pionir »indeksiranje interneta« Yahoo), so v tem, kako in na kakšen način je Google to razvil in uvedel.

Pionir oblaka, ki je hkrati povzročil popularizacijo oblaka, je podjetje Amazon, ki je leta 2006 objavil Amazon Web Services (v nadaljevanju AWS) ter storitvi Amazon Elastic Compute cloud (v nadaljevanju EC2) in Amazon Simple Storage Service (v nadaljevanju Amazon S3). Pri EC2 gre za skupek storitev, ki v skladu z opredelitvijo oblaka ponuja procesor, pomnilniški prostor, podatkov shrambo, omrežje itd., Amazon S3 pa je storitev za shrambo podatkov (Clark, 2012; Marks & Lozano, 2010, str. 9; Mohamed, b.l.; Sarna, 2010, str. 115).

Zanimivo je omeniti pozen prihod podjetja Microsoft na oblak. Ta je šele leta 2008 najavil, da vstopa v svet oblaka s storitvijo Windows Azure, ki ponuja računalniško okolje, vmesno programje ter poslovne storitve (Redkar, 2009, str. 46).

1.2 Opredelitev računalništva v oblaku

»Ne bom niti poizkusil povedati kaj oblak je. Vsakdo ima svojo definicijo.«, je zapisal Andi Gutmans, soavtor skriptnega jezika PHP (angl. *Hypertext Preprocessor*), ki ne samo, da je vodilni skriptni jezik za spletne aplikacije, temveč tudi vodilni za implementacijo aplikacij za oblak (Babcock, 2010, str. 5).

V strokovni literaturi je mogoče zaslediti ogromno (različnih) opredelitev oblaka. Nekatere opisujejo oblak v podrobnosti, druge ne pokrijejo niti glavnih značilnosti oblaka. Kot zanimivost velja omeniti opredelitev avtorjev Vanquero et al., ki skušajo celovito opredeliti oblak kar z 22 različnimi lastnostmi (Vaquero, Roderio-Merino, Caceres, & Lindner, 2008, str. 50–55). Večina opredelitev temelji ali se sklicuje sprva na delovno različico, nato na formalno opredelitev oblaka, ki ga je izdal Nacionalni inštitut za standarde in tehnologijo (angl. *National Institute of Standards and Technology*, v nadaljevanju NIST). Osnutek definicije je nastal leta 2009, kateremu sledijo popravki in izboljšave. Petnajsta različica je bila januarja 2011 objavljena in dana v javno razpravo. Oktobra istega leta je NIST nato objavil formalno opredelitev oblaka pod dokumentom NIST SP 800-145 (NIST, 2011; Hulme, 2011):

Oblak je model, ki omogoča enostaven, uporaben in preko omrežja dostopen skupek računalniških virov kjerkoli in kadarkoli. Računalniški viri, kot so omrežja, strežniki, podatkovne baze, aplikacije in storitve so nastavljivi in se prilagajajo zahtevam uporabnika z minimalnim posredovanjem ponudnika oblaka. Opredelitev navaja pet glavnih lastnosti oblaka, tri storitvene modele, ki jih oblak nudi ter štiri izvedbene modele glede na njihovo dostopnost (Mell & Grance, 2011, str. 2–3).

Lastnosti, ki jih mora imeti oblak, da izpolni zahtevo NIST opredelitve oblaka (Mell & Grance, 2011, str. 2–3) so:

- **samopostrežna storitev na zahtevo** (angl. *on-demand self-service*). Uporabnik lahko sam prilagaja želene in potrebne zmogljivosti virov (npr. strežniški čas), brez posredovanja posrednikov ali skrbnikov ponudnikov storitev.
- **Dostop do omrežja** (angl. *broad network access*). Viri in storitve so, neodvisno od njihove lokacije in ponudnika, dosegljivi preko omrežja skladno s standardi, ki jih uporabljajo heterogeni odjemalci, npr. pametni telefoni, tablice, prenosniki in delovne postaje. Uporaba terminov »enostavno dostopni s standardnimi mehanizmi« (angl. *easy-to-access standardized mechanisms*) in »sposobnost globalnega dosega« (angl. *global reach capability*) se prav tako nanašata na opisano karakteristiko.
- **Akumulacija virov** (angl. *resource pooling*). Ponudniki morajo akumulirati vire tako, da omogočijo več uporabnikom hkrati dostop do različnih fizičnih in navideznih virov. Odvisno od zahteve uporabnika, je v primeru zahteve po viru, te možno dinamično dodeliti in po uporabi odstopiti drugim. Uporabniku je informacija o točni lokaciji vira neznana in nepomembna. Lokacijo pa lahko določimo na višji ravni: v kateri državi, v katerem podatkovnem centru, ipd. se nahajajo uporabljeni viri. Kot primer vira navajam: shrambo podatkov, pomnilnik, pasovna širina omrežja, navidezne naprave, razvojne knjižnice in spletne aplikacije.
- **Odzivna elastičnost** (angl. *rapid elasticity*). Glede na povpraševanje in zahteve uporabnika se morajo kapacitete virov temu primerno odzvati s povečanjem ali

manjšanjem, hitro in uporabniku neopazno. Iz stališča uporabnika so viri na voljo v neomejenih količinah in jih je mogoče kadarkoli najeti. Navidezna neomejena količina pomeni, da zmogljivost vira za posameznega uporabnika presega vse potrebe in jo lahko enačimo z neomejenim.

- **Merljiva storitev** (angl. *measured service*). Zagotoviti je treba avtomatsko meritev uporabe virov in njihov nadzor. Glede na vrsto storitve se to izvaja na različnih ravneh delovanja (npr. pomnilnik, procesorski čas, pasovna širina, število uporabnikov). Uporaba virov je torej merljiva in jo je možno predstaviti kot poročilo uporabe. Zagotovljena mora biti transparentnost s strani ponudnika in uporabnika. Uporabi se poslovni model imenovan »plačilo po uporabi« (angl. *pay-per-use*).

Oblak ponuja različne storitvene modele in vsak od njih mora zadoščati zgoraj naštetim lastnostim. V odvisnosti od tega, katere računalniške vire bo uporabnik najel, NIST razlikuje tri storitvene modele oblaka (Mell & Grance, 2011, str. 2–3):

- **aplikacija kot storitev – SaaS** nudi uporabniku aplikacije, ki se izvaja na oblaku, ki zadostuje prej naštetim lastnostmi. Dosegljivost in uporaba aplikacije je neodvisna od vrste odjemalca. Uporabniku je infrastruktura oblaka (omrežje, strežniki, operacijski sistem, podatkovna baza) neznana in je ne more upravljati. Opravlja pa lahko nastavitve in parametre, ki so vezani na samega uporabnika aplikacije (angl. *user-specific settings*).
- **Računalniško okolje kot storitev – PaaS** je namenjena za izdelavo uporabniških programov. Uporabnik lahko izdeluje nove programe s pomočjo orodij in storitev, knjižnic in programskih jezikov, ki so na voljo s strani ponudnika. Uporabnik nima možnosti upravljanja z infrastrukturo oblaka na nižji ravni (omrežje, strežnik, operacijski sistem, podatkovna baza), jo pa preko svoje aplikacije uporablja. Omogočen pa mu je nadzor in upravljanje nad nameščenim uporabniškim programom in nad nekaterimi parametri za gostitev programa.
- **Infrastruktura kot storitev – IaaS** nudi uporabniku neposredno uporabo procesorskega časa, podatkovne shrambe, omrežja in drugih osnovnih komponent nižje ravni. Uporabnik lahko namesti in izvaja svoje aplikacije. Uporabnik ne more upravljati neposredno s strojno opremo, lahko pa upravlja z operacijskim sistemom, podatkovno bazo, nameščenimi aplikacijami in nekaterimi omrežnimi komponentami.

Neodvisno od storitvenega modela NIST opredeljuje še štiri izvedbene modele glede na njihovo dostopnost (Mell & Grance, 2011, str. 2–3):

- **privatni oblaki** (angl. *private cloud*). Infrastruktura oblaka je namenjena izključno določenemu podjetju, organizaciji ali posamezniku. Upravljanje oblaka je v domeni uporabnika oblaka, lahko tudi preko tretje osebe. Lokacija oblaka je lahko znotraj organizacije ali se gosti na drugi lokaciji.
- **Oblak skupnosti** (angl. *community cloud*). Storitve oblaka se uporabljajo znotraj določne skupnosti, katero povezuje skupna dejavnost, namen ipd. Upravljanje z oblakom je znotraj te skupnosti, s strani tretje osebe ali kombinacija prejšnjih možnosti. Tudi tu lokacija ni pomembna.

- **Javni oblak** (angl. *public cloud*). Storitve oblaka so na voljo širši javnosti in podjetjem. Upravljanje z oblakom je v domeni ponudnika teh storitev. Lokacija je v domeni ponudnika oblaka.
- **Hibridni oblak** (angl. *hybrid cloud*). Kot že ime pove je to skupek dveh ali več različnih modelov oblaka, ki med seboj komunicirajo preko standardov, ki skrbijo za prenos podatkov in aplikacij.

Opisana opredelitev je najbolj razširjena in največkrat omenjena v strokovni literaturi. To pa ne pomeni, da ne obstajajo opredelitve, ki temeljijo na svojih dognanjih. Vsekakor velja navesti opredelitev raziskovalno-svetovalne hiše Gartner, ki je svojo opredelitev oblaka objavila leta 2008, torej pred uradno objavo NIST-a. K temu je botrovala velikokrat ohlapna in različna uporaba termina oblaka, tako s strani ponudnikov, kot tudi uporabnikov oblaka (Smith & Cearley, 2008). Navajam le Gartner povzetek opredelitve, saj je dostop do originalne opredelitve plačljiv. Gartner (2009) opredeljuje oblak kot slog računalništva, kjer so IT viri na voljo uporabniku z uporabo internetne tehnologije. Pri tem mora oblak izpolnjevati pet lastnosti:

- **storitvenost** (angl. *service-based*). Uporabniku je na voljo storitev, katere vmesniki so dobro definirani in pripravljeni za uporabo. Kakovostno storitve določajo uporabniku občutljivi atributi kot so razpoložljivost storitve, latenca, kapacitete v primerjavi s ceno in strojna oprema uporabljena za delovanje storitve. Implementacija storitve je uporabniku nepomembna in skrita. Pomembno je, kaj storitev ponuja in ne, kako to naredi.
- **Skalabilnost in elastičnost** (angl. *scalable and elastic*): storitev se prilagaja uporabnikovim zahtevam in potrebam z rezervacijo in sproščanjem virov. To se izvaja avtomatsko, pri nekaterih storitvah hitreje, pri drugih počasneje. Elastičnost je lastnost skupne uporabe virov, skalabilnost pa kapacitete nižje ležeče infrastrukture. Elastičnost omogoča čim bolj ekonomično uporabo virov.
- **Skupna raba** (angl. *shared*): Skupna raba nižje ležeče infrastrukture omogoča, da se jo deli med različnimi uporabniki storitev, ne da bi se uporabniki tega zavedali. S tem dosežemo maksimalno učinkovitost pri izrabi računalniških virov.
- **Merljivost po uporabi** (angl. *metered by use*): uporaba storitve se mora meriti po njeni uporabi. Na podlagi tega ponudnik določi različne načine najema storitve. Cena najema mora temeljiti na storitvi in ne na ceni računalniškega vira. Enoto plačevanja določi ponudnik in je lahko čas uporabe nekega vira, količina prenesenih podatkov, ipd. Model obračuna je lahko po uporabi, časovni naročnini ali v naprej dogovorjenem zakupu.
- **Uporaba tehnologije interneta** (angl. *uses internet technologies*): storitev je dostopna z uporabo omrežja z uveljavljenimi internetnimi standardi in formati (npr. protokol za izmenjavo hiperteksta ter grafičnih, zvočnih in drugih večpredstavnostnih vsebin na spletu (angl. *hypertext transfer protocol* – HTTP), ...).

Glede na dostopnost računalniških virov Gartner prav tako razlikuje dva izvedbena modela oblaka: privatnega in javnega.

Primerjava opisanih opredelitev kaže, da opredelitve izhajajo iz dejanskega stanja trga ponudnikov oblakov, kjer so koncepti delovanja že bili zasnovani in uveljavljeni ter temeljijo na skupni ideji o storitvenem računalništvu.

1.3 Storitveni modeli računalništva v oblaku

Storitveni model oblaka opredeljuje, kateri so tisti računalniški viri, ki jih ponudnik ponuja oz. jih uporabnik lahko najame kot storitev. Poleg termina storitveni modeli oblaka se v strokovni literaturi uporablja še termini plasti oblaka, stebri oblaka, vrste oblaka in struktura oblaka.

Prav tako obstajajo različne delitve storitvenih modelov glede na to, katere vire vključujejo posamezne ravni. V opredelitvi oblaka sem na kratko že opisal NIST-ovo model, katerega krajše zapišemo **SPI** – **Software, Platform, Infrastructure**, ki pomeni aplikacijo (SaaS), računalniško okolje (PaaS) in infrastrukturo (IaaS) (Ahson & Ilyas, 2011, str. 2–3). Model je prikazan na Slika 1 in ga bom podrobneje obravnaval v naslednjih poglavjih.

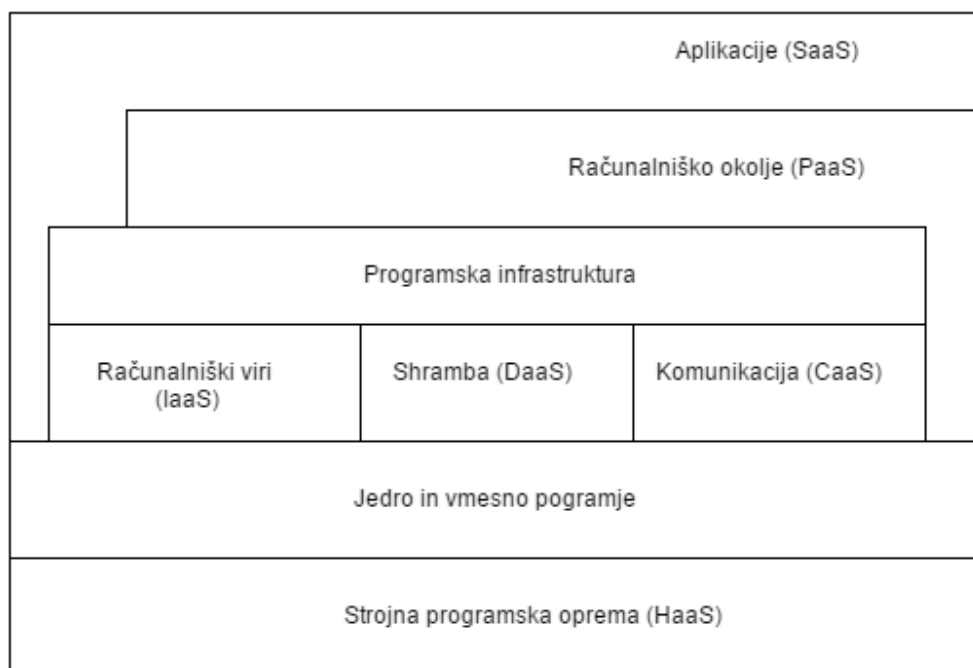
Slika 1: Model delitve storitev po NIST



Vir: T. Foo, *Cloud Computing for Financial Institutions: With the Benefits Come Security Concerns*, 2011.

Poleg opisanega modela obstajajo tudi drugi storitveni modeli. Manj uporabljen, a vseeno pomemben, tako z akademskega kot poslovnega vidika je model UCSB-IBM, ki je plod Univerze v Kaliforniji Santa Barbara (angl. *University of California, Santa Barbara - UCSB*) in podjetja IBM (oddelek IBM T.J. Watson Research Center). Model deli oblak na pet ravni, kot prikazuje Slika 2. Model je bil razvit za lažje razumevanje področja oblaka, njegovo raziskovanje in v izobraževalne namene. Omenjeni model je bil tudi večkrat razširjen s strani različnih raziskovalcev, npr. model Jackson's Expansion on the UCSB-IBM (Ahson & Ilyas, 2011, str. 4–11).

Slika 2: Ravni delitve oblaka po UCSB-IBM



Vir: S. A. Ahson & M. Ilyas, *Cloud Computing and Software Services: Theory and Techniques*, 2011, str. 5, slika 1.1.

1.3.1 Infrastruktura kot storitev – IaaS

Infrastruktura se navezuje na nižje ležeče fizične komponente, ki so potrebne za delovanje informacijskega sistema. Te komponente so procesorji, pomnilniki, omrežne komponente in v nekaterih primerih tudi podatkovne baze in operacijski sistemi (Jula, Sundararajanb, & Othmana, 2014, str. 3812). Končni uporabniki tega storitvenega modela so sistemski administratorji, ki imajo v primerjavi z ostalimi storitvenimi modeli (SaaS in PaaS) mnogo več odgovornosti za upravljanje sistema na svoji strani (Badger, Grance, Patt-Corner, & Voas, 2012, str. 7–1 in 7–5).

Pri IaaS je osnovna zahteva uporabnika najem strežnika ali več njih z možnostjo podatkovne hrambe. Naloga ponudnika je, da na to zahtevo uporabniku dostavi želeno v skladu s karakteristikami oblaka. Prilagoditve aplikacije, ki se bodo izvajale na strežniku in nastavitve sistema ter podatkovne baze pa so v domeni uporabnika oblaka (McDonald, 2010, str. 10). Dr. Höllwarth (2012, str. 47) navaja: »IaaS lahko razumemo tudi kot z uporabo programske opreme abstrahiran in virtualiziran vir podatkovnega centra. Tehnologije virtualizacije in velike razpoložljive pasovne širine omrežij so kot najpomembnejše osnovne tehnologije postavile temelje za oblak.«.

Motivacija najema infrastrukture v primerjavi s tradicionalnim nakupom je predvsem v racionalni rabi nižje ležečih komponent, predvsem strojne opreme. Realna ocena svetovne uporabe strojne opreme v podatkovnih centrih se giblje med 5% in 20%. Pri tem je treba upoštevati, da se v konicah uporabe potreba po virih poveča za faktor od 2 do 10. Večje kot je razmerje med njima, večje so izgube (Armbrust et al., 2009, str. 10). IaaS omogoča bolj racionalno rabo virov na način, da jo uporablja kolikor je to potrebno. Pri konicah uporabe ali pa na zahtevo uporabnika po povečanju virov, se viri rezervirajo in po njeni uporabi sprostijo – elastičnost. Pri najemu IaaS pa je po navajanju Badger et al. (2012, str. 7–8) treba upoštevati naslednja priporočila, ki jih je potrebno preveriti pri ponudniku in zagotoviti pri uporabniku:

- preprečevanje vdora med različnimi instancami v več najemniški strukturi,
- varnost podatkov, po potrebi tudi njihova lokacija,
- varno brisanje podatkov na zahtevo uporabnika,
- omejen administrativni dostop do virov na strani uporabnika,
- možnost migracije oz. izvoza instanc na druga okolja,
- upoštevanje dobrih praks virtualizacije.

Ravno pri IaaS storitvenem modelu se pojavijo vprašanja razlik tega modela v primerjavi z ostalimi področij računalništva, ki so bila v uporabi pred obstojem oblaka, kot npr. storitveno in mrežno računalništvo (angl. *grid computing*), računalništvo v gručah (angl. *cluster computing*) ter storitvena usmerjena arhitektura (angl. *service oriented architecture*, v nadaljevanju SOA). V literaturi je mogoče najti veliko primerjav, vendar bom v nadaljevanju opisal le primerjavi s storitvenim in mrežnim računalništvom, ki jo v literaturi zasledim najpogosteje.

IaaS model je podoben **storitvenemu računalništvu**, kjer je osnovna ideja zagotavljanja računalniških virov podobno kot javno dostopne storitve. Skladno s tem je tudi višina plačila storitve, ki je tolikšna, kolikor virov je uporabnik uporabil. IaaS nudi uporabniku storitev, pri tem pa ga abstrahira od podrobnosti infrastrukture, vključno s fizično strojno opremo, njeno lokacijo, podatkovno porazdelitvijo, lestvičenjem, varnostjo, varnostno kopijo podatkov, itd. Pri IaaS ima ponudnik popolni nadzor nad infrastrukturo. Uporabnik

pri storitvenem računalništvu pa hoče imeti vsaj delno kontrolo nad lokacijo strežnikov in nad tem, kaj se izvaja na katerem strežniku (Mather et al., 2009, str. 22).

Mrežno računalništvo loči dva različna, vendar povezana cilja: prvo je oddaljen dostop do IT virov in drugo združevanje procesorske moči (Zhang, Chen, Zhang, & Huo, 2010, str. V11-73). Temu sledi tudi oblak, vendar je razlika kot jo navajata Manvi in Krishna Shyamb (2014, str. 425) v tem, da je pri mrežnem računalništvu storitev ali v celoti vključena ali izključena in ne omogoča vmesnega prilaganja. IaaS pa omogoča storitev na zahtevo uporabnika in s tem eliminira nepotrebne stroške pri obračunu uporabe.

Ponudnike modela IaaS Marks in Lozano (2010, str. 264) delita v dve skupini. V eni skupini so ponudniki, ki zagotavljajo IaaS, v drugi pa ponudniki, ki zagotavljajo tehnologijo, ki omogoča IaaS. V prvo skupino poleg ponudnikov oblaka (npr. AWS) sodijo tudi upravljalci storitev oz. namenski gostitelji (npr. Rackspace in Savvis) in integrirani prodajalci (npr. IBM in HP). Ponudniki tehnologije vključujejo ponudnike programske opreme za upravljanje fizične ali virtualne infrastrukture (npr. VMWare) ter ponudnike strojne opreme, ki je potrebna za delovanje infrastrukture. Slednje vključuje večina večjih ponudnikov strojne opreme (Intel ipd.).

1.3.2 Računalniško okolje kot storitev – PaaS

Srednji model v strukturi SPI je računalniško okolje (angl. *platform*) – PaaS, ki predstavlja razvojno okolje z množico orodij (npr. podatkovne baze, datotečni sistem, razvojne knjižnice itd.), ki razvijalcem programske opreme nudi možnost inženiringa, razvoja in integracije svoje aplikacije na oblaku (Rhoton & Haukioja, 2013, str. 21). Mather et al. (2009, str. 19) model PaaS označuje kot posebno različico SaaS modela, kjer je uporabnik modela razvijalec programskih aplikacij. Orodja, ki jih razvijalec uporablja preko brskalnika pa so storitev na oblaku (Ahson & Ilyas, 2011, str. 62–63).

PaaS pozna dve vrsti temeljne tehnologije. Prva je razvoj programske opreme, njeno testiranje in izvajanje na oblaku. Razvoj aplikacij za distribuirana okolja je kompleksen in zaradi hitrega povečevanja pasovne širine omrežij lahko to poenostavimo z dvema tipoma tehnologije. Prvi je uporaba omrežnih razvojnih orodij, kjer razvijalec lahko brez dodatne lokalne namestitve orodij in aplikacij razvije aplikacijo neposredno na oblaku z uporabo brskalnika ali aplikacije za oddaljen dostop, pri čemer se vsa orodja izvajajo v tej aplikaciji. Drug tip je, da lokalno nameščeno razvojno okolje integrira razvijajočo aplikacijo neposredno na oblak.

Druga vrsta temeljne tehnologije je obsežna porazdelitev operativnega okolja aplikacije. To se nanaša na razširljivo uporabo programskega vmesnega programja, podatkovni baz in datotečnega sistema distribuiranega med več fizičnimi strežniki. To omogoča, da

operativno okolje aplikacije ni omejeno ne eno samo fizično strojno opremo, temveč gre čez meje potrebnega in je navidezno neskončno veliko (Ahson & Ilyas, 2011, str. 63–64).

Mather et al. (2009, str. 21) pri primerjavi funkcionalnosti PaaS-a s tradicionalnim razvojnim okoljem navaja sledeče prednosti:

- večnajemna razvojna orodja: tradicionalna razvojna orodja so namenjena posamezniku (licenca je navadno pogojena na eno delovno okolje), v nasprotju s PaaS okoljem, ki nudi več uporabniški dostop, vsak z možnostjo več aktivnih projektov.
- Večnajemniška namestitvena arhitektura: razširljivost navadno ni predmet prvotnega razvoja programske opreme in je prepuščena sistemskim administratorjem v namestitveni fazi (produkcija). Pri PaaS je razširljivost programa, izenačevanje obremenitve (angl. *load balancing*) in nadomestni način delovanja (angl. *failover*), vgrajena v osnovno razvojno okolje.
- Integriran nadzor: pri tradicionalnem razvoju spremljanje delovanja aplikacije navadno ni prisotno, kar je pri PaaS že integrirano v razvojno okolje.
- Integrirano zaračunavanje: PaaS nudi različne mehanizme obračuna uporabe za SaaS.

Kljub prednostim, ki jih prinaša PaaS v primerjavi s tradicionalnim razvojnim okoljem je pri prehodu na oblak, v tem primeru na PaaS model, potrebno po priporočilih Badger et al. (2012, str. 6–5) paziti na sledeče karakteristike:

- generični vmesniki (angl. *generic interfaces*): preferirajo uporabo PaaS, ki zagotavlja dovolj generične vmesnike za prenosljivost in interoperabilnost med okoljem in razvojno aplikacijo,
- standardni programski jeziki in orodja: uporaba uveljavljenih programskih jezikov in orodij ima prednost pred lastno razvitim okoljem PaaS-a, v kolikor za to ni praktičnega razloga,
- dostop do podatkov: priporočajo standardni dostop do podatkov (npr. strukturiran povpraševalni jezik, angl. *structured query language – SQL*),
- varnost podatkov: preveriti je treba varnostne mehanizme in tehnologijo dela s podatki določenega ponudnika in pri tem paziti da zagotavlja zaupnost, skladnost, integritete in potrebe po razpoložljivosti za podjetje, ki bo potem uporabljajo razvito aplikacijo,
- programska arhitektura (angl. *application frameworks*): priporočena je uporaba arhitekture, ki vključuje tudi orodja in funkcionalnost za delo z varnostnimi komponentami,
- testiranje komponent: priporočeno je, da okolje vključuje knjižnice, ki zagotavljajo preverbe vedenja komponent tako s funkcionalnega kot tudi s zmogljivostnega vidika,
- uporaba varnosti: preveriti je treba, ali se v PaaS-u razvita aplikacija lahko integrira z morebitnim obstoječim varnostnim protokolom podjetja in ali lahko uporablja posebne varnostne komponente (npr. kriptografijo ipd.),

- brisanje podatkov: okolje naj bi omogočalo zanesljivo brisanje podatkov na zahtevo stranke.

1.3.3 Aplikacija kot storitev – SaaS

Pri tradicionalnem nakupu programske opreme kupec prejme aplikacijo in z njo povezano licenco. Običajno prvotnem nakupu sledi tudi nakup vzdrževanja, ki vključuje prejemanje posodobitev programske opreme in/ali drugih vzdrževalnih storitev. Celotna odgovornost za upoštevanje licenčne pogodbe ter za združljivost programske opreme in njenih posodobitev z operacijskim sistemom ali drugimi aplikacijami je na strani lastnika aplikacije (Marks & Lozano, 2010, str. 18).

SaaS kot najvišje ležeča plast storitvenega modela je namenjen končnim uporabnikom. Aplikacije temeljijo na spletni tehnologiji in so dobra alternativa namiznim aplikacijam iz več vidikov. Uporabnik ne kupi programske opreme, temveč jo najame. Uporaba aplikacije se obračuna kot naročnina ali po uporabi (angl. *pay-per-use*). V nekaterih primerih je uporaba aplikacija brezplačna, navadno z omejeno ponudbo funkcionalnosti. S tega vidika plačilo krije vse elemente najema, od strojne opreme in programske opreme do vzdrževanja (Mather et al., 2009, str. 18; Ahson & Ilyas, 2011, str. 3). Če bi iskali razlog za nastanek oblaka, bi bil to vsekakor SaaS (Marks & Lozano, 2010, str. 36).

Poleg zgoraj opisane razlike Mather et al. (2009, str. 18–19) navajajo še nekaj prednosti uporabe SaaS rešitve v primerjavi s tradicionalno in razlike med njima:

- ponudnik storitev ima kontrolo nad uporabo aplikacije, s čimer prepreči nepooblaščen uporabo ali nadaljnje kopiranje aplikacije.
- Centralizacija aplikacije ponudniku omogoča enostavno nadgradnjo in vzdrževanje, s tem pa zmanjšanje stroškov.
- S stališča uporabnika ni potrebna posebna strojna oprema, temveč le lahki odjemalci za spletno povezavo. Posebne sistemske nastavitve niso potrebne. Največji poseg lahko predstavlja le nastavitev požarnega zidu. Nekatere aplikacije je mogoče tudi delno prilagajati s pomočjo programskih vmesnikov (angl. *application program interface*, v nadaljevanju API).
- Kot najpomembnejšo razliko pa izpostavlja več najemniški model na SaaS strani v primerjavi z eno najemniškim (ali nakupnim) na tradicionalni strani. Pri SaaS modelu se uporaba strojne opreme deli med vse uporabnike in s tem zagotavlja njen boljši izkoristek.

V skladu z zapisanimi prednostmi in razlikami lahko zapišem tudi odgovor na večkrat zastavljeno vprašanje tako v praksi kot v literaturi: »Kakšna je razlika med modelom SaaS in rešitvami ponudnikov programskih rešitev (angl. *Application service providers*, v

nadaljevanju ASP)?« Port (2009) pojasni, da je tudi ASP aplikacijam, tako kot SaaS, je glavni cilj zmanjšati IT stroške podjetja. Podobnost se konča, ko modela pogledamo od bližje. Glavna razlika med njima je njuna organizacija. Pri SaaS vsi uporabniki v ozadju uporabljajo iste računalniške vire medtem, ko je pri ASP to strogo ločeno. ASP je torej enonajemniški model, v primerjavi s SaaS, ki je večnajemniški. Posledično je pri ASP modelu veliko več uporabniških prilagajanj in vzdrževanja ter s tem več stroškov. V nasprotju s tem je pri SaaS zaradi manjših stroškov tudi najemnina storitev občutno nižja. V prid SaaS-u je tudi tehnologija razvoja, ki je osredotočena na razvoj internetnih aplikacij, v nasprotju z ASP, ki temelji na modelu strežnik–odjemalec.

1.3.4 Ostale različice storitvenega modela računalništva v oblaku

V prejšnjih poglavjih opisani SPI modeli so najpogostejše uporabljeni modeli za delitev storitev v oblaku. Niso pa to edini modeli. Že v uvodu poglavja o storitvenem modelu sem pisal o modelu UCSB-IBM, ki pozna poleg znanih modelov SPI še dodatne tri modele, in sicer podatki kot storitev (angl. *data as a service* – *DaaS*), komunikacija kot storitev (angl. *communications as a service* – *CaaS*) in strojna oprema kot storitev (angl. *hardware as a service* – *HaaS*) (Ahson & Ilyas, 2011, str. 5). DaaS ponuja shrambo podatkov neodvisno od tipa shrambe, bodisi datotečna struktura ali podatkovna baza (Hastings, 2014, str. 3). CaaS Studt in Hofstader (2009) opredelita kot združitev socialnih omrežij, oblaka in pametnih telefonov, kateri nudijo komunikacijo na zahtevo preko glasovnih, tekstovnih ali multimedijskih sporočil. HaaS Bruce (2014) opredeli kot strojno opremo, ki jo uporabnik prejme kot opremo zakupljene storitve. Kot primer navaja strojno opremo (usmerjevalnik, internetni telefon, brezžični oddajnik...), ki je del nakupa ponudnika internetne storitve.

Tako v literaturi kot tudi pri ponudnikih oblakov zasledim različne modele storitev. Modeli lahko bolj podrobno ali na drug način delijo obstoječ SPI model, ga združujejo ali pa isti model drugače poimenujejo (primer je že opisan UCSB-IBM model). Veliko avtorjev dodaja nove ključne komponente, kot so upravljanje, varnost in kakovost (Mahmood & Hill, 2011, str. 9). Poleg SPI navajam še nekaj uporabljenih modelov, katerih podrobnosti tu ne bom opisoval (Hastings, 2014, str. 3; Linux magazine 2008; Ahson & Ilyas, 2011, str. 62):

- shramba kot storitev (angl. *storage as a service* – *STaaS*),
- varnost kot storitev (angl. *security as a service* – *SECaaS*),
- testno okolje kot storitev (angl. *test environment as a service* – *TeaaS*),
- IT kot storitev (angl. *IT as a service* – *ITaaS*),
- vse kot storitev (angl. *everything as a service* – *XaaS*).

Akronimi niso enolično določeni in najdemo pod istim akronimom tudi različne opredelitve. Kot ilustrativni primer navajam prej omenjeni DaaS modela UCSB-IBM, ki ga

Elgazzar, Hassanein in Martin (2014) opredelijo kot »Discovery as a Service«, Cisco in Citrix pa »Desktop as a Service« (Citrix – Desktops as a Service, 2016; Cisco – Desktops as a Service(Cloud-Based), 2016). Zaradi lažje predstave opis modela puščam zapisan v angleškem jeziku.

1.4 Izvedbeni modeli računalništva v oblaku

Izvedbeni modeli predstavljajo način dostave storitvenih modelov (SaaS, PaaS in IaaS) uporabnikom. Posameznik ali podjetja sama definirajo izvedbeni model glede na njihove poslovne, operativne ali tehnične potrebe (Mahmood & Hill, 2011, str. 6).

V naslednjih poglavjih povzamem opise izvedbenih modelov, ki jih opredeljuje NIST. Poleg teh pa velja omeniti še **vladni oblak** (angl. *government cloud*), ki je namenjen javnim ustanovam, kjer pogosto veljajo posebni z zakonom določeni pogoji in zahteve za varstvo podatkov (Höllwarth, 2009, str. 44).

1.4.1 Javni oblak

McDonald (2010, str. 5) opredeli javni oblak kot najemniško delitev infrastrukture, katera se deli med vse uporabnike določenega ponudnika oblaka. Ker viri fizično niso neskončni, ostaja omejitev pri porabi virov posameznega uporabnika glede na skupno porabo fizičnih virov. V kolikor ne gre za izredni nadstandardni dogovor, se med ponudnikom in stranko določi le minimalno potrebne vire. Dogovor se sklene v naprej in je del sporazuma o ravni storitve (angl. *service level agreement*, v nadaljevanju SLA). Sporazum določa, da so viri dostavljeni kupcu, ta pa nima nikakršnega vpliva na to, kako in na kakšen način bodo dostavljeni. Prav tako mu je onemogočena kakršnakoli kontrola nad viri. Vse napisano je v domeni ponudnika. To pa je razlog, da se poraja vprašanje o varnosti podatkov posameznika ali podjetja in vodi v to, da se vse več podjetij odloči za uporabo privatnega oblaka. Koulopoulos in Champy (2012, str.10) trdita, da se bosta na dolgi rok javni in privatni oblak združila v enega. Združeni oblak bo imel sposobnost zaščite pomembnih podatkov in se bo hkrati znal prilagoditi potrebam trga in novim poslovnim modelom.

Za prednosti tega modela Dr. Höllwarth (2012, str. 40) navede ugodnost za podjetja, ki ne želijo vlagati v lastni IT ter mobilnost, ki jo lahko podjetje doseže z zakupom celotnega delovnega prostora v oblaku. Kot slabost pa navaja izgubo nadzora in s tem prevzem tveganja in pomisleke glede skladnosti, varnosti, dostopnosti ter zmogljivosti. Zaključí, da je ključnega pomena natančno preveriti karakteristike ponudnika za vsak primer posebej.

Za prvega pravega ponudnika javnih oblakov za poslovne subjekte štejemo podjetje Salesforce.com, ki je z modelom SaaS ponudil rešitve za upravljanje odnosov s strankami (angl. *customer relationship management*, v nadaljevanju CRM). Pri tem je bilo veliko dvomov, da to ni izvedljivo, saj so za podjetja zelo pomembni podatki locirani izven

njihovih objektov. Enostavnost in plačilo po uporabi je bilo za podjetja tako privlačno, da je ponudnik doživel velik uspeh na trgu. Sledili so mu tudi ostali. S tem se je povečala zaupnost v ta model oblaka (Marks & Lozano, 2010, str. 15).

Po podatkih RightScale (2016), kjer so v anketo zajeli 1060 IT strokovnjakov (polovica iz podjetij z več kot 1000 zaposlenimi), je glede na izvedbo oblaka še vedno največ uporabnikov javnega oblaka, in sicer 89%, vendar v primerjavi s prejšnjim letom odstotek stagnira. Zajem podatkov je na poudarku modela IaaS.

1.4.2 Privatni oblak

Po trditvah Koulopoulos in Champy (2012, str. 10) je uporaba privatnega oblaka smotrna v primeru, ko je zaznati večjo potrebo po varnosti podatkov in kompromisi zaradi občutljivih podatkov niso sprejemljivi. V primeru uporabe te vrste oblaka Mather et al. (2009, str. 23) trdijo, da je vsa odgovornost za operativno izvajanje izključno na strani uporabnika oblaka.

Glavni atribut, ki opisuje privatni oblak je omejen dostop. Ta je omogočen le podjetju ali določeni osebi. Uporabnik, ki dostopa do takšnega oblaka ima nad njim vso potrebno kontrolo. To ne pomeni nujno, da je uporabnik izdelal podatkovni center ali da ga upravlja, temveč da ima precejšno kontrolo nad njim v skladu z najemniško pogodbo ali lastništvom gostujočih virov (McDonald, 2010, str. 5–6). Koulopoulos in Champy (2012, str. 24) ločita tri različice privatnega oblaka:

- namenski (angl. *dedicated*): infrastruktura se nahaja v prostorih uporabnika ali v kolokacijskem centru in jo opravlja interni IT oddelek,
- skupnost (angl. *community*): infrastruktura se nahaja v prostorih tretje osebe, lastništvo in upravljanje je pod okriljem ponudnika, v skladu s SLA (navadno individualna) in pogodbenimi določili,
- upravljavski (angl. *managed*): infrastruktura je v lasti uporabnika, upravlja pa jo ponudnik.

Zasebni oblak v primerjavi z javnim na eni strani omogoča individualne prilagoditve, boljšo varnost in skladnost, vendar so na drugi strani večji stroški nakupa ali vnaprejšnjega zakupa opreme (Höllwarth, 2009, str. 41).

V primerjavi z javnim oblakom, ki se ga v splošnem največ uporablja, velika podjetja, ki najemajo več kot 1000 navideznih naprav (angl. *virtual machine*, v nadaljevanju VM) po podatkih RightScale (2016) največ uporabljajo privatni oblak. Prav tako je v zadnjih letih rast uporabe privatnega oblaka za velika podjetja z veliko kapaciteto VM veliko večja v primerjavi z uporabo javnega oblaka.

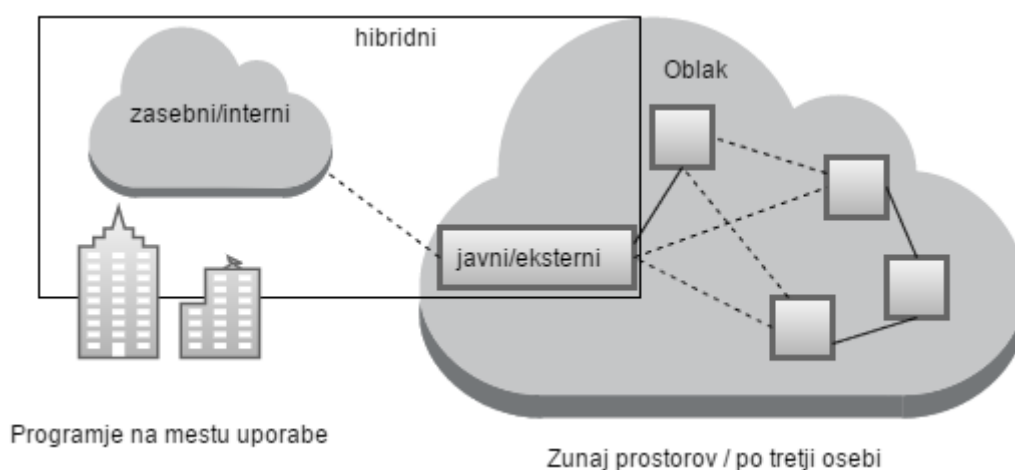
1.4.3 Hibridni oblak

Ta izvedbeni model je sestavljen iz ostalih izvedbenih modelov (privatni, javni in oblak skupnosti). S tem modelom lahko podjetje za manj pomembne zadeve uporablja storitve v javnem oblaku, za pomembnejše (npr. osnovno programsko opremo) pa uporabi privatni oblak (Mather et al., 2009, str. 25). Slika 3 prikazuje izvedbeni model hibridnega oblaka.

Höllwarth (2009, str. 42) za uspešne rešitve v hibridnem oblaku navaja dva ključna dejavnika:

- učinkovita integracija med izvedbenimi modeli, tako da uporabnik ne bo zaznal razlik ali prehodov in
- vzpostaviti uporabniku enotno okolje iz stališča upravljanja vseh izvedbenih modelov.

Slika 3: Struktura hibridnega oblaka



Vir: T. Mather et al., Cloud Security and Privacy, 2009, str. 25, slika 2.9.

Po podatkih RightScale (2016), uporaba hibridnega oblaka v zadnjih letih najbolj narašča. V zadnjem letu je uporaba hibridnega oblaka 71% in v primerjavi z letom prej je odstotek narasel za 13 odstotnih točk.

1.4.4 Oblak skupnosti

Izvedbeni model oblaka skupnosti je namenjen skupini ljudi ali organizacijam s skupnimi nameni ali potrebami (McDonald, 2010, str. 7).

Höllwarth (2009, str. 45) podaja ilustrativni primer uporabe tovrstnega izvedbenega modela: če uporabnik uporablja različne storitve, se mora pri tem večkrat identificirati v posamezni oblak, z uporabo oblaka skupnosti pa so te storitve dosegljive z isto

identifikacijo. Prav tako so posamezna področja prilagojena določenemu področju ali panogi. Kot primer navaja davčne svetovalce.

1.5 Predpogoj za izvedbo računalništva v oblakih

»Oblak ni toliko tehnologija, kot je kombinacija pred njim obstoječih tehnologij.« trdijo Mather et al. (2009, str. 11).

V nadaljevanju navajam nekatere pomembnejše tehnologije, ki jih oblak uporablja in brez katerih v takšni obliki kot je danes ne bi obstajal.

1.5.3 Virtualizacija kot eden najpomembnejših elementov računalništva v oblakih

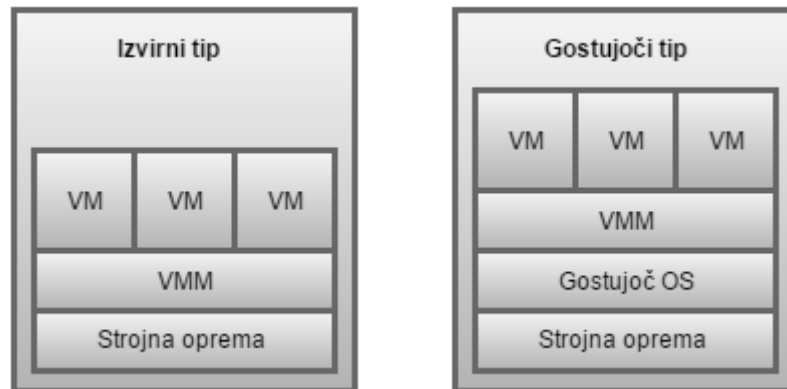
Najpomembnejša in ključna tehnologija za izvedbo oblaka je virtualizacija (Chorafas, 2011, str. 7; Park, 2012, str. 425). Glavni razlog za to je zagotavljanje elastičnosti oblaku (Babcock, 2010, str. 53).

Kot navaja Vaughan-Nichol (2006, str. 12) segajo začetki virtualizacije okoli leta 1965 kot plod sodelovanja podjetja IBM in raziskovalne univerze Massachusetts Institute of Technology – MIT. IBM je nato začel s prodajo prvih sistemov, ki so omogočali virtualizacijo.

Virtualizacija danes, kot jo opisuje Chorafas (2011, str. 48–49), preoblikuje fizične vire v več virtualnih okolij – naprav. S tem omogoča, da si VM delijo skupne fizične vire. Vsaka VM se izvaja ločeno od ostalih in na njej se izvaja svoj operacijski sistem (v nadaljevanju OS) ter svoje aplikacije. Prav tako je VM bistveno lažje spreminjati (dodati in odvzeti) virtualne karakteristike kot so zmogljivost procesorja, velikost pomnilnika in trdega diska kot, da bi to izvedli na fizični ravni strojne opreme. Kot pravi McDonald (2010, str. 2–4), se zaradi virtualizacije tako bistveno bolj poveča racionalizacija uporabe strojne opreme.

Za zagotavljanje varnosti podatkov posamezne VM pred vdorom druge VM, preprečevanje sočasnih dostopov dveh ali več VM do skupnega fizičnega vira ter za virtualno predstavo nižje ležečih komponent skrbi nadzornik virtualnih naprav (angl. *virtual machine manager* ali krajše *hypervisor*, v nadaljevanju VMM). Kot navajajo Rodríguez-Haro et al. (2012, str. 269–270) VMM deluje kot programska oprema neposredno nad strojno opremo in se imenuje izvirni (angl. *native*) ali gostujoči (angl. *hosted*), ki deluje v primarnem OS. Pri slednji različici se posamezna VM izvaja kot svoj ločen proces v primernem OS. Kot primer produktov VMM navajajo VMware ESX, Hyper-V in Xen za prvo različico ter VMware Server in VirtualBox za drugo. Slika 4 prikazuje strukturo obeh različic.

Slika 4: Struktura virtualizacije



Vir: F. Rodríguez-Haro et al., *A summary of virtualization techniques*, 2012, str. 270, slika 2.

Virtualizacija se ne uporablja le za strežnike, temveč tudi druge vire. Virtualizacija vira, ki ga oblak s pridom uporablja, je tudi omrežje. Ta uporablja več virtualnih omrežij (angl. *virtual network*) na eni sami fizični podlagi (angl. *network virtualization environment*; Chowdhurya & Boutabab, 2010, str. 862–863).

1.5.2 Ostali elementi za zagotavljanje obstoja računalništva v oblaku

Kot že napisano, oblak uporablja skupek ostalih tehnologij in metod, kjer so nekatere bolj, druge manj pomembne. Nekatere navadno celo povzemamo kot samoumevne. V tem poglavju bom poleg že navedenih (mrežno in storitveno računalništvo ter virtualizacija) naštel še nekaj elementov, ki so pomembni del obstoja oblaka kot ga poznamo danes.

Po Mather et al. (2009, str. 12–17) povzemam naslednje elemente oblakov:

- širok spekter v omrežje povezanih (lahkih) odjemalcev,
- vse bolj sofisticirani brskalniki, katere lahko namestimo na lahke odjemalce,
- širokopasovni dostop, s katerim je mogoče dosegati visoke hitrosti prenosa podatkov po omrežju,
- podatkovni centri in strežniške farme,
- vse cenejši in hitrejši mediji za shranjevanje podatkov, tudi preko omrežja (angl. *storage area network*),
- primerni programski vmesniki (API),
- storitveno usmerjena arhitektura (SOA).

1.6 Prihodnost računalništva v oblaku

Oblak ni več samo trend, temveč vpliva na zasnovo, kako podjetja oblikujejo in uporabljajo informacijske sisteme (Tang & Liu, 2015, str.60). Povečevanje fleksibilnosti in dejstvo, na kako enostaven način lahko z uporabo oblaka uporabniki dostopajo do računalniških virov in storitev, vplivata na odločitve podjetij za uporabo oblaka. Pri tem ima pojav tudi trajen učinek na IT (Koehler, Anandasivam, Dan & Weinhardt, 2010). Zaradi številnih koristi, ki jih omogoča uporaba tehnologije oblaka je ta postal zelo priljubljen in število uporabnikov se naglo povečuje. Med uporabniki niso le posamezniki in podjetja, temveč se povečuje tudi število vladnih oz. državnih organov.

Kot navajajo Garrison, Wakefield in Kim (2015, str. 377) po virih družbe Gartner iz leta 2011, bodo podjetja in vladni organi do leta 2020 uporabljali več kot polovico svojih IT storitev z uporabo tehnologije oblaka. Po besedah Sabi, Uzoka, Langmia in Njeh (2015, str. 183) bo oblak do tega leta postal standardni način uporabe računalniških virov in storitev in bo postal samoumeven. Iz tega naslova naj bi termin oblak (računalništvo v oblaku) izzvenel, tako kot se je to zgodilo s terminom osebno računalništvo (angl. *personal computing* – PC).

2 ANALIZA MODELOV ZA IZBIRO PONUDNIKA OBLAKA

Posledica naraščajočega povpraševanja po uporabi oblaka je hitra rast števila ponudnikov oblaka. Pri tem so poleg manjših in srednje velikih ponudnikov, ki navadno delujejo na lokalnem ali regijskem področju, vključene tudi velike multinacionalke, kot npr. IBM, Microsoft, Google in Amazon, ki imajo svoje vire razpršene na več celinah in pokrivajo vsa razvita področja našega planeta. Izbira ponudnika, ki bi izpolnila vse zahteve kakovosti storitve (angl. *quality of service*, v nadaljevanju QoS) na tako velikem in ne transparentnem področju kot je oblak, je izredno zahtevna in kompleksna (Garg, Versteeg, & Buyya, 2013, str. 1012; Repschläger, Wind, Zarnekow, & Turowski, 2013, str. 2–3). Heterogena ponudba oblaka se izraža v podobnih storitvah, ki jih nudijo različni ponudniki, pri tem pa se uporabijo različne metode obračuna stroška, storitev deluje na različnih ravneh zmogljivosti ter s precej različnimi in drugačnimi značilnostmi (Garg et al., 2013, str. 1012; ur Rehman, Hussain, & Hussain, 2011, str. 45). Na drugi strani kompleksnost izbire ponudnika zvišuje že sama uvedba nove tehnologije, v tem primeru oblaka, saj je navadno relativno kompleksna, pri tem pa pri kupcu vzbuja nemalo dvomov in vrsto različnih možnosti (Luoma & Nyberg, 2011, str. 5). Da bi bilo podjetjem, vladnim ustanovam, neformalnim organizacijam ali posamezniku lažje pri odločevanju, trg potrebuje odločitveni model za pomoč pri izbiri ustreznega ponudnika oblaka (Liu, Chan, & Ran, 2016, str. 2).

V naslednjem poglavju opisujem ugotovitve raziskave in analiziranja obstoječih modelov, ki se navezujejo na vprašanja, kakšno arhitekturo izbrati, katero izvedbo oblaka izbrati

glede na potrebe podjetja ter katerega ponudnika oblaka izbrati. Pri tem ni nujno, da model neposredno odgovarja na zastavljeno vprašanje, temveč je lahko le pomemben člen, ki pripomore pri iskanju glavnega odgovora. Na primer pri izbiri ponudnika oblaka za pomemben člen štejem opredelitev kriterijev in lastnosti, določanje njihove pomembnosti, način, kakovost in verodostojnost merjenja lastnosti (objektivnih in subjektivnih) ter uporaba različnih metod. Za raziskavo sem uporabil strokovno literaturo, predvsem znanstvene in strokovne članke, konferenčne prispevke ter raziskave, ki so dostopne v spletnih knjižnicah.

Pri analizi modelov sem pozornost posvetil tudi metodološkemu pristopu, ki so jih uporabili avtorji. Same modele sem razdelil po namenu in po vrsti storitve oblaka, za katerega je bil model izdelan. Pri tem je lahko model namenjen ali specifični storitvi kot so SaaS, PaaS ali IaaS ali splošni uporabi, kar pomeni, da zavzema poljubno storitev oblaka.

2.1 Predstavitev obstoječih modelov

Kot ugotavljajo Sun, Ma, Zhang, Dong in Hussain (2016, str. 42 in 44), je že nekaj let vprašanje izbire ponudnika storitev v oblaku predmet obravnave raziskav tako v akademskem svetu kot tudi v gospodarstvu. Pri reševanju zastavljenega vprašanja se uporablja različne metode, s katerimi si želimo, da bi kar najbolj ustrezno izpolnili zahteve posameznika oz. podjetja in našli najbolj ustreznega ponudnika in s tem tudi samo storitev. Vendar pri tem velja upoštevati, da univerzalnega odgovora ni, saj določena storitev, ki ustreza potrebam določenega podjetja, ni nujno dobra ali celo ne zadošča zahtevam drugega podjetja. Pri tehnologiji oblaka je to toliko bolj očitno, saj imamo na trgu ogromno podobnih storitev z nekaterimi manjšimi odstopanji, za katere pa ponudniki nudijo različne ravni dostopa, drugačne načine obračuna uporabe virov, heterogene pogodbe, za enake stvari pa uporabljajo različno terminologijo. Razlike se pojavijo celo pri posameznem ponudniku. Garg et al. (2013, str. 1012) kot primer navajajo Amazon EC2, ki ponuja enake kapacitete, katerih cena je različna za različne regije.

Podobna vprašanja o izbiri so se pojavila tudi že pred oblakom, in sicer za potrebe mrežnega računalništva in računalništva v grupah. Vendar kot navajajo ur Rehman et al. (2011, str. 45), je uporaba teh modelov za izbiro oblaka neprimerna in le navidezno zadošča tovrstni izbiri.

Reševanje vprašanja, ali preiti na tehnologijo oblaka in vprašanja izbire ponudnika se je z razvojem oblaka hitro večalo. Tako je mogoče prve modele zaslediti že zgodaj. Sprva so bili modeli izdelani za individualne izzive podjetij in za specifično poslovno področje, navadno z veliko mero prilagodljivosti določenemu izzivu. Zaradi tega modeli niso bili splošno uporabni. Modela narejen za eno podjetje ni bilo mogoče uporabiti na drugem podjetju, ne da bi ga pri tem modificirali v takšni meri, da lahko govorimo že o novo nastalem modelu. Šele v zadnjih nekaj letih je možno zaslediti modele, ki so namenjeni

širši množici podjetij kot npr. za celotno panogo, določeno organizacijsko strukturo ali za določeno storitev (npr. SaaS). Še korak dlje predstavljajo modeli, ki so namenjeni splošni uporabi za vse vrste storitev. Pri slednjih je navadno treba opraviti manjše modifikacije, da so uporabni za določen namen. V literaturi novejšega datuma zasledim osnovna ogrodja modelov. Ti služijo kot osnova za izdelavo lastnega modela, podajajo smernice in priporočila, kako iz določenega ogrodja sestaviti model po meri določene težave.

V Tabela 1 navajam seznam modelov, katere sem na podlagi strokovne literature vključil v raziskavo in jih analiziral. Tabela poleg vira navaja še uporabljeno metodologijo pri izdelavi modela ter vrsto storitve, za katero je bil model razvit. Model, ki je bil poleg specifične vrste (SaaS, IaaS...) razvit za splošno uporabo, v preglednici poimenujem »splošno«. V literaturi je poleg napisanega seznama mogoče najti še veliko ostalih modelov, vendar jih zaradi časovnih omejitev nisem uspel analizirati. Opazil sem, da je veliko modelov različica obstoječih, katere so razvili isti avtorji oz. nekateri med njimi v novi sestavi.

Tabela 1: Seznam obstoječih modelov in uporabljena metodologija

Vir	Metodologija	Storitev
Baranwal & Vidyarthi (2014)	Ranked Voting Method	IaaS
Chang, Liu, & Wu (2012)	Dinamično programiranje in PA	splošno
Chang & Wills (2016)	OSM	DaaS
Chen, Yan, Zhao, Lee, & Singhal (2012)	Teorija logičnega sklepanja	splošno
Dastjerdi, Tabatabaei, & Buyya (2010)	Teorija logičnega sklepanja	IaaS
Garg, Versteeg, & Buyya (2011)	AHP	splošno
Garg et al. (2013)	AHP	IaaS
Ghezzi, Panzica La Manna, Motta, & Tamburrelli (2015)	Strategija izbire	SaaS
Godse & Mulik (2009)	AHP	SaaS
He, Han, Yang, Grundy, & Jin (2012)	PA	SaaS
Jung et al. (2013)	PA	PaaS
Karim, Ding, & Miri (2013)	AHP	IaaS, SaaS
Kragelj & Rajkovič (2012)	Več-parametrna metoda DEX	IaaS
Kwon & Seo (2014)	AHP	IaaS
Lee (2014)	AHP	splošno
Limam & Boutaba (2010)	MAUT	SaaS
Liu et al. (2016)	SAW, TOPSIS, AHP	splošno
Martens, Teuteberg, & Gräuler (2011)	MCDM	splošno
Martens & Teuteberg (2012)	Optimizacija	splošno
Menzel, Schönherr, & Tai (2013)	ANP	splošno

se nadaljuje

Tabela 1: Seznam obstoječih modelov in uporabljena metodologija (nad.)

Vir	Metodologija	Storitev
Ngan & Kanagasabai (2012)	MCDM	splošno
Quinton, Romero, & Duchien (2014)	Ontološka preslikava	PaaS
Repschläger et al. (2013)	AHP	splošno
Saripalli & Pingali (2011)	SAW	splošno
Siegel & Perdue (2012)	MCDM	splošno
Silas, Rajsingh, & Ezra (2012)	ELECTRE	splošno
Sun et al. (2016)	MCDM	splošno
Sundareswaran, Squicciarini, & Lin (2012)	PA in drevesna struktura	splošno
Tang & Liu (2015)	Analiza kriterijev	SaaS
ur Rehman, Hussain, & Hussain (2013)	MCDM	IaaS
Wittern, Kuhlenkamp, & Menzel (2012)	MCDM	splošno
Yang, Lin, & Dou (2013)	SAW in MDP	SaaS
Ye, Bouguettaya, & Zhou (2012)	Dinamično programiranje	IaaS, SaaS
Yu (2014)	Faktorizacija matrik	splošno
Zeng, Zhao, & Zeng (2009)	MAUT	splošno
Zhao, Ren, Li, & Sakurai (2012)	SAW	splošno
Zheng, Wu, Zhang, Lyu, & Wang (2013)	PA	SaaS

Legenda: DaaS – podatki kot storitev (angl. *Data as a Service*)
 OSM – angl. *organizational sustainability modeling*
 MAUT – angl. *multi-attribute utility theory* (v nadaljevanju MAUT)
 PA – požrešni algoritem
 TOPSIS – angl. *technique for order of preference by similarity to ideal solution* (v nadaljevanju TOPSIS)
 SAW – angl. *simple additive weighting* (v nadaljevanju SAW)
 MCDM – angl. *multiple-criteria decision making* (v nadaljevanju MCDM)
 ANP – angl. *analytic network process*
 ELECTRE – angl. *elimination and choice expressing reality*
 MDP – angl. *markov decision process*

Vir: L. Sun, H. Dong, F. K. Hussain, O. K. Hussain in E. Chang, Cloud service selection: State-of-the-art and future research directions, 2014, str. 136, tabela 1.

Kot že prej napisano, so bili prvi modeli izdelani za specifičen problem, katerega so reševali. Namenjeni so bili določenemu področju in navadno specifičnemu podjetju. Takšen je tudi, gledano iz kronološkega vidika, eden izmed prvih modelov, ki sem jih našel in vključil v svojo raziskavo, katerega avtorja sta Godse in Mulik (2009, str. 155–156). Model služi kot pomoč pri izbiri ponudnika za SaaS. Razvoj modela temelji na študiji posameznega primera, in sicer avtomatizaciji prodaje (angl. *sales force automation*). Temu primerno so prilagojeni tudi kriteriji, ki opredelijo izbiro ponudnika storitev. Kriterije sta avtorja določila s pomočjo izkušenj in intervjujev strokovnjakov IT. Kriterije sta glede na lastnosti razdelila v naslednje skupine:

- funkcionalnost, kjer so zajeta funkcionalna področja;
- v skupino arhitekture sta uvrstila lastnosti integracije sistema z ostalimi sistemi, skalabilnost in zanesljivost, kjer določata sistemu obvladovanje prometa ob konicah, njegovo dosegljivost ter varnost, katerega izpostavljata kot enega najbolj ključnih kriterijev pri SaaS produktih in se sklicujeta na ISO 27000 standard (skupen standardov, ki opredeljuje varnostne mehanizme);
- uporabnost sistema, ki se osredotoča na uporabniški vmesnik in pomoč ter delovanje na pametnih napravah, tudi v primeru nepovezanosti s spletom;
- uveljavljenost ponudnika na trgu glede na prepoznavnost in reference;
- stroški, kjer primerjata enkratni strošek nakupa sistema v primerjavi z letno najemnino sistema.

V svojem odločitvenem modelu sta uporabila večparametrsko odločitveno metodo AHP. Model v glavnem temelji na uporabniški subjektivni oceni (Qu et al., 2015, str. 381). Slednje ugotavljata tudi avtorja in navajata da je zaradi subjektivnosti modela, določanje in primerjava kriterijev premalo stroga (Godse & Mulik, 2009, str. 158).

Podoben model, ki je bil predmet moje analize, prav tako uporablja eno izmed metodologij MCDM – metodo DEX in je plod domačih avtorjev Kraglja in Rajkoviča (2012, str. 244–249). Model obravnava področje IaaS in pri tem vključuje tudi prehod IT sistema na oblak. Avtorja pri tem navajata: »Za lažjo uporabo oz. prehod je treba tudi osvojiti potrebno raven sposobnosti dojemanja sprememb, ki prinašajo tudi možnost dviga učinkovitosti zaposlenih v podjetju [...]«. Kriterije sta avtorja uvrstila v tehnični sklop, sklop vidika organizacije in ostalo, kjer se nahajajo druga dodatna merila (avtorske pravice, zakonodaja idr.). Odločitveno skupino sestavljajo vodilni člani (navadno na IT področju), kjer dopuščata možnost sodelovanja zunanjih neodvisnih sodelavcev. Podobno kot pri prejšnjemu primeru, je tudi v tem modelu pomanjkljivost subjektivni vpliv odločitvene skupine ter, kar navajata tudi avtorja, da je za dokončno odločitev potreben človeški faktor.

Poleg opisanih modelov, kjer je prisotna subjektivnost, poznamo tudi bolj objektivni pristop k izdelavi modelov (Qu et al., 2015, str. 381). Kot primer pristopa izdelave modela z objektivnimi kriteriji so Zheng et al. (2013, str. 1213–1214) predstavili QoS odločitveno ogrodje in ga poimenovali CloudRank. Njihov pristop opredeli različne lastnosti QoS, pridobljene iz dosedanjih izkušenj uporabe, kjer združuje podobne uporabnike skupaj. Pri tem kredibilnost uporabnikov ni preverjena. Na podlagi podobnih vrednosti, z uporabo dveh algoritmov (CloudRank1 in CloudRank2), kategorizirajo storitve za vsakega uporabnika. Prvi algoritem paroma razvrsti storitve glede na preference končnega uporabnika. Z drugim algoritmom se določi zanesljivost preferenc za celotni seznam. Ocena zanesljivosti vključuje razlike med uporabniškimi preferencami.

Omenil bi še en model oz. ogrodje, ki je osredotočen na uporabnika in primarno zadeva uporabnikove osebne preference in izkušnje QoS. Avtor Yu (2014) je model poimenoval CloudRec. Osnova ogrodja je matrično razvrščanje QoS neznanih storitev oblaka. Uporabil je iterativni algoritem za optimizacijo strukture kriterijev nepopolnih QoS podatkov. Z uporabo matrične trojne delitve (angl. *matrix tri-factorization*) je izdelal razrede uporabnikov in storitev, kjer se uporabi tako zgodovina QoS kot tudi relacije med uporabniki in storitvami. S tem odpravljamo težavo tako različnih tipov podatkov kot tudi manjkajočih podatkov (Sun, Dong, Hussain, Hussain, & Chang, 2014, str. 142).

Ur Rehman et al. (2013) so izdelali MCDM model za izbiro IaaS. Pri tem uporabljajo QoS zgodovino, ki je razdeljena po časovnih režah. Za vsako režo se sprva določi najbolj optimalen ponudnik oblaka, nato se s pomočjo agregacije med njimi določi najbolj primerne ponudnika za celotno časovno premico.

Model, ki prav tako temelji na mehkih funkcionalnih in nefunkcionalnih zahtevah uporabnikov, so izdelali Sun et al. (2016). Model so poimenovali Cloud-FuSeR. Sprva so s pomočjo kalkulacije na podlagi funkcionalnih zahtev uporabnikov razvrstili podobne storitve skupaj. S pomočjo AHP metode so ovrednotili nefunkcionalne zahteve uporabnikov, katere so največkrat nejasne in nenatančne. Na koncu so s pomočjo tehnike za določanje prednostnega seznama po podobnostih do popolne rešitve (angl. *Technique for Order of Preference by Similarity to Ideal Solution*, v nadaljevanju TOPSIS) kategorizirali kandidate, upoštevajoč rezultate funkcionalne in nefunkcionalne razdelitve.

Poleg naštetih so na podlagi uporabniških izkušenj nastali še nekateri drugi modeli, kot npr. Baranwal in Vidyarthi (2014), Dastjerdi et al. (2010), Garg et al. (2011, 2013), He et al. (2012), Karim et al. (2013).

Pri tem Qu et al. (2015, str. 381) opozarjajo, da so modeli, ki temeljijo na QoS kriterijih neučinkoviti pri ocenjevanju tako kompleksne in dinamične funkcionalnosti oblaka. V ta namen je bilo zasnovanih veliko primerjalnih preizkusov (angl. *banchmark*) za specifične kvantitativne lastnosti storitev v oblaku. Kot primer navajam model avtorjev Chang et al. (2012), ki zajema merljive funkcijske in stroškovne kriterije. Merljivost so natančno opredelili z matematičnimi enačbami, kjer kot rezultat vrnejo množico ponudnikov.

Model, ki je v raziskavi najmlajši, je plod avtorjev Liu et al. (2016) in je skupek širokega spektra kriterijev in uporabljenih metodologij. Poudarjajo, da so sekundarni kriteriji (npr. konkurenčni pritiski ali vodstvene sposobnosti) vse preveč zapostavljeni. Model vključuje objektivne kriterije (npr. stroški,...), kot tudi subjektivne kriterije (npr. tehnologija, okolje,...) in je po njihovih trditvah nov pristop k reševanju vprašanja o izbiri pravega ponudnika oblaka. Kot zanimivost velja omeniti, da so v svoji raziskavi uporabili nadpovprečno število različnih metodologij (MAGDM (angl. *multi-attribute group decision-making*), SV (angl. *statistical variance*), SAW, Delphi-AHP, TOPSIS idr.).

Model oz. v tem primeru ogrodje, ki ga vsekakor velja opisati, so razvili v Cloud Services Measurement Initiative Consortium (v nadaljevanju CSMIC; Siegel in Perdue, 2012, str. 411). Gre za standardno metodo merjenja in primerjave poslovnih storitev neodvisno od tega ali so storitve nahajajo znotraj ali zunaj podjetja. Metodo so poimenovali Service Measurement Index (v nadaljevanju SMI), ki je skupek poslovnih indikatorjev (angl. key performance indicator, v nadaljevanju KPIs) (CSMIC, 2014, str. 2). Ogrodje je bilo prvič predstavljeno septembra 2011 in kasneje nekajkrat dopolnjeno. Trenutna različica 2.1 je bila nazadnje spremenjena julija 2014 (Monteiro & Vasconcelos, 2013, str. 265). Kot pišejo Garg et al. (2013, str. 1013), so SMI kriteriji izdelani na podlagi standardov Mednarodne organizacije za standardizacijo (angl. *International Organization for Standardization*, v nadaljevanju ISO). Navajajo tudi, da gre za celovit pogled QoS, ki ga potrebuje uporabnik za izbiro ponudnika oblaka. V ta namen so opredelili 7 skupin kriterijev: odgovornost, prilagodljivost, kakovost storitve, stroški, učinkovitost, varnost in zasebnost ter uporabnost. Vsaka skupina vključuje več kriterijev, kjer vsak vsebuje množico KPIs, ki definirajo podatke, ki so potrebni za merjenje določenega kriterija. Nekateri od njih so specifični glede na vrsto storitve (SaaS, IaaS...), drugi so splošni za vse storitve (Siegel & Perdue, 2012, str. 413). V literaturi sem našel nekaj primerov uporabe ogrodja SMI, ki je služil kot izhodišče izdelavi nekega modela. Nekaj primerov, ki se sklicujejo na to ogrodje: Siegel in Perdue (2013), Garg et al. (2013), Monteiro in Vasconcelos (2013), Kourtesis, Alvarez-Rodríguez in Paraskakis (2014).

Obravnavani modeli so najpogosteje uporabili metodologijo več parametrskega odločanja – MCDM, in sicer metodo AHP. Poleg omenjene metodologije se pogosto pojavijo tudi različne vrste požrešnih metod (angl. *greedy algorithm*), lahko v kombinaciji s kakšno drugo metodo. Na tretje mesto po številu uporabljenih metod postavljam logični opis (angl. *logic-description logic*). Ostale metode se pojavijo posamično ali v nekaj primerkih.

2.2 Pomanjkljivosti obstoječih modelov

Analiza razpoložljivih modelov kaže, da je v zadnjih nekaj letih razvitih veliko modelov, ki so na nek način dotaknejo oblakov oz. storitev v oblaku. Prvotni modeli se najpogosteje osredotočajo bolj na specifičen problem specifičnega podjetja ali panogo poslovanja – turistične agencije, upravljanje odnosov s strankami ipd. Kasneje so bili razviti modeli, ki so se osredotočili na splošno izbiro ponudnika oblaka za vse vrste storitev, npr. za IaaS, Paas in SaaS. Tovrstnih modelov je v literaturi največ, manj se jih osredotoči na specifično storitev. Med slednjimi je pričakovano največ modelov, ki so izdelani izključno za SaaS, najmanj pa modelov osredotočenih na PaaS (Sun et al., 2014, str. 146–147).

Modeli, ki so namenjeni splošni uporabi so na prvi pogled najbolj uporabni. Podrobnejša analiza pokaže, da unija kriterijev, ki so specifični le za posamezno vrsto storitve ni tako učinkovita kot delitev glede na vrsto. Kriterija delovanje storitve na pametnih telefonih ali

uporabniški vmesnik storitve je smotrna predvsem pri modelu namenjenemu SaaS in zanemarljiva pri IaaS ali PaaS.

Glede na naravo kriterijev so najpogostejše obravnavani kriterije povezani s stroški, predvsem sama cena najema in model obračuna. Stroškom sledi kakovost storitve in varnost. Prvotna je pogosto razdeljena na podkriterije, tako kvantitativno merljive kriterije, kot tudi kriterije, katere je možno vrednotiti le s subjektivno oceno uporabnikov, na podlagi njihovih izkušenj. Primer delitve tega sklopa sta zanesljivost in zmogljivost storitve, kamor sodijo razpoložljivost, odzivnost itd..

Pri analizi obstoječih modelov v splošnem zaznavam naslednje večje pomanjkljivosti:

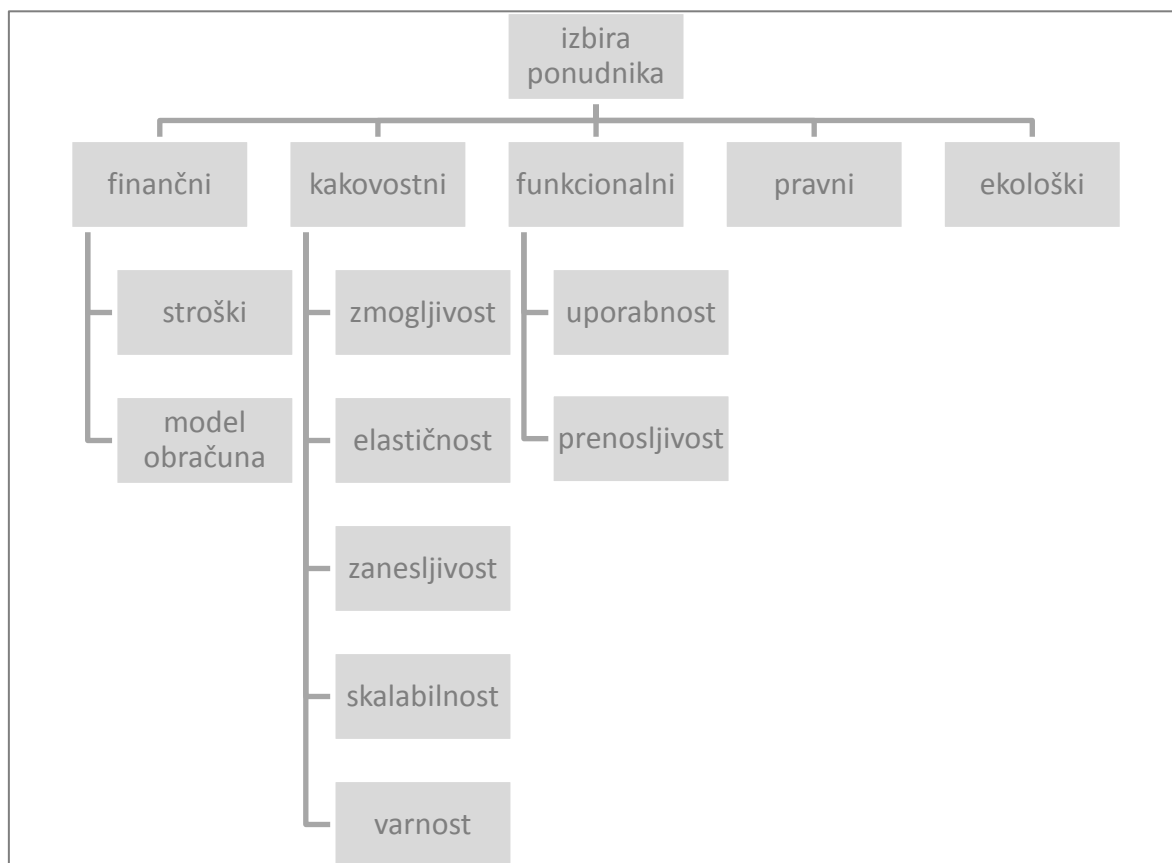
- raven abstrakcije je pri mnogih modelih tako nizka, da je določen kriterij razdeljen v zelo podrobne enote, ki pa zahtevajo od uporabnikov veliko mero strokovnega znanja. Ta lastnost se najbolj odraža pri določanju kriterijev kakovosti in funkcionalnosti storitve. Deljenje ni nujno slabo, kljub temu, da oteži razumevanje modela, vendar mora biti model sestavljen v odvisnosti od tega, kakšna je raven znanja osebe, ki vrednoti posamezne kriterije. To posebej velja za kriterije, ki se jih ne da izraziti kvantitativno, temveč le s subjektivnim mnenjem osebe, ki to vrednoti.
- Osredotočanje modelov na enonajemniško strukturo (Sun et al., 2014, str. 148). V praksi je uporaba oblaka najpogostejše povezana z večnajemniško strukturo, kar za sabo prinese dodatne omejitve, prednosti in slabosti. Za potrebe IaaS v ospredje postavljam delitev skupnih virov strojne opreme. Menim, da je nujno potrebno opredeliti kriterije, ki se navezujejo na skupno delitev virov kot npr. možnost skalabilnosti sistema. Slednje je ključnega pomena za sisteme, katerih zavrnitev zahteve uporabnika bi pomenila veliko poslovno izgubo. S stališča uporabnika zavrnjena zahteva pomeni, da storitev ni razpoložljiva.
- Modeli obravnavajo kriterije brez njihove morebitne odvisnosti. V model je potrebno vključiti njihovo soodvisnost. Primer soodvisnih kriterijev sta odškodnina v primeru kršitve pogodbe in dominanca ponudnika na trgu (Saripalli & Pingali, 2011, str. 322).

3 OPREDELITEV KRITERIJEV

V naslednjem poglavju naštejem in opišem kriterije, ki so vključeni v analiziranih modelih literature iz prejšnjega poglavja. Izmed njih sem uporabil tiste, za katere menim, da so bolj pomembni pri odločanju za izbiro ponudnika in hkrati ne povečujejo kompleksnosti modela do te mere, da uporabniku ne bi bil razumljiv. S premišljeno izbiro kriterijev želim doseči ravnovesje med razumljivostjo modela in njegovo učinkovitostjo. Pri tem pazim, da raven znanja uporabnika za razumevanje kriterija ne zahteva posebnega specialističnega poznavanja določenega področja temveč le toliko, kolikor je to nujno potrebno za razumevanje doseganja cilja. S tem omogočim vključevanje tako inženirjev kot vodilnega

kadra. Kriterije sem razvrstil glede na njihove lastnosti, in sicer na finančne, kakovostne, funkcionalne, pravne in ekološke kriterije (Slika 5).

Slika 5: Struktura kriterijev pridobljenih iz literature



3.1 Finančni kriteriji

Finančni kriteriji so najpogostejše obravnavani kriteriji pri tovrstnih modelih. Sprva opišem stroške in njihovo sestavo, nato opišem še nekatere modele obračuna storitev.

3.1.1 Stroški

Iz ekonomskega stališča podjetja je ključnega pomena zmanjševanje stroškov informacijskih sistemov (Wang, Lin, & Huang, 2008, str. 6). Kriteriji stroškov so poleg tehnoloških najpogostejše uporabljeni v tovrstnih modelih, zato se jim posveča veliko pozornosti. Po mnenju Liu et al. (2016, str. 1) celo preveč oz. se s tem zapostavlja ostale kriterije.

Dober primer, kjer pomembnost stroškov prednjači, je model za SaaS izbiro ponudnika, avtorjev Godse in Mulik (2009, str. 156), kjer stroške razdelita v dve skupini: letna

najemnina storitve in enkratni stroški, ki nastanejo pri implementaciji storitve. Pri tem stroške vzdrževanja in strojne opreme vključujeta v letno najemnino. Stroške začetnih nastavitvev in konfiguriranja ter svetovanja pa vključujeta v začetni enkratni strošek. Letno najemnino postavita kot najpomembnejši kriterij, takoj za njim pa še enkratni strošek. Pomembnost teh dveh kriterijev izstopa od pomembnosti ostalih kriterijev. Podoben je model za izbiro CRM ponudnika, avtorja Lee (2014, str. 57). Finančno korist, ki vsebuje začetne stroške implementacije, stroške vzdrževanja in plačevanje po uporabi, označuje kot najvplivnejši kriterij za izbiro CRM ponudnika.

Celotni kriterij stroška sestavlja več parametrov, kjer so eni bolj transparentni, drugi manj. Nekateri deli celotnega stroška so omenjeni že v prej opisanim modeloma. Povzeto po analizirani literaturi lahko celotni kriterij razdelim na parametre, kot si sledijo v nadaljevanju.

Začetni strošek implementacije vključuje strošek ponudnika za izvedbo inicialnih nastavitvev in konfiguracij ter strošek, ki nastane na strani podjetja samega. Slednje vključuje izobraževanje zaposlenih, morebitno migracijo obstoječega sistema, transformacijo delovanja IT oddelka ter svetovanje s strani ponudnika ali drugega zunanjega izvajalca.

Najemnina, katera vključuje tudi **vzdrževanje in nadgrajevanje** s strani ponudnika, se določa glede na količino uporabljenih računalniških virov in dodatnih aktivnosti, ki jih najamemo (varnostne kopije ipd.). Pri najemnini je treba biti pozoren, saj njena sprememba lahko pomeni tudi spremembo kakovosti same storitve (Höllwarth, 2012, str. 199).

Za določanje najemnine je potrebno v naprej oceniti predvidene količine uporabljenih virov in določiti nekatere attribute, ki vplivajo na to (npr. število uporabnikov ipd.). Prav tako je treba opredeliti vse potrebne in želene aktivnosti, ki bodo spremljale najem storitve (npr. varnostne kopije, nadzori dostopa in revizije ipd.)

Zaradi različnih razlogov lahko pride do potrebe po kasnejši zamenjavi ponudnika. V ta namen je treba oceniti višino stroškov, ki bi nastavi za izvedbo **prehoda na drugega ponudnika**. Ta ocena vključuje le stroške na strani obstoječega ponudnika in interne stroške podjetja zaradi migracije sistema, ne pa tudi stroškov inicialne vzpostavitve pri drugem ponudniku. Slednje spada v stroške iz prve skupine.

3.1.2 Modeli obračuna storitve

Modeli obračuna storitve oz. najemnine se razlikujejo v odvisnosti od vrste storitve, področja kateremu je storitev namenjena in od ponudnikov. Modeli se spreminjajo in se zaradi konkurenčnosti prilagajajo trgu. Sama primerjava je pogosto otežena zaradi različnih modelov obračuna.

Höllwarth (2012, str. 199) te modele razdeli v tri skupine glede na lastnost:

- brezplačni – navadno omejen obseg storitev z možnostjo plačljive nadgradnje, primerno za zasebnike;
- po uporabniku – praviloma poimensko dodeljeni računi;
- po obsegu – po dejanski uporabi virov (npr. pomnilnika ipd.).

K temu Repschläger et al. (2013, str. 5) dodajajo še model obračuna, ki temelji na časovnem najemu virov in koncept vnaprejšnje rezervacije (angl. *booking*).

Glede na definicijo oblaka bi lahko rekel, da je edini »pravi« model obračuna **po obsegu** (angl. *pay as you go*). Model prihaja iz telekomunikacij in je predvsem poznan pri plačevanju telefonskih pogovorov. Obračun pogovora se vrši na podlagi časa trajanja pogovora. Pri oblaku je obračun odvisen od števila ali količine uporabljenih virov. Kot primer navajam IaaS, kjer povzamem, da je enota instanca, katera deluje kot minimalni potreben vir za obdelavo prometa. V primeru povečanja prometa se število instanc povečuje in s tem obračun navezuje na število porabljenih instanc za obdelavo celotnega prometa. Povečevanje številavirov (v mojem primeru instanc) je z uporabniškega vidika vidno kot neskončno veliki viri. V praksi se je pokazalo, da prav zaradi te »neskončnosti« virov najemniki plačujejo več, kot so predvideli. Razlogov zato je več (Armbrust et al., 2009, str. 10–11):

- neracionalna uporaba virov je vsekakor ena od glavnih slabosti, kadar ni zgornje omejitve nad njimi. Sistemi, ki uporabljajo najeto infrastrukturo zasedajo več virov, kot je to potrebno. Z boljšim testiranjem in razvojem sistemov bi porabo virov mnogokrat zmanjšali.
- Čas, potreben za sprostitev virov, je največkrat predmet plačila najema oblaka. Viri se v trenutku, ko jih ne potrebujemo več, ne sprostijo, kar pomeni, da se plačuje vire, ki jih ne uporabljamo.
- V trenutku, ko se izvede zahteva po dodatnih virih, se ti povečajo za eno enoto (npr. za instanco). Torej se strošek najema poveča za velikost dodanega vira (npr. instance), kljub temu, da je uporaba novo dodanega vira minimalna. Torej imamo v nekem trenutku na voljo več virov, kot jih dejansko potrebujemo za obdelavo opravila (angl. *on-demand overspending*). Lahko rečem, da je enota za obračun najemnine preobširna. Enako je pri telefoniji, kjer je razlika med minutnim in sekundnim obračunom telefonskega pogovora. Primer: pri IaaS bi bilo potrebno obračunati najemnino po manjši enoti, kot je npr. instanca. Eden od kriterijev je dejanska poraba procesorskega časa glede na najemnika, neodvisno od tega koliko instanc in procesov ga uporablja. Podobno bi lahko napisal za ostale vire, vendar za to obstajajo tudi tehnične omejitve.

Najbolj tipične modele obračuna povzemam v preglednici Tabela 2, in sicer glede na vrsto storitev, ločeno za SaaS, PaaS in IaaS.

Tabela 2: Modeli obračunov za oblak

Storitev	Model obračuna
SaaS	število uporabnikov
	časovna poraba vira
	število izvedb
	število obdelanih zapisov
	poraba pasovne širine omrežja
	količina ali trajanje shranjenih podatkov
PaaS	število strank
	tip stranke (razvijalci, končni uporabniki)
	shramba podatkov
	obdelava podatkov
	uporaba mrežnih virov
	časovna uporaba ogrođja
IaaS	procesorski čas
	čas shranjenih podatkov (v urah)
	uporaba pasovne širine omrežja
	časovna uporaba omrežja (v urah)
	uporaba dodatnih storitev (revizija, avtomatična razširitev)

Vir: L. Badger, T. Grance, R. Patt-Corner, & J. Voas, *Cloud Computing Synopsis and Recommendations*, 2012, str. 5-1, 6-1 in 7-1.

3.2 Kriteriji o kakovosti storitve

Sporazum o storitvi je pravno zavezujoč dogovor med podjetjem kot uporabnikom oblaka in ponudnikom storitve. Elementi sporazuma so pravne odgovornosti, ki jih mora ponudnik izpolnjevati do stranke, seznam zahtev, ki jih ponudnik ne bo dostavil zaradi omejitev ipd. ter seznam zahtev, ki jih mora uporabnik sprejeti takšne kakršne so. Del sporazuma je tudi sporazum o ravni storitve. Sporazum o ravni storitve – SLA je sporazum med uporabnikom in ponudnikom storitve o zahtevani QoS, ki jo bo ponudnik dostavil stranki (Kearney & Torelli, 2011, str. 43). Dogovor vključuje tehnična zagotovila ponudnika, kako naj bi storitev delovala in kazni v primeru kršitve teh zagotovil (Badger et al., 2012, str. 3–1).

Za podjetje je ključnega pomena, da razume elemente sporazuma, kaj točno določajo in kakšne so morebitne omejitve. Zavedati se mora svojih odgovornosti in odgovornosti ponudnika ter sankcije v primeru kršitve sporazuma (Badger et al., 2012, ES-2). Do kršitev sporazuma s strani ponudnika lahko pride zaradi različnih razlogov: preobremenitve virov,

slab nadzor prometa, nepošteno poslovanje ipd. V izogib ponudnikom s slabimi QoS, Macías in Guitart (2016, str. 460) priporočata izbiro uglednega ponudnika. Pri tem imajo veliki ponudniki, ki s svojo razširjeno arhitekturo omogočajo zaupanja vredno QoS, prednost pred manjšimi. To seveda ne pomeni posploševanja, da manjši ponudniki zaradi tega nudijo slabše QoS. Na drugi strani je slaba stran velikih ponudnikov, da nudijo le standardizirane SLA z malo možnostmi za pogajanje in prilagajanje (Martens & Teuteberg, 2012, str. 873).

V naslednjih podpoglavjih so opisani nefunkcionalni kriteriji QoS, ki so predmet SLA in do sedaj v delu še niso bili opisani. Tu zajamem le nekatere kriterije, ki so bili najpogosteje uporabljeni v analiziranih modelih. Nekateri izmed opisanih modelov se bralcu na prvi pogled zdijo samoumevni, vendar je nujno, da so del SLA pogodbe, saj neizmerno vplivajo na celotno QoS.

3.2.1 Zmogljivost

Po navajanju Schmelzer (2008) je zmogljivost (angl. *performance*) storitve abstraktni termin, ki različnim ljudem pomeni različno. Za posameznika, ki upravlja z računalniškim omrežjem je zmogljivost lastnost njegove odzivnosti, varnosti in razpoložljivosti. Za poslovnega analitika ali projektne vodje zmogljivost pomeni, da se procesi izvajajo v skladu s specifikacijami. Za razvijalca je zmogljivost storitve doseganje funkcionalnih potreb. S poslovnega vidika zmogljivost pomeni doseganje ključnih kazalnikov uspešnosti (KPIs). Kazalniki opredelijo storitev in jo naredijo uporabniku razumljivo. Na trgu je mogoče najti veliko komercialnih orodij za merjenje KPIs in izdelavo poročila zmogljivosti (Garg et al., 2013, str. 1022). Za celovit opis je potrebno zajeti različne KPIs iz različnih področij. V nadaljevanju bom opisal nekatere pogosto uporabljene KPIs v modelu odločanja izbire ponudnika oblaka.

3.2.1.1 Odzivnost

Čas, ko uporabnik čaka na odgovor storitve, imenujemo odzivnost storitve (angl. *service response time*). Odzivnost je torej čas od oddaje zahtevka do prejetja odgovora oz. rezultata s strani sistema (Chorafas, 2011, str. 244). Odzivnost oblaka lahko merimo, bistveno težje pa opredelimo vrednosti posameznih elementov, ki sestavljajo celotno odzivnost. Eden od zelo pomembnih elementov odzivnosti, na katerega ponudnik v celoti ne more vplivati, je **latenca** (ang. *latency*) omrežja.

Latenca omrežja je čas potreben, da oddana zahteva na viru doseže prejemnika na cilju ter rezultat potuje nazaj od cilja do vira. Pri storitvi v oblaku imamo tako vsaj en obhod, sestavljen iz zahteve in rezultat. Težava se pojavi, ker latenca ni v domeni ponudnika storitve in zanj ne more odgovarjati. Zate je potrebno izdelati analizo in preveriti, ali celotna odzivnost zadošča uporabnikovim zahtevam. Ravno latenca je razlog, da storitve

oblaka niso primerne za obdelavo časovno kritičnih nalog (Badger et al., 2012, str. 8-1–8-2 in 9-3–9-4). Poznamo pa tudi vpliv latence omrežja, ki jo lahko štejemo v domeno ponudnika in nastane zaradi komunikacije med geografsko porazdeljenimi podatkovnimi centri (Son, Jung, & Jun, 2013, str. 609).

3.2.1.2 Razpoložljivost

Razpoložljivost (angl. *availability*) je nefunkcionalna zahteva opredeljena kot odstotek časa, ko je storitev dostopna (Nabi, Toeroe, & Khendek, 2016, str. 54). Toeroe in Tam (2012) razpoložljivost opredelita z enačbo (1), kjer je spremenljivka MTTF čas, ko je storitev ponovno na voljo po odpravi zadnjega izpada in MTTR spremenljivka, ki opisuje oceno časa za odpravo napake.

$$\text{razpoložljivost} = \frac{MTTF}{(MTTF + MTTR)} \quad (1)$$

Serrano et al. (2016, str. 234) opredelitev, kjer nastopajo časovne spremenljivke, označijo kot stopnjo uporabe (angl. *use rate*). Poleg te opredelijo še razpoložljivost, ki jo poimenujejo stopnja odsotnosti (angl. *abandon rate*) in je razmerje med sprejetimi uporabniškimi zahtevami in vsemi zahtevami.

Razpoložljivost se tako zapisuje z odstotkom, kjer 100% predstavlja storitev, ki nikoli ni nedosegljiva, torej deluje brez prekinitev. Razpoložljivost storitve v IT je nad 99% oz. za visoko razpoložljive sisteme je razpoložljivost vsaj 99,999%. Ta zapis imenujemo pet-devet sistem (5-9s, angl. *five nines*). Preglednica Tabela 3 prikazuje nekaj primerov razpoložljivosti za časovno obdobje enega leta ter maksimalni čas, ko je storitev lahko izpadla.

Tabela 3: Seznam oznak razpoložljivosti z maksimalnimi časi izpada

Razpoložljivost	Oznaka	Maksimalni čas izpada*
99%	2-9s	3 dni 15 ur in 36 minut
99,9%	3-9s	8 ur 45 minut in 15 sekund
99,99%	4-9s	52 minut in 34 sekund
99,999%	5-9s	5 minut in 15 sekund
99,9999%	6-9s	32 sekund

Legenda: *maksimalni čas izpada velja za obdobje enega leta

Vir: M. Toeroe & F. Tam, *Service Availability: Principles and Practice*, 2012, tabela 1.1.

Če je razpoložljivost storitve nižja kot je bilo dogovorjeno, je stranka posledično upravičena do določenega dobroimetja (Myerson, 2009). Kljub temu, da imajo veliki

ponudniki replicirane podatkovne centre v različnih regijah, v izogib izpadu poslovanja Armbrust et al. (2009, str. 14) priporočajo uporabo več storitev različnih ponudnikov ter distribucijo podatkov.

3.2.2 Zanesljivost

Zanesljivost (angl. *reliability*) je poleg varnosti in prilagodljivosti med prvimi tremi skrbmi uporabnikov pri prehodu iz tradicionalnega računalništva na oblak (Gill, 2011, str. 46). Baranwal in Vidyarthi (2014, str. 835) trdita, da je zanesljivost ponudnika merilo za ugled.

Badger et al. (2012, str. 8-2) opredelijo zanesljivost kot verjetnost, da bo storitev delovala brez kritičnih napak v določenem časovnem obdobju. Ta pa je odvisna od zanesljivosti štirih komponent: strojne in programske opreme ponudnika, zaposlenih pri ponudniku, poveztivosti do storitev ter zaposlenih v podjetju.

Za zagotavljanje zanesljivosti je uporaba tehnologije razpršenih podatkov nujna. Pri tem je nujna tudi fizična razpršenost podatkovnih centrov. S tovrstno razpršitvijo ponudniki preprečijo izpad delovanja v primeru hujših elementarnih nesrečah, vojnah ipd. in s tem povečajo zanesljivost. Geografska porazdelitev in hkrati izbira lokacije podatkovnih centrov je po navajanju Royal Pingdom (2008) in Badger et al. (2012, str. 8-6) pomembna tudi z vidika stroškov (strošek izgradnje centra, poceni energija, poceni delovna sila, davek ipd.), skrb za varnost, kakovost javne infrastrukture, izobraženost populacije in oddaljenost od ostalih centrov.

Razpršenost podatkovnih centrov po drugi strani vzbuja podjetjem skrb o tem, kje točno bodo njihovi podatki shranjeni. Po navajanju Binning (2009) je to eno od najpogostejše zastavljenih vprašanj uporabnikov storitev oblaka. Kot pišeta Jansen in Grance (2011, str. 17) podjetja zanimajo podrobne informacije o lokaciji podatkovnega centra ter njegovi tehnološki in fizični zaščiti. Odgovori na tovrstna vprašanja morajo zadostiti tudi zakonskim in regulatornim zahtevam.

Ne glede na trenutno zanesljivost ponudnika lahko pride do izpada sistema. V tem primeru je zelo pomemben atribut **ponovna vzpostavitev** sistema (angl. *disaster recovery*). Vzrok za izpad sistema so lahko višje sile, kot npr. elementarne nesreče ali vojna, kraja strojne opreme ali sodni pregon ter ne tako redka odpoved (dela) strojne opreme. Zato je pomemben dogovor o ponovni vzpostavitvi sistema, ki vključuje največji potreben čas za ponovno vzpostavitev sistema (Badger et al., 2012, str. 8-5). Ponudniki v izogib večji škodi replicirajo podatke in okolja (podatkovne centre) na več mestih, regijah ali celinah. Kot navaja Mombrea (2012) je ravno slednje razlog, da posamezna odpoved strojne opreme ne ogrozi delujočega sistema ali podatkov in ima v primerjavi s tradicionalno arhitekturo veliko prednost.

3.2.3 Varnost

Po navajanju Sarna (2010, str. 171) je varnost ena izmed najpomembnejših zahtev podjetij pri vzpostavitvi informacijskega sistema. Varnost opredeli kot možnost vpogleda v podatke za določenega uporabnika. Ta lahko dostopa le do podatkov, ki so mu namenjeni in ničesar drugega. Po raziskavi opravljeni v začetku leta 2016 RightScale navaja, da je bil izziv oz. skrb zagotavljanja varnosti oblaka na prvem mestu vse do leta 2015. Šele v tem letu ga je prehitela skrb za zadostno zagotavljanje virov in strokovnega znanja, skrb za varnost pa se je uvrstila na drugo mesto.

Za zagotavljanje celostne varnosti je potrebno upoštevati različne vidike varnosti (Yeluri & Castro-Leon, 2014, str. 22):

- **dostopnost**, katero sta Jansen in Grance (2011, str. 31) opredelila kot merilo v kolikšnem obsegu so dosegljivi in uporabni vsi računalniški viri organizacije. Skupaj z dostopnostjo sta **skladnost in spremljanje** potrebna kriterija za zagotavljanje nemotenega dostopa do varnostnega nadzora in statusa delovanja nižje ležečih plasti oblaka z možnostjo revizije. Merjenje teh atributov omogočajo podjetjem primerjavo s potrebnimi ali zahtevanimi vrednostmi. Pri tem so jim v pomoč različni standardi, kot npr. NIST-ov zvezni zakon za upravljanje informacijske varnosti (angl. *federal information security management act*) (IS Decisions, b.l.).
- **Zaščita in pregled podatkov** se navezuje na lokacijo, kjer se nahajajo tako primarni podatki, kot tudi sistem s katerim podjetje upravlja. Ključni elementi poleg lokacije so še varnostno kopiranje, ponovna vzpostavitev podatkov ter ločevanje in verodostojnost podatkov.
- **Arhitektura**, katero sestavljajo elementi zaščite pred nepooblaščenimi vdori, podpora za večnajemniško okolje in zagotavljanje medsebojne izoliranosti ter varnostna politika, ki se izvaja skladno in natančno s pogodbo.
- Pri **identifikaciji** je treba upoštevati preverjanje pristnosti, pravila in pravice uporabnikov tako v samem sistemu, kot tudi med sistemi znotraj organizacije.
- Da oblak v skladu z varnostnimi zahtevami zagotavlja izvajanje storitev na zahtevo in se pri tem hitro odziva na povečanje ali zmanjšanje virov, je potrebna visoka stopnja **avtomatizacije in orkestracije**.

Za ovrednotenje stopnje varnosti storitev oblaka si lahko podjetja pomagajo z merjenjem posameznih elementov varnosti. Pri tem morajo biti dosledni, natančni in celovito opraviti vse potrebne meritve. Druga, bolj priporočljiva možnost, je izbira ponudnika, ki s standardnimi certifikati dokazuje svojo raven doseganja varnosti storitve. Priporočljivo je, da ponudnik poseduje več različnih certifikatov povezanih z različnimi zahtevami varnosti. Nekaj primerov standardiziranih poročil in certifikatov (Tang & Liu, 2015, str. 66–67): poročilo SOC 2 Type 2 in PCIDSS, FedRAMP pooblastilo ter certifikati ISO/IEC 27001 in CSA STAR.

Prav zaradi dvoma o varnost podatkov na oblaku je ta kriterij vsa leta prednjačil pred ostalimi. Armbrust et al. (2009, str. 15) navajajo, da je bil stavek »Naši občutljivi poslovni podatki ne bodo nikoli v oblaku.« slišán že večkrat. Nekaj tega dvoma je povsem upravičenega, saj ne gre sklepati, da so oblaki povsem varno okolje. Po navajanju Smith (2014) oblak v primerjavi s tradicionalno strukturo na lokaciji podjetja (angl. *on-premises*) zagotavlja večjo varnost podatkov in celotnega sistema. To velja toliko bolj za mala in srednje velika podjetja, katerih IT arhitektura ne dosega tako sofisticiranih mehanizmov kot to dosegajo veliki ponudniki oblaka. Ti v razvoj varnosti vlagajo velike vsote denarja in virov, saj bi morebitna varnostna hiba lahko pomenila konec njihovega poslovanja. To dokazuje tudi, da je število odmevnih vdorov v javni oblak precej manjše, v nasprotju z odmevnimi vdori v različne sisteme s tradicionalno zasnovo. Kljub manjšem številu vdorov pa so posledice vdora lahko precej bolj obsežne pri oblaku kot pri individualnem sistemu uporabnika (npr. vdor v spletni portal Ashley Madison leta 2015 in razkritje imen članov). Armbrust et al. (2009, str. 15) celo trdijo, da so predhodno šifrirani podatki na oblaku bolj varni kot nešifrirani podatki v lokalnem podatkovnem centru podjetja in dodajajo, da ni bistvenih ovir za izdelavo okolja v oblaku, ki bi bilo enako ali celo bolj varno od tradicionalne lokalne zasnove podatkovnega centra.

3.2.4 Skalabilnost

Oblak je zelo primerna rešitev za sisteme, ki so za nemoteno delovanje odvisni od zanesljivosti in skalabilnosti (Madria & Sen, 2015, str. 50). Skalabilnost je merilo zmožnosti izboljšanja propustnosti sistema, zmanjšanja odzivnega časa ali povečanja števila sočasnih uporabnikov z dodajanjem strojne opreme med samim delovanjem storitve (Williams & Smith, 2004, str. 1). Obstajata dve vrsti skalabilnosti, vertikalna in horizontalna. Pri vertikalni skalabilnosti se virtualnim okoljem za povečanje zmogljivosti dodaja vire. Pri horizontalni se aktivira dodatna virtualna okolja, ki lahko obdelajo vse zahteve pri povečanih obremenitvah in konicah. Pri ponudnikih oblaka je veliko bolj v uporabi horizontalna skalabilnost (Baranwal & Vidyarthi, 2014, str. 835).

Skalabilnost tako zagotavlja uporabniku navidežno neskončno kapaciteto računalniških virov. Uporabniki velikokrat pišejo, da njim samim ni treba skrbeti za konice, vendar je to pri deljenih virih lahko hitro težava. S tem postavljajo dvom tudi na ostale trditve glede zmogljivosti storitve (Chorafas, 2011, str. 37).

Zelo odmevni primer, ki opisuje enormno povečanje virov, se je zgodil ponudniku oblaka Amazon. Podjetje Animoto je po objavi svoje storitve na družabnem omrežju naraslo v treh dneh iz 50 instanc (strežnikov) na 3500 instanc. Še nekaj dni kasneje je Animoto uporabljal več kot 5000 instanc za obdelavo vseh zahtev. Nihče ni mogel predvideti, da se bo število vsakih 12 ur podvojilo (Hsiao, 2008).

3.2.5 Elastičnost

Ena glavnih ugodnosti oblaka je elastičnost, kjer se kapaciteta virov prilagaja trenutnim potrebam uporabnika (Islam, Lee, Fekete, & Liu, 2012, str. 85). Ta lastnost je bolj pomembna za IaaS kot za PaaS ali SaaS (Baranwal & Vidyarthi, 2014, str. 833).

Elastičnost je sposobnost sistema, da prilagodi kapaciteto virov z njihovo rezervacijo in sproščanjem skladno z obremenitvijo sistema. Pri tem naj se uporabljeni in zahtevani viri čimbolj približajo (Herbst, Kounev, & Reussner, 2013, str. 24). V literaturi je mogoče najti opredelitve, kjer elastičnost enačijo s skalabilnostjo (Schouten, 2012). Enačenje ovržejo Herbst et al. (2013, str. 24–25) z razlago, da je skalabilnost predpogoj za elastičnost in se ne nanaša na to, kako hitro, pogosto in podrobno se viri prilagajajo potrebam. Skalabilnost je le zmožnost povečanja sistema in se v nasprotju z elastičnostjo ne nanaša neposredno na to, kako se povečanje virov približa zahtevanim potrebam.

Pri elastičnosti je približevanje oskrbe z viri trenutnim zahtevam za uporabnike ključnega pomena. Ponudniki navadno omogočajo spremembe oskrbe virov za določeno enoto. Pri tem se kapaciteta virov poveča več kot jo zahteve potrebujejo, s tem pa se povečajo tudi stroški. Tej formuli je potrebno dodati še model obračuna (časovno enoto). Oba parametra doprineseta k previsokemu obračunu stroškov glede na podane zahteve. Druga skrajnost je omejitev povečanja virov in s tem zavračanje zahtev ob konicah. Vseeno pa viri niso v celoti izkoriščeni ob normalnem delovanju sistema (Armbrust et al., 2009, str. 10–12; Islam et al., 2012, str. 86; Baranwal & Vidyarthi, 2014, str. 836).

3.3 Funkcionalni kriteriji

V nadaljevanju sta opisana dva v literaturi uporabljena funkcionalna kriterija, ki se nanašata na QoS. Sprva je opisana uporabnost storitve, katero sestavljajo razumljivost, enostavnost namestitve, učljivost in funkcionalnost. Nato opišem še prenosljivost in integracijo storitve.

3.3.1 Uporabnost

Ko se podjetje odloča preiti na oblak, je uporabnost storitve zelo pomemben kriterij, ki ga mora podjetje vključiti v analizo. Hitrejši kot bo proces prehoda na oblak, manjši bodo stroški podjetja. Uporabnost vpliva na čas potreben za vzpostavitev in namestitev storitve ali okolja, učenje in razumevanje storitve. Večja kot je uporabnost, krajši je čas celotnega prehoda, manjši so stroški podjetja (Garg et al., 2013, str. 1013 in 1016).

3.3.2 Prenosljivost

Pri prehodu na oblak je nanj treba prenesti obstoječo infrastrukturo ali sistem. Pri tem je za uporabnike zelo pomembno, da ima ponudnik orodja s katerimi si lahko pomagajo pri migraciji. Prenosljivost je velika skrb uporabnikov in lahko celo prepreči prehod na oblak. Pri tem je treba upoštevati tudi prenosljivost med ponudniki, v primeru, če se uporabnik kasneje odloči, da bo zamenjal ponudnika. Pri tem je pomembno, da ponudnik uporablja standardizirane protokole, metode in vmesnike (npr. TCP/IP (angl. *transmission control protocol*), SOAP (angl. *simple object access protocol*), XML (angl. *extensible markup language*), REST (angl. *representational state transfer*) itd.). To je pri IaaS lažje, ker gre za nižje ležeče komponente, pri PaaS in SaaS pa ponudniki velikokrat uporabljajo protokole, ki so jih razvili samo za potrebe oblaka. Navadno tehnične podobe ali celo enake funkcionalnosti tudi različno poimenujejo, kar doprinese še k večji netransparentnosti med njimi (Badger et al., 2012, str. 8-5).

Zelo sorodna prenosljivosti je **integracija** z drugimi sistemi. Pri integraciji je pomembno, da ponudnik omogoča standardiziran dostop do zelenih vmesnikov.

3.4 Pravni kriteriji

Pravni kriteriji morajo imeti največjo tezo, saj so sankcije v primeru kršitve le teh, podjetju škodljive. Za kršitve pravnih določil so navadno določene visoke globe, odvisno od teže kršitve in države ali skupnosti (npr. Evropska unija), ki je zakon izdala (Höllwarth, 2012, str. 99–105). V pravne kriterije vključujemo zakonodajo o varstvu osebnih podatkov in davčno zakonodajo.

Kot navaja Höllwarth (2012, str. 73–74) na najvišji ravni abstrakcije delimo kriterije v dve kategoriji: kriteriji, ki zagotavljajo **skladnost** s pravom in **sklepanje pogodb**. Teža različnih tem je seveda lahko različna: od kršitve zakona in sodnega pregona, do kršitve določene ureditve in npr. izgub licenc do kršenja pravil in poslovnega modela in s tem npr. omadeževanje poslovnega ugleda.

Tipičen kriterij, ki ga zasledim v literaturi je omejevanje lokacije shranjevanja podatkov. Dober primer tovrstne zahteve so posebne institucije, kot so npr. banke ali zavarovalnice, ki v nekaterih državah ne smejo svojih podatkov izvoziti izven meja države poslovanja.

3.5 Ekološki kriteriji

V literaturi v modelih vključujejo tudi kriterije, ki se navezujejo na ekologijo. Najpogosteje se na tem področju uporablja kriterij **izpustov toplogrednih plinov CO₂**. Čeprav podatkovni centri s samim izpustom nimajo neposredne povezave, pri ogromni porabi

električne energije posredno vplivajo na izpuste. Ponudniki za svoje podatkovne centre skušajo uporabljati obnovljive vire, kot je voda, sončna toplotna energija, vetrna energija in biomasa (Höllwarth, 2012, str. 219–220). Tem primerno so izbrane tudi lokacije podatkovnih centrov, npr. ob reki za hlajenje ali v državah kjer je v povprečju skozi celo leto hladnejši zrak.

Da bi lahko kategorizirali, kateri podatkovni center porabi več ali manj energije, je organizacija Green Grid ustvarila standard poimenovan **faktor URE** – učinkovita raba energije (angl. *power utilization efficiency*). Faktor URE označuje porabo energije v podatkovnem centru za vse naprave, ki so potrebne za normalno delovanje takšnega centra (baterije za neprekinjeno napajanje, klimatske naprave, osvetlitev, nadzor in nadzorni center, sistemi spremljanja, požarna varnost itd.; Höllwarth, 2012, str. 220–223).

4 ŠTUDIJA PRIMERA: PROGRAMSKA REŠITEV ZA MARINE

Raziskava študija primera omogoča razumevanje zapletenih vprašanj ali stvari. Raziskava temelji na dognanjih in ugotovitvah predhodnih raziskav, katerim je mogoče razširiti ali dodati nova znanja (The Case Study as a Research Method, 2016). Študija primera je primerna predvsem, kadar je pomemben kontekst proučevanja, v katerem se primer nahaja in ko se je iz primera možno veliko naučiti (Patton, 1987, str. 19). Yin (2009, str. 18) študijo primera opredeli kot empirično raziskovalno metodo, ki podrobno proučuje sodobne pojave v sklopu svojega realnega konteksta. Raziskava je primerna predvsem, kadar meje med pojavom in kontekstom niso jasno opredeljene.

Študije več primerov imajo v raziskovanjih večjo veljavo kot študija enega primera in dosegajo boljše ter bolj robustne rezultate raziskav (Bengtsson, 1999, str. 3). Maoz (2002, str. 164–165) navaja, da tovrstna raziskava prinaša preveč subjektivnega pogleda avtorja in ta posledično zapostavlja različne metodološke vidike. Kot posledica zapostavljanja metodologij pri študijah enega primera po njegovem te postajajo sinonim za poljubne raziskave, kjer je rezultat vedno ugoden za izbran primer. Kritika je nekoliko neupravičena, kar dokazujejo navajanja nekaterih prednosti te metodologije. Bennett in Elman (2010, str. 505–506) kot prednost navajata implicitno primerjanje raziskav z možnostjo opredelitve primera: devianten, manj ali bolj tipičen, skrajni primer. Prav tako Yin (2009, str. 47–50) navaja več razlogov za uporabo študije enega primera: primer predstavlja kritičen primer, primer predstavlja edinstven primer, primer je reprezentativen ali tipičen, primer predstavlja novost in primer je longitudinalen. Echstein opredeli več prednosti študije enega primera in jih podkrepi s primeri (Willis, 2014).

Sam sem v svojem delu uporabil študijo enega primera. Študija več primerov zahteva dodatne vire in čas ter presega sredstva, ki jih imam na voljo. Z drugega vidika mi študija enega primera omogoča njegovo podrobnejšo in natančnejšo analizo.

Pri študiji primerov je izrednega pomena njihova izbira, toliko bolj pri študiji enega primera. Kot navaja Eisenhardt (1989, str. 537) naključna izbira primera ni potrebna in je celo nezaželena. Za potrebe raziskave svojega dela sem se osredotočil na primer, ki bo dovolj reprezentativen in bo s svojimi lastnostmi zadostil mojim kriterijem raziskave.

Za študijo primera sem izbral programsko rešitev, ki je celostna rešitev za potrebe upravljanja marin. Podano rešitev že dve desetletji trži in razvija majhno slovensko podjetje iz Ljubljane. Izbiro te programske rešitve za študijo primera utemeljujem z naslednjimi razlogi:

- v podanem primeru me podrobno zanima **celotni kontekst** povezan z delovanjem programske rešitve, vključno z zahtevami in specifikami različnih strank, tehnološkimi omejitvami, pravnimi določili itd. Viri in čas potreben za podrobno analizo primera posledično zahteva manjše število obravnavanih primerov.
- Programsko rešitev podjetje trži po različnih državah: v Sloveniji, na Hrvaškem, v Španiji, na Nizozemskem, v Italiji, Švici, na Kitajskem, v Avstraliji, Braziliji, ... Tako imam na voljo veliko množico zahtev iz različnih držav, kjer se delovanje marin razlikuje. Razlike so zaradi zakonskih regulativ, kulture, namena in načina delovanja tamkajšnjih marin ter specifičnih zahtev lastnikov marin. Delovanje marin se prav tako razlikuje glede na njeno velikost, lokacijo (morje, jezero, reka) in vrsto (morska, celinska, ograjena in vgrajena; Mill, 2007, str. 113). Zaradi velike in raznolike množice parametrov menim, da gre za **kakovosten reprezentativen primer**, s katerim je možno prikazati različna obnašanja modela.
- Patton (1987, str. 19) predlaga, da izberemo primer s katerega se lahko **največ naučimo**. Podjetje svojim strankam svetuje tudi glede IT infrastrukture v odvisnosti od različnih parametrov. Pri izbranem primeru je implementacija možna s povsem tradicionalno infrastrukturo, z uporabo privatnega oblaka, z najemom javnega IaaS ali tudi z najemom rešitve kot SaaS.

V nadaljevanju sprva opišem programsko rešitev s funkcionalnega in nefunkcionalnega vidika. Raziščem in opišem motiviranost uvedbe oblaka v dani primer in analiziram zahteve strank. Najpomembnejše in najpogostejše navedene zahteve opišem. S pomočjo intervjujev opredelim in opišem kriterije, ki so potrebni za študijo primera. V zadnjem podpoglavju tega poglavja opisujem alternative ponudnikov storitve IaaS.

4.1 Opis programske rešitve

Z več kot dvema desetletjema izkušenj z rešitvami za upravljanje marin, podjetje razume potrebe moderne marine za zagotavljanje stroškovno učinkovite, zanesljive in prilagodljive rešitve. Glavni namen programske rešitve za upravljanje marina je omogočiti upravljalcu popoln nadzor nad procesi znotraj marine. Učinkovita programska rešitev pomaga

zmanjšati stroške, povečati produktivnost in povečati dobiček. Sistemska orodja omogočajo popolno obdelavo podatkov, od preprostih poročil, do pomoči pri strateških odločitvah.

4.2 Funkcionalna področja programske rešitve

Programska rešitev je sestavljena iz paketa osnovnih in dodatnih funkcionalnosti, opsijskih dodatnih modulov in zunanje strojne opreme. Uporabniki programske rešitve so vodstvo, zaposleni v zalednem poslovanju, mornarji (osebe, ki pomagajo pri pristajanju plovil), podporna služba, serviserji in stranke marine (lastniki plovil).

Osnovni paket, ki je namenjen primarnemu upravljanju marine, vključuje funkcionalnosti za:

- delo na recepciji,
- rezervacije,
- nadzor in kontrola privezov,
- fakturiranje,
- upravljanje odnosov s strankami (CRM) in
- osnovna poročila.

Poleg osnovnega paketa programska rešitev vključuje dodatne, naprede funkcionalnosti, kot so različne integracije (SMS pošiljanje, elektronska pošta...); planiranje dela, dodatnih storitev in servisa; upravljanje s pogodbenimi strankami; direktorski informacijski sistem (angl. *executive information system* – *EIS*); finančni modul z integracijo z glavno knjigo. Dodani moduli omogočajo še upravljanje zaloge; vodenje *charter-ja*; upravljanje z dodatnimi storitvami kot so oddaja sob, trgovina, organizacija dogodkov in izobraževanj, upravljanje jahtnega kluba, restavracijo, kamp, menjalnico itd. Poleg funkcionalnosti namenjenim osebju marine, vključuje tudi funkcionalnost namenjeno strankam marine, tako imenovanem portalu. Na portalu lahko lastniki plovil pregledujejo lastne podatke, objave marine ter oddajo različne zahteve.

Vso primarno funkcionalnost je možno nadgraditi z zunanjimi moduli, ki vključujejo dodatno strojno opremo. Z namestitvijo teh modulov marina doseže večjo stopnjo avtomatizacije, npr. avtomatska identifikacija plovila, avtomatska kontrola privezov ipd. Prav tako je mogoče na daljavo upravljati z viri na pomolih, kot je polnjenje goriva, vode ali električnih akumulatorjev.

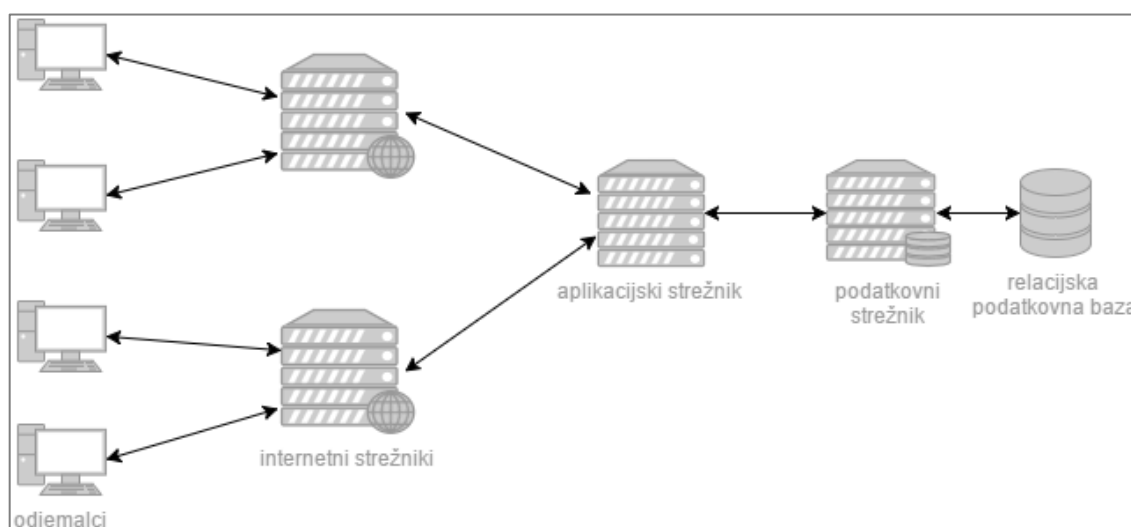
4.3 Arhitektura programske rešitve

Arhitekturo dotične programske opreme razdelim v 4 sklope:

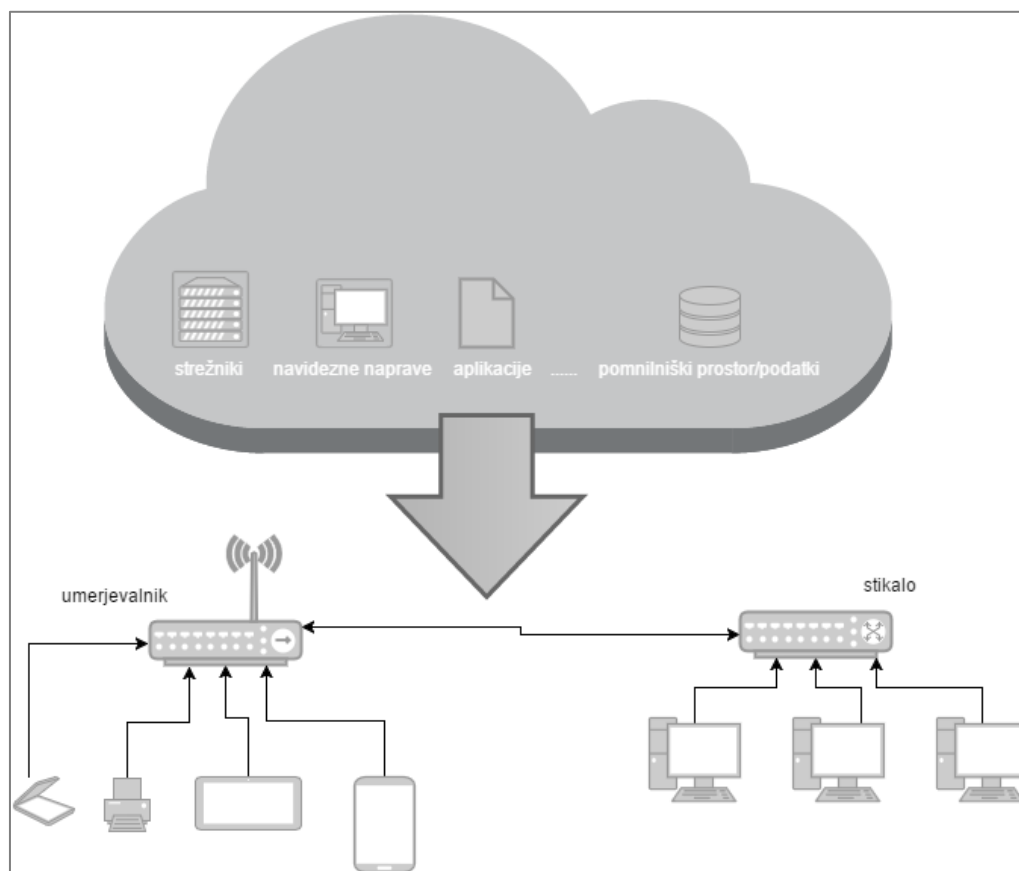
- za vsebino mojega dela sta najpomembnejša aplikacijski in podatkovni strežnik, katerih funkcionalnost je mogoče zasnovati z uporabo oblaka. V primeru tradicionalne arhitekture jih lahko združimo v en sam fizični strežnik, vendar dobra praksa to odsvetuje.
- Lahki odjemalci oz. pametne naprave, s kateri upravljajo tako zaposleni marine, kot tudi njihove stranke. Pametne naprave največ uporabljajo mornarji na pomolih, podporna služba in serviserji.
- Osebne ali namizne računalnike navadno uporablja osebje na recepciji, zaposleni v zaledju in vodstveni kader. S tehnološkega vidika lahko te računalnike zamenjamo s pametnimi napravami, vendar je njihova uporaba v primerjavi z namiznimi računalniki veliko manj funkcionalna. Zaledne službe veliko bolj uporabljajo tipkovnico za vnos podatkov, dodatna orodja za izdelavo naprednih poročil ipd.
- Zunanje naprave s katerimi rešitev komunicira (npr. identifikatorji za plovila) na moje delo nimajo vpliva in jih ne bom opisoval.

Na spodnjih slikah je prikazana zasnova programske rešitve z uporabo tradicionalne arhitekture z ločenim aplikacijskim in podatkovnim strežnikom (Slika 6) in z uporabo oblaka (Slika 7). Pri slednji opciji sta aplikacijski in podatkovni strežnik implementirana na oblaku (javni ali zasebni).

Slika 6: Tradicionalna IT zasnova



Slika 7: Zasnova z uporabo oblaka



4.4 Motivacija za uvedbo oblaka

Še pred pravim obstojem oblaka, ko je podjetje začelo z implementacijo programske rešitve za marine, je zaznalo vrzel pri vzpostavitvi IT arhitekture v marinah. Marine niso imele lastne strojne opreme, nakup le te pa bi pomenil velik poseg v njihova finančna sredstva. To toliko bolj velja za manjše marine, vendar jih je potreba po konkurenčnosti in tudi zmanjševanju stroškov pripeljala do ugotovitve, da je informatizacija nujno potrebna. Njihovo arhitekturo je sestavljalo nekaj osebnih računalnikov za potrebe delovanja zaledne službe in recepcije. Glavni strošek bi bil nakup strežnikov in vzpostavitev lokalnega omrežja. Kmalu se je porodila ideja o strežniku v lasti podjetja, ki bi služil kot skupni vir opreme za potrebe podatkovnega strežnika. Glavna ovira, poleg takrat zelo drage strojne opreme, so bila tudi draga in težje dostopna omrežja – internet je v razvitem svetu leta 1995 uporabljalo le dobrih 10% populacije (Relatively Interesting, 2012). Prav tako je bila vrzel v hitrosti prenosa podatkov po omrežju. Hitrost prenosa podatkov v lokalnih omrežjih je dosegala 12,5MBps (Tinnefeld, 2015, str. 13–14), hitrost prenosa preko interneta pa po objavi digitalnega obrežja z integriranimi storitvami (angl. *integrated services digital network – ISDN*) leta 1996 64Kbps, pred tem polovico manj (BT launches high speed ISDN access to the internet, 1996). Kljub temu, da je podjetje to delno realiziralo

(varnostne kopije ipd.), je bila takšna zasnova bolj izjema kot pravilo. Vseeno pa je ideja ostala, potrebovali so le tehnološko bolj fleksibilno in predvsem cenejšo rešitev.

Že v zgodnjih letih razvoja oblaka je podjetje strmelo k tovrstni rešitvi oz. ponudbi in odločitev o spremembi razvoja se je izkazala za dobro odločitev. Razvoj je sledil trendu, kjer so lahki odjemalci zamenjali težke.

Kmalu je razvoj oblakov bliskovito napredoval in sedaj ima že nekaj let velik vpliv na IT področju in je pomemben gospodarski faktor (Repschläger, Wind, Zarnekow, & Turowski, 2011, str. 163). Čehovin (2011) napoveduje, da bo do leta 2020 večina uporabnikov izvajala aplikacije na oblaku, namesto lokalno na osebnih računalnikih. Zelo strmo rast napovedujeta tudi analitična hiša IDC (2014), ki navaja da bo do leta 2018 za potrebe oblaka v razvoj vloženo več kot 127 bilijonov ameriških dolarjev ter raziskovalno svetovalna hiša Gartner (2015), ki ocenjuje porabo 201 bilijona do leta 2019.

V okvirih tega konteksta je implementacijo programske rešitve mogoče izvesti na različne načine. Pri tem ima podjetje na razpolago sledeče:

- implementacijo programske rešitve preko ponudnikov **javnih oblakov** – IaaS. V tem primeru je strojna oprema v domeni ponudnika IaaS, prav tako skrb za varnostne kopije, podjetje pa poskrbi za dostavo programske rešitve do svoje stranke. Pri tej opciji podjetje potrebuje zanesljivega ponudnika IaaS, kjer mora celotna rešitev ostati v okviru pričakovanih stroškov strank.
- realizacijo arhitekture s pomočjo **privatnega oblaka**. Pri tem se podjetje v osnovi odloča med dvema različicama izvedbe:
 - uporabo tehnologije za realizacijo na način, kot je bila prvotna ideja o skupnem strežniku. Torej podjetje samo vzpostavi infrastrukturo znotraj neke države ali regije za več marin. S tem so stranke razbremenili skrbi za strojno opremo. Pri tej izvedbi bi se v sodelovanju s partnerji zelo približali željam in zahtevam strank – poudarek je na fizični lokaciji podatkovnega centra znotraj države, kjer marina posluje, hkrati se porazdeli strošek za uvedbo oblaka. Za varnostne kopije podatkov pa bi uporabili druge lokacije ali celo javne storitve oblakov (**hibridni oblak**). Tu pomanjkljivost hitrosti prenosa podatkov izpred let danes ni več smotrna. V današnjem času doseganje hitrosti prenosa na internetu 100Mbps ni nikakršna izjema (Smart, 2012). Pri tem načinu se zastavlja vprašanje o inicialni opredelitvi kapacitet strojne opreme in njena porazdelitev ob konicah, saj skupnost v resnici sestavljajo medseboj konkurenčne marine. Marine, ki so fizično locirane skupaj oz. poslujejo ena blizu druge ter si delijo iste vire oblaka, lahko hitro postanejo žrtev preobremenitve sistema zaradi istega vzroka, npr. navtični sejem ali regata v njihovi bližini.
 - Podobno realizacijo lahko dosežejo z najemom privatnega oblaka pri javnih ponudnikih oblaka. Tovrstni najem je sicer dražji v primerjavi z javnim, vendar

zagotavlja minimalne vire strojne opreme, katera je namenjena (angl. *dedicated*) izključno najemniku – zagotovljena višja raven varnosti.

Čeprav oblak velja za pojav, ki v IT in poslovnem svetu prinaša kopico ugodnosti in prednosti, Misra in Mondal (2011, str. 504) opozarjata, da naj podjetja vzpostavijo objektivni pogled in podrobno preučijo realne vrednosti vključevanja tehnologije oblaka v svoje poslovanje. Pri tem se Liu et al. (2016, str. 1) sprašujejo, ali so oblaki res primerni za vsako podjetje in hkrati navajajo, da po podatkih European Network and Information Security Agency (v nadaljevanju ENISA) 68,1% podjetij meni, da se z uporabo oblaka lahko izognejo investicijskim izdatkom za strojno opremo, IT podporo ipd. Pri tem je 30,6% podjetij prepričano, da tehnologija oblaka odstrani gospodarske in strokovne ovire, ki preprečujejo prenovo poslovnih procesov z uvedbo tehnologije (ENISA, 2009).

4.5 Analiza zahtev

V tem poglavju izpostavljam in opišem nekatere pomembnejše nefunkcionalne zahteve strank, ki so jih identificirali in analizirali v podjetju študije primera. Pri tem sem se osredotočil na zahteve, ki se neposredno navezujejo na zasnovo arhitekture sistema. Pri izbiri zahtev upoštevam tudi trenutno stanje in vizijo podjetja glede implementacije programske rešitve na oblaku.

Celotni seznam zahtev presega okvirje mojega dela. Velja pa omeniti, da je poleg v nadaljevanju opisanih zahtev, pogosto zaznati še naslednje zahteve:

- nekatere stranke želijo, da bi se njihovi podatki hranili znotraj države poslovanja, kar je glede na njihovo naravo neupravičeno in brez utemeljenega razloga,
- stranke želijo visoko razpoložljiv sistem,
- sistem mora biti odziven, s čim manjšo latenco,
- zadoščati mora dovolj veliko propustnost v odvisnosti od trenutne obremenitve sistema, kar je predvsem pomembno ob konicah, kot npr. regata, ki jo organizira marina,
- ponudnik mora biti dovolj prilagodljiv pri določanju vsebine pogodbe.

4.5.1 Strojna oprema

V zahtevah strojne opreme so bistvene le zahteve, ki se navezujejo na aplikacijski in podatkovni strežnik. Za spletni strežnik povzemam, da je del aplikacijskega strežnika. Ostala strojna oprema, kot so odjemalci (lahki in težki), lokalno omrežje vključno z brezžično povezavo (za lahke odjemalce na terenu) in ostala strojna oprema (npr. avtomatski identifikatorji) ne vpliva na moje delo in je ne bom zajel v analizo zahtev.

Podatkovni strežnik deluje na operacijskem sistemu Linux. Programska rešitev za podatkovno skladišče uporablja Oracle, različice 11g ali več. Zahteve za minimalne pogoje strojne in programske opreme so po priporočilih Oracla, ki jih je mogoče najti v njihovi spletni dokumentaciji (Oracle, b.l.):

- 2 GB začasnega bralno-pisalnega pomnilnika (angl. *random access memory – RAM*). Podjetje svetuje vsaj 4GB pomnilnika.
- 3,5–5 GB trdega diska (za potrebe delovanja Oracla brez podatkov). Priporočilo podjetja je vsaj 50 GB trdega diska.
- Procesor v razredu zmogljivosti Core2dur ali i5/i7 Xeon.

4.5.2 Licenciranje

Oracle je vodilno podjetje na področju podatkovnih baz (Ling, Qi, Qianyan, & Wei, 2013, str. 442). V svoji ponudbi nudi veliko različnih izdaj (angl. *edition*), z različnimi licencami in različnimi cenami. Po priporočilih ComputerWorldUK (2014) je pomembno, da podjetne izbere pravo izdajo, saj s tem lahko prihrani velike vsote denarja. Kriterij za izbiro izdaje podatkovne baze je odvisen od več parametrov. V primeru, da je stranka manjša marina, si lahko izbere tudi brezplačno licenco Oracle Express, ki ima sicer določene omejitve, kot npr. omejena velikost podatkovne baze (Daneasa, b.l.). Takšen način je za marine ugoden, saj je sicer strošek nakupa licence za Oracle zajeten (Wang, 2015). Pri nakupu licence je prav tako potrebno skrbeti za njeno obnovo in jo obnavljati v skladu s pogoji licenciranja (Kittlaus & Clough, 2009, str. 132). Pri implementaciji oblaka, imamo na voljo dva načina licenciranja (Murthy, Ameen, Sanjay, & Yasser, 2013, str. 645):

- uporaba lastne licence (angl. *bring your own license*). Pri tem načinu se na oblak namesti programska oprema, katere lastnik licence je uporabnik sam. Torej uporabnik uporablja lastno programsko opremo in mora pri tem sam skrbeti za obnove licence.
- Licenčni stroški so vključeni v najemnino (angl. *license charges included*). V tem primeru ponudnik oblaka nudi, npr. poleg navideznega računalnika, že nameščeno programsko opremo (npr. podatkovno bazo Oracle) za katero uporabnik plačuje licenco v sklopu najemnine oblaka. Uporabnik s tem nima skrbi glede upravljanja z licencami (obnove ipd.).

4.5.3 Delovanje v nepovezanem načinu

Eno od pogostih in ključnih vprašanj strank je delovanje storitve v primeru nepovezanosti z internetom. To vprašanje je smiselno za vse različice implementacije oblaka (privatni oblak, javni oblak itd.). Pri tem je treba poudariti, da je lokalno omrežje še vedno delujoče in vse naprave, ki so povezane vanj lahko med seboj komunicirajo. Torej je treba

zagotoviti delovanje sistema tudi, ko aplikacijski strežnik in podatkovna baza nista dosegljiva. Pri tem je treba zadostiti vsaj osnovne funkcionalnosti za poslovanje marine. Kot navajajo Badger et al. (2012, str. 8-2) je možnost sinhroniziranja podatkov oblaka zelo zaželeno. Pri tem je treba upoštevati, da sinhronizacija vključuje funkcionalnosti kot so kontrola nad različicami dokumentov, skupinsko kolaboracijo ipd. Dober primer takšnega delovanja je Google Docs, ki omogoča delovanje v nepovezanem načinu in kot piše Tucker (2008) lahko del oblaka vedno vzameš s seboj.

4.5.4 Varnost podatkov

V primeru implementacije z oblakom so podatki fizično locirani izven podjetja in podatki se prenašajo po javnem omrežju (internetu). Prav zato je zagotavljanje varnosti podatkov eden večjih izzivov (Vazquez-Poletti, Santos-Muzon, Florente, & Valero, 2014, str. 5). Tega se zavedajo tako IT strokovnjaki kot tudi stranke. Chang in Ramachandran (2016, str. 138) navajata, da je varnost podatkov najpomembnejša prioriteta in skrb pri uporabnikih oblaka. Tudi marine izpostavljajo skrb za varnost podatkov in nadzor nad njimi. Pri tem želijo različna dokazila za doseganje zadostne kakovosti varnosti podatkov (npr. ISO certifikati, certifikati institucije Precast/prestressed concrete institute – PCI ipd.) in garancije s strani ponudnika oblaka v primeru kršitve pogodbe (npr. zavarovanja v primeru naravnih nesreč, kraje podatkov ipd.).

Za doseganje varnosti podatkov je nujno potrebno poskrbeti tudi za varnostno kopijo podatkov in ponovne vzpostavitev sistema v primeru izpada sistema. Kopija podatkov je fizično locirana na drugi lokaciji kot primarni podatkovni strežnik. V primeru izpada sistema je potrebno v primernem (ali zahtevanem) času vzpostaviti ponovno nemoteno delovanje (Zhonglin & Yuhua, 2011, str. 1–2). Navadno je pri izvedbi oblaka kopija podatkov in ponovna vzpostavitev sistema vključena v najemnino, vendar naj to ne bo samoumevno. Preveriti je treba, ali ponudnik nudi to funkcionalnost pri storitvi oz. ali je treba to dodatno naročiti in plačati.

4.6 Opredelitev kriterijev

Poglavje opisuje zajem podatkov, ki sem jih pridobil s pomočjo primarnih virov – dokumentacije podjetja in s poglobljenimi intervjuji z zaposlenimi v podjetju. Povzetek opišem s seznamom kriterijev, ki sem jih pridobil z analiziranjem podatkov ter ovrednotenjem uteži, ki so jih postavili zaposleni.

V intervjujih so sodelovali lastnik podjetja, direktor, vodja IT oddelka, vodja marketinga in prodaje, arhitekt programskih rešitev, dva analitika, svetovalec in dva razvijalca. Intervjuje sem opravil z vsako osebo ločeno, saj sem hotel ohraniti sogovornikovo subjektivnost. Osrednja tema intervjuja je bila implementacija oblaka za njihovo stranko.

Glede na izvedbo je mogoče implementirati zasebni, hibridni oblak ali izbrati javni oblak. Pri slednjem je treba upoštevati kriterije za izbiro najboljšega ponudnika glede na stanje in zahteve njihove stranke. Izvedbe hibridnega oblaka nisem obravnaval ločeno, saj predstavlja logično unijo kriterijev izbire privatnega in javnega oblaka skupaj.

Najbolj tipičen kriterij, ki so se ga dotaknili vsi zaposleni, so **stroški**. Poudarek je na primarnih stroških, bodisi da je to najemnina ali nakup strojne opreme v primeru izvedbe na način privatnega ali hibridnega oblaka. Pri tem je pomembna formula amortizacije nakupa računalniških virov skozi leta in s tem možna primerjava (mesečnega) najema javnega oblaka. Sam izračun amortizacije so v podjetju že opredelili in poleg primarnih virov vključuje še infrastrukturo za zagotavljanje varnostne kopije, vzdrževanje celotne infrastrukture in oceno nujnega nadgrajevanja v določeni dobi. Podrobnosti formule presegajo okvirje mojega dela in je tu ne bom navajal. Delitev stroškov je bila pri vodstvenem kadru bolj podrobna v primerjavi z ostalimi zaposlenimi. Izpostavljen je bil strošek za vzpostavitev vzorčnega (angl. *demo*) okolja. Vzorčna okolja so nujna za potrebe faze prodaje pri vzpostavljanju sinergije s potencialno stranko.

Pri stroških so navedli pomembnost modela obračuna pri najemu storitve. V prvi vrsti so izpostavili domače ponudnike, ki v večini ne ponujajo pravega obračuna po uporabi, temveč imajo na voljo le časovno opredeljene (npr. mesečne) najemnine glede na izbrano okolje. To je v primeru privatnega oblaka nepomembno, saj gre za namenski (angl. *dedicated*) najemni prostor. V primeru javnega oblaka pa bi si želeli model obračuna po uporabi. Pri tem jim je pomembna enota obračuna po uporabniku, po uporabi pomnilnika in procesorja, količini prenosa podatkov ipd. Manjša kot je enota obračuna, bolj natančen je obračun, nižji so stroški najema.

Vodstvenemu kadru je poleg stroškov pomembna še **kakovost** storitev, ki jo lahko zagotovijo sami s ponujenim zasebnim oblakom ali najemom javnega oblaka. Raven abstrakcije kakovosti je pri vodstvenem kadru višja kot pri arhitektih in razvijalcih. Vodilnim je pomembno, da ja raven zagotavljanja varnosti podatkov najvišja možna za normalno okvirje IT poslovanja. Za normalno raven opredeljujem poslovne rešitve za podjetja, ki nimajo posebnih zakonskih predpisov, kot npr. banke in zavarovalnice.

Na področju kakovosti so vodilni izpostavili še zanesljivost sistema in delovanje sistema v nepovezanem načinu. Tehnični del zaposlenih, predvsem arhitekti pa so kakovost podrobneje razdelili in navedli tudi naslednje pomembne elemente kakovosti, razvrščene od najbolj do najmanj pomembnega:

- razpoložljivost sistema,
- ponovna vzpostavitev,
- elastičnost storitve,
- šifriranje podatkov (na mestu hrambe kot tudi med prenosom),

- odzivnost.

Večina meni, da po pomembnosti kakovosti sledijo **funkcionalni** kriteriji storitve. Tu je bila raven abstrakcije vodilnih uslužbencev še višja v primerjavi z arhitekti. Prav tako vodilni kader to področje po pomembnosti enači s kakovostjo, z razliko od ostalih. Intervjujanci so si skoraj enotni, da je najpomembnejši kriterij funkcionalnosti uporabnost storitve. Analitiki in arhitekti so takoj za tem postavili še:

- prilagodljivost ponudnika tako na področju izvedbe storitev kot tudi prilagajanje pri obračunu stroškov. Prilagodljivost Garg et al. (2013, str. 1016) opredelijo kot zmožnost (in čas) ponudnika, da se odzove in prilagodi zahtevam najemnika oblaka. Za to lastnost Leong, Toombs in Gill (2015) v poročilu za hišo Gartner navajajo, da je ena izmed pomembnih atributov, ki naj bi jih ponudnik dosegal pri prodiranju na trg.
- Enostavnost storitve, ki nudi uporabniku hitrejše razumevanje uporabe in s tem zmanjšanje stroškov pri uvajanju in izobraževanju kadra.

Pravnim in ekološkim kriterijem intervjujanci niso namenili posebne pozornosti. Razlog za to je, da programska rešitev, ki jo tržijo ne zahteva posebnih zakonodajnih uredb. Zakonska določila, katera morajo spoštovati, kot je npr. zakon o varstvu osebnih podatkov, so po mnenju intervjujancev samoumevna in jih zato posebej ni treba navajati. Za kriterije ekologije so enotni, da so iz poslovnega in konkurenčnega stališča najmanj pomembna, čeprav vseeno dobrodošla, predvsem pri ponudnikih javnih oblakov – uporaba obnovljivih virov, racionalna poraba energije ipd.

4.7 Analiza ponudnikov računalništva v oblaku

Pri analizi ponudnikov storitev oblaka se v svojem delu osredotočim na ponudnike IaaS. Globalni trg ponudnikov IaaS je огromen, število ponudnikov je veliko in njena rast se zelo hitro večja (Mor, 2016).

Analizo trga sem razdelil v tri sklope: domači ponudniki, ponudniki v Evropi in ostali svetovni ponudniki storitev. Slednji sklop zavzamejo največji in najbolj razširjeni ponudniki, ki imajo svojo infrastrukturo razporedeno po več celinah.

Vhodne kriterije, ki opredelijo začetno množico alternativ, sem izbral glede na različne parametre:

- zahteve strank za specifične ponudnike (najpogosteje največji ponudniki),
- zahteve strank po ponudnikih znotraj države poslovanja (za tovrstne potrebe sem izbral evropske ponudnike),
- domače ponudnike,

- ponudnike, kateri so že bili predmet analize v podjetju študije primera ali z njimi že sodelujejo (lahko preko svojih partnerjev) in
- ponudnike, ki so jih priporočili njihovi poslovni partnerji.

Analizo sem izvedel s pomočjo spletnih strani ponudnikov, ki navadno ponujajo podrobne opise vseh storitev, podkrepljene z video vsebino. Za ponudnike, ki imajo na voljo brezplačna okolja sem vzpostavil testno okolje in s pomočjo tega izvedel nekaj osnovnih testov kot so možnost migracije in orodja za pomoč pri prehodu na oblak, uporaba različnih orodij, namestljivost lastnih orodij, enostavnost uporabe, integracija (predvsem z lokalno strojno opremo – npr. tiskalnik), hitrost prenosa podatkov, razdelitev prometa in elastičnost (kjer je bilo to omogočeno v okviru brezplačnega okolja). Ostala tehnična merjenja parametrov presegajo okvirje mojega dela in jih zato izpuščam. Pri analizi sem si pomagal tudi s spletno literaturo, objavljeno z novejšim datumom objave in podatki opravljenih analiz v podjetju.

V nadaljevanju opisujem izbrane ponudnike, razdeljene po prej omenjenih sklopih: domači, ponudniki v Evropi in vodilni, svetovni ponudniki.

4.7.1 Slovenskih trg ponudnikov Iaas

Razvoj storitev oblaka na slovenskem trgu je v rasti, čeprav je v primerjavi s svetovnih trgom še v začetnih fazah, a pri tem dobrih idej ne manjka (Mavrič, 2011). Pri tem je največ storitev tipa SaaS in IaaS. Analiza domačega trga je bila trši oreh v primerjavi s ponudniki na svetovnem trgu, saj je dostopnost do podatkov preko spleta bolj skopa. Kot navaja Mavrič (2011, str. 37.) je verjetno težava v tem, da večina ponudnikov svoje storitve še ne zna primerno tržiti in da je razpoznavnost storitve težava tako na lokalni kot tudi na globalni ravni.

Leta 2009 je bil ustanovljeno združenje EuroCloud, katere članica je tudi Slovenija. Razlogi za ustanovitev združenja so združiti znanje iz različnih držav na področju oblaka, si medsebojno pomagati pri nadaljnjem razvoju in v splošnem spodbujati razvoj tehnologije oblaka. Vzpostavitev novih poslovnih stikov in njihova promocija je nujna za prodor na širši trg oblaka (Eurocloud – Spletna stran Zavoda E-Oblak, 2016). Za dodatno spodbujanje k razvoju združenje vsako leto razglasi naziv za najboljšo in najbolj inovativno storitev. Združenje je izdalo certifikat, ki opredeljuje varnost oblaka in zagotavljanje varstva zasebnih podatkov v njem. Tovrstni certifikati vsekakor zagotavljajo boljšo kakovost storitev, ki jih nudijo ponudniki oblaka.

V nadaljevanju je v preglednici Tabela 4 na kratko opisanih nekaj slovenskih IaaS ponudnikov.

Tabela 4: Seznam slovenskih ponudnikov IaaS

Ponudnik	Spletni naslov	Opis
Si.Mobil	www.simobil.si	Kapacitete se opredelijo na mesečni ravni, ki se lahko po potrebi spreminjajo. Nudijo možnost namestitve različnih operacijskih sistemov. Obračun se izvaja na mesečni ravni. V primeru najema storitev za daljše obdobje ponudnik nudi dodatne popuste.
Pošta Slovenije	www.posta.si	Ponudnik sodeluje s podjetjem HP in uživa njihov status HP CloudAgile partnerja. Njihovi podatkovni centri so znotraj Slovenije in imajo visoko stopnjo fizične varnosti. Ponujajo vertikalno in horizontalno razširljivost kapacitet virov po uporabi. Obračun stroškov je lahko mesečna najemnina ali po uporabi z enoto porabe na dan. Ponudnik omogoča vzpostavitev testnega okolja s predhodno registracijo.
NIL HyperCenter	http://www.nil.com/	Njihovo ogrodje so poimenovali Flex IT. Ponujajo prilagodljivo arhitekturo delovnim procesom stranke in zagotavlja učinkovitejše delovne procese. Navajajo visoko odzivnost storitev in kontinuiteto delovanja. Kredibilnost procesov dokazujejo s pridobljenima ISO 9001 in ISO 27001 certifikatoma. Enota obračuna je na mesečni ravni. Poleg tipične strojne opreme nudijo tudi najem omrežnih komponent, uporabna orodja in svetovanje pri zasnovi arhitekture.
Tuš Hosting	http://www.tushosting.si/	Ponudnik ponuja v naprej dobro definirane pakete storitev. Zagotavlja nemoteno delovanje v enem najmodernejših podatkovnih centrih s fizičnim varovanjem pri nas. V sodelovanju z domačimi in tujimi internet ponudniki zagotavlja zanesljivo povezljivost.

V sklopu omejenih virov in razpoložljivega časa pri slovenskih ponudnikih ni bilo mogoče vzpostaviti testnega okolja preko spleta. V primerjavi s svetovnimi ponudniki je pridobitev podrobnejših informacij veliko težja oz. te niso javno objavljene. Ponudniki vedno nudijo možnost prilagajanja glede na potrebe stranke, kar je v primerjavi z vodilnimi ponudniki tovrstnih storitev poglavitna prednost. Prednost domačih ponudnikov je tudi v prilagajanju zakonodaji, predvsem glede varnosti osebnih podatkov, saj je domača zakonodaja glede tega strožja od ameriške (Mavrič, 2011).

4.7.2 Evropski trg ponudnikov IaaS

Evropski trg je v primerjavi s svetovnim v precejšnjem zaostanku (McCallion, 2012). Kljub temu analitična hiša IDC napovedujejo hitrejšo rast evropskega trga v primerjavi s svetovnim (Darrow, 2012). Evropski trg se v primerjavi z ameriškim zelo razlikuje, kar se odraža tudi na trgu oblaka. Pri primerjavi obeh trgov Darrow (2012) navaja naslednje glavne razlike:

- ameriški ponudniki na Evropo gledajo kot monoliten trg, vendar ga sestavlja veliko različnih držav, kultur in jezikov. Tako je manjšim evropskim ponudnikom prilaganje na dokaj razdrobljen trg lažje.
- Dosedanji vzorci razvoja IT tehnologije so pri oblaku precej drugačni, kot je bilo to pri hitri rasti drugih tehnologij (npr. mobilne telefonije).
- Zakonodaja, ki govori o varnosti osebnih podatkov, je v nekaterih državah precej bolj stroga kot izven Evropske unije (v nadaljevanju EU). Kot navaja Pohorec (2011) »Zakonodaja EU predvideva, da se določeni podatki hranijo na lokaciji znotraj EU.«. Nekateri celo obsojajo določene države, kjer zakon določa, da morajo biti osebni podatki shranjeni lokalno, da zavirajo rast oblaka in ob tem dodajajo, da je nujno potrebno razlikovanje med privatnim in javnim oblakom in zmožnostjo delitve opravil med enim in drugim.
- Več kot polovica vseh implementacij oblaka je uporabila privatni oblak, kar je posledica zakonodaje omenjene v predhodni točki. Z ureditvijo tega zakona bodo prehodi na oblak tudi s strani strank manj stresni.

Glede na dano primerjavo zgoraj omenjeni avtor dodaja še, da če so storitve SaaS primerljive, je razlika toliko bolj očitna pri IaaS. Glavni razlog je vsekakor, da v Evropi ni ponudnika s tako veliko in obširno infrastrukturo kot jo ima npr. Amazon, razen Amazon sam.

Razliko glede vrste storitve kaže tudi raziskava, v kateri je sodelovalo več kot 1000 evropskih ponudnikov, od katerih le četrtnina nudi IaaS in samo 4% PaaS storitve. Ostalo pripada storitvam SaaS. Pri tem pa je vseeno treba poudariti, da je več kot polovica SaaS ponudnikov sklenilo partnerstvo z drugimi IaaS ponudniki. Med njimi je več kot polovica partnerstev znotraj lastne države, dobra četrtnina znotraj EU, slabih 20% pa ima partnerstvo sklenjeno s ponudnikom v ZDA (Bilderbeek, 2014).

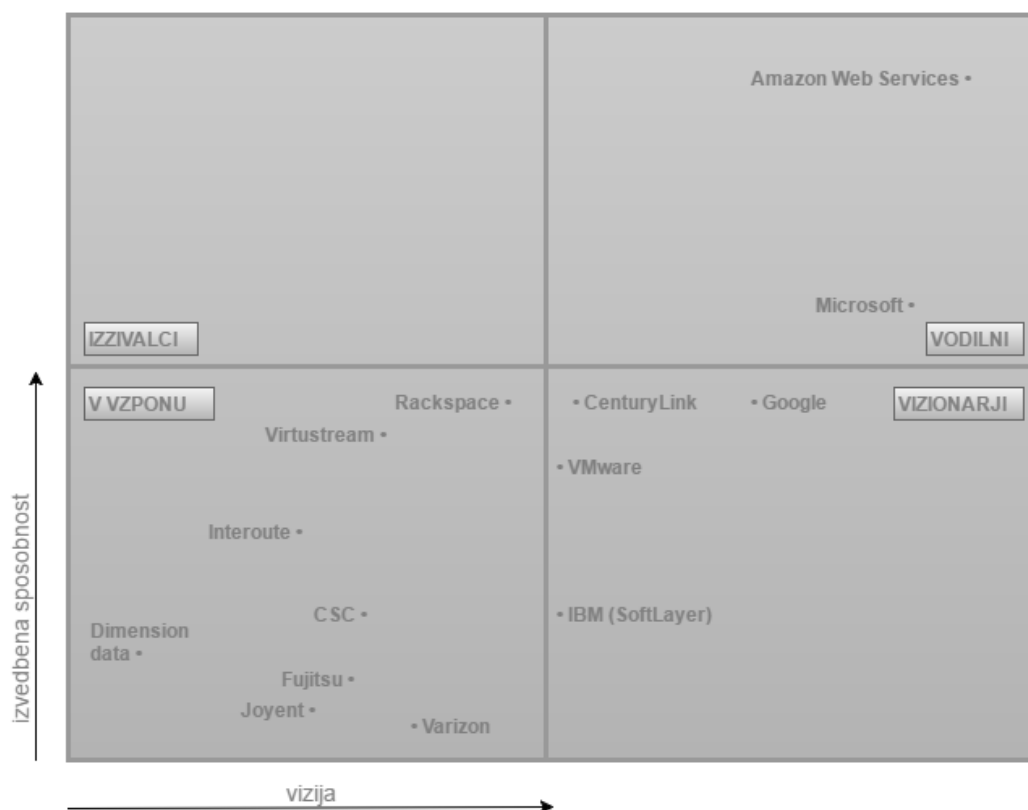
4.7.3 Vodilni ponudniki IaaS

Globalni trg ponudnikov storitev IaaS se neprestano širi in dotok novih ponudnikov je konstanten. Pri tem pa na vrhu še vedno prevladujejo posamezni ponudniki, t. i. veliki štirje (angl. *big four*), Amazon, Microsoft, IBM in Google (Mor, 2016). Njihov delež

obsega več kot polovico (54%) celotnega svetovnega trga oblaka IaaS (Preimesberger, 2015).

Letno poročilo hiše Gartner (Leong et al., 2015) vsebuje graf vodilnih ponudnikov storitev IaaS kot ga prikazuje Slika 8. Graf v prvem kvadrantu (ponudniki v vzponu) prikazuje ponudnike, ki imajo dobro zastavljeno vizijo in široko zastavljen ter ambiciozen načrt. To so navadno novi ponudniki na trgu ali pa ponudniki, ki še niso uspeli doseči želenega deleža trga. V naslednjem kvadrantu so vizionarji, ki so lahko novo nastopajoči ali že povsem uveljavljeni ponudniki, katerih razvoj je v velikem razmahu. Njihova glavna značilnost je vizija prihodnosti in veliki vložek v razvoj tovrstne tehnologije, kar je velik doprinos k splošni funkcionalnosti teh storitev. V kvadrantu izzivov so že uveljavljeni ponudniki, ki ponujajo uveljavljene in kakovost storitve. V letnem poročilu 2015 Gartner ne navaja nobenega ponudnika. Zadnji kvadrant je namenjen vodilnim ponudnikom kot sta Amazon in Microsoft. Čeprav morda ne ponujata katere specifične storitve ali ne zadoščata katerim določenim področjem, ju odlikuje ambiciozen načrt, kakovostna ponudba storitev, velik tržni delež in veliko referenčnih strank.

Slika 8: Vodilni ponudniki storitev IaaS



Vir: L. Leong, D. Toombs in B. Gill, *Magic Quadrant for Cloud Infrastructure as a Service, Worldwide*, 2015.

Raziskava in primerjava vodilnih ponudnikov IaaS, ki jo je izvedla TechRepublic (Rodrigues, 2013) kaže na veliko izboljšanje storitev ponudnikov IaaS. V splošnem je zaznati dokaj podobne rezultate med ponudniki. Pri tem pa je zanimivo to, da v nekaterih za IaaS oblak pomembnih kriterijih, vodilni ponudniki ne zasedajo najboljših rezultatov (npr. zanesljivost). Druga stran je (mesečni) strošek storitve, kjer vodilni ponudniki ponujajo očitno najugodnejše cene v primerjavi z ostalimi.

Amazon in Microsoft veljata za vodilni podjetji na področju oblaka, vendar pa je pri medsebojni primerjavi Amazon v veliki prednosti pred Microsoftom. Lindič (2007, str. 87) navaja, da je Amazon najpogosteje uporabljen primer podjetja, tako v raziskovalni kot tudi poslovni sferi.

AWS je hčerinsko podjetje podjetja Amazon.com, ki je osredotočeno na področje oblaka in ima zelo jasno opredeljeno vizijo o visoko avtomatiziranih, stroškovno učinkovitih, fleksibilnih in na zahtevo dostavljenih storitvah (Leong et al., 2015). Glavna storitev je The Elastic Compute Cloud – EC2, ki nudi zanesljiv in prilagodljiv najem računalniških virov z modelom obračuna po uporabi (Amazon Web Services (AWS) – Cloud Computing Services, 2016). Poleg omenjene storitve Amazon nudi še celo vrsto storitev in orodij kot npr. različna skladiščenja objektov, upravljanje s podatki in podatkovnimi bazami, upravljanje z omrežji, delitev prometa in nalog, migracijska orodja itd.

Struktura njegovih podatkovnih centrov je razširjena po več celinah. Največ jih je v ZDA, kjer je prav tako lokacija posebnega centra, ki deluje za potrebe ameriške vlade. Poleg omenjenih ima Amazon podatkovne centre še v Nemčiji, na Irskem, Japonskem, v Singapurju, Avstraliji, Braziliji ter na Kitajskem (Leong et al., 2015). Za razliko od drugih velikih korporacij Amazon nerad razkriva podrobnosti o svojih centrih. Že točno število podatkovnih centrov je vprašljivo, saj navajajo le regije, v katere so podatkovni centri dodeljeni. Znotraj ene regije delujejo t. i. razpoložljiva območja (angl. *availability zones*), v kateri pa deluje peščica podatkovni centrov (Burrington, 2016).

Kljub temu, da Amazon ni izbira za prav vsako podjetje, ima največ izkušenj, tako z malimi in srednjimi podjetji kot tudi z velikimi korporacijami. Gostuje celo časovno kritične sisteme. Velja za »varno izbiro« ponudnika, predvsem za uporabnike, ki želijo širok spekter funkcionalnosti oblaka (Leong et al., 2015). Kljub največji infrastrukturi se je tudi Amazon v preteklosti spopadel s težavami in doživel daljše izpade. Eden večjih izpadov se je zgodil leta 2011, ko je podatkovna shramba, kjer je po podatkih analitikov delovalo več kot 500.000 instanc, napačno zaznala, da replikacija podatkov ni sinhronizirana. Posledično se je začela sočasna sinhronizacija vseh instanc, kar je privedlo do preobremenjenosti sistema. Čeprav se je to zgodilo samo v enem razpoložljivem območju, je to vplivalo tudi na ostala območja, kar se po zasnovani arhitekturi ne bi smelo zgoditi. Izpad so delno rešili na nekaterih lokacijah v 3 urah, v celoti pa šele po 40 urah (Pohorec, 2011).

Microsoft je sprva ponujal le storitve PaaS in šele od leta aprila 2013 dalje je razširil svojo ponudbo tudi na IaaS (Leong et al., 2015). Svojo storitev imenuje Microsoft Azure Infrastructure Services, ki vključuje najem strojne opreme in omrežnih komponent. Glede na svoj pozen vstop na trg IaaS je presenetljivo hitro osvojil trenutni položaj in po navajanju Stotlerjeve (2015) ima najvišjo rast prihodkov v zadnjem obdobju. Stotlerjeva navaja še ugodno napoved za Microsoft, saj 42% podjetij, ki namerava izvesti prehod na oblak razmišlja o izbiri rešitve Microsoft Azure in od tega jih polovica meni, da je Microsoft najpomembnejši ponudnik ta trenutek. Mor (2016) navaja, da je eden izmed glavnih razlogov za takšno rast Microsofta to, da je za stranke, ki so do sedaj že uporabljale njihovo okolje (Microsoft Windows), prehod na oblak bolj naraven in manj stresen. Dodaja še, da razvoj funkcionalnosti Microsoft Azure uspešno sledi Amazonovemu EC2, vključujoč elastičnost in skalabilnost instanc, podatkovno obdelovanje in shrambe podatkov ter integracije z ostalimi sistemi.

Ravno zaradi Microsoftovega agresivnega prodora na trg, Leong et al. (2015) opozarjajo na pazljivost, da gre za relativno mlad in še vedno nastajajoč sistem.

Microsoft svoje podatkovne centre poimenuje »regije«. Tudi teh je največ v ZDA, kjer dva delujeta za potrebe vladnih organizacij. Podatkovne centre imajo nameščene še na Irskem, Nizozemskem, v Avstraliji, Hong Kongu, na Japonskem, v Singapurju in Braziliji. Njihov podatkovni center nameščen na Kitajskem upravlja podjetje 21Vianet Group.

5 IZDELAVA MODELA ZA IZBIRO PONUDNIKA RAČUNALNIŠTVA V OBLAKU S POMOČJO METODOLOGIJE ANALITIČNO HIERARHIČNEGA PROCESA (AHP)

Hitra rast trga, neprestani razvoj tehnologije, dosegljivost interneta in inovativne ideje izjemno povečujejo število ponudnikov in njihovih storitev na trgu oblaka. Po napovedih hiše Gartner (2015) je največjo rast pričakovati na področju marketinga, e-poslovanja in naprednih analitičnih rešitev. Trenutno kazalci kažejo, da se bo tržno področje oblaka še naprej naglo širilo. Ocena IDC (2014) je, da bo leta 2018 javni oblak zasedal več kot polovico svetovne programske opreme, strežnikov in pomnilniškega prostora.

Kljub temu, da so primarni cilj trga oblaka mala in srednje velika podjetja (Chorafas, 2011, str. 34), vanj kot stranka oz. uporabnik vstopa vse več velikih podjetij. Stereotip, da na oblak vstopajo le *start-up* podjetja je preteklost (Olavsrud, 2016). Po navajanju hiše Gartner (2015) skoraj polovica velikih podjetij, ki so sodelovala v raziskavi, med pet najpomembnejših prioritet uvršča modernizacijo obstoječe (tradicionalne) infrastrukture na mestu (angl. *on-premises*). Vstop velikih podjetij sili ponudnike k širjenju in še hitrejšemu razvoju kot sicer. Ne redko to lahko celo vpliva na strategijo ponudnika oblaka.

Vse našteto naredi trg ponudnikov kompleksen in medsebojno neprimerljiv. Uporabniki morajo biti zaradi tega pri izbiri ponudnika zelo previdni (Leong et al., 2015). Izbira ponudnika storitve je kompleksna in zahtevna ter za odločevalca o izbiri predstavlja pravi izziv (Garg et al., 2011, str. 210). Izziv je zanimiv, tako za akademske raziskovalce, kot tudi za odločevalce v poslovni sferi.

Po navajanju Sun et al. (2014, str. 135) je celotni izziv sestavljen iz dveh sklopov problemov, katere je potrebno rešiti. Prvi predstavlja vzpostavitev standardnih vmesnikov in v naprej definiranih formatov storitev. Ti bodo pripomogli k vzpostavitvi skupne infrastrukture trga oblaka, kjer bo možno trgovanje storitev med različnimi tipi kupcev. S tem bo ponudnikom omogočena objava storitev na enotni platformi, uporabnikom pa njihova izbira. Za to je potrebno dobro razumevanje, znanje o domeni oblaka na visoki ravni ter vzpostavitev standardnih metod za opis storitve in možnost njihove medsebojne primerjave. Drugi del izziva predstavlja izbira posameznega ponudnika s pomočjo različnih tehnik odločanja.

V ta namen podjetja potrebujejo zanesljivo metodo za lažji izbor »pravega« ponudnika, ki bo zadostil vsem njihovim potrebam. Pri tem so odločevalci izpostavljeni določenemu stresu (Gupta, Forgionne, & Mora, 2006, str. 4), zato je pomembno, da je model enostaven, razumljiv in učinkovit. S pomočjo takšnega modela je odločanje lažje, hkrati pa ovrednoti in utrdi tezo, ki jo mora odločevalec doseči. S tem je odločitveni proces manj stresen.

V naslednjih podpoglavjih sprva opišem, kaj pomeni večparametrsko odločanje in predstavim v tovrstnih modelih pogosto uporabljeno metodologijo analitično hierarhičnega procesa ali krajše AHP. Metodo opišem in jo nato uporabim pri izdelavi svojega modela. Za izdelavo modela poiščem sintezo kriterijev iz literature in kriterijev, ki sem jih opredelil pri svojem primeru študije. Opišem novo dobljeni model. Kriterijem novega modela določim pomembnosti pri čemer sem upošteval sintezo med opredeljeno pomembnostjo kriterija v literaturi in zaznano ravni pomembnosti iz študije primera. Dotaknem se tudi analize odločitve na podlagi modela in analize sprememb vhodnih podatkov ali modela.

5.1 Opis metodologije analitično hierarhičnega procesa – AHP

»Odločanje je proces za identifikacijo stvari, ki najbolj izpolnjujejo cilje, želje in omejitve odločevalcev.« (Sun et al., 2014, str. 136 v Zhang, Ranjan, Nepal, Menzel, & Haller, 2012).

»Odločanje je *aktivnost*, ki nastopa kot del splošnega reševanja problemov na praktično vseh področjih človekovega delovanja. Težavnost odločitvenih problemov je zelo raznolika. Segra od enostavnih osebnih odločitev, ki so večinoma rutinske in se jih večinoma niti ne zavedamo, vse do zahtevnih problemov skupinskega odločanja [...]« (Bohanec, 2006, str. 3).

Hatamura (2006, str. 1) opredeli odločanje kot proces, kjer imamo več alternativ, da se nekaj zgodi. Pri tem pomislimo na verjetnost posamezne alternative in določimo možnost izvedbe 100% za eno in 0% za vse ostale.

Za odločitveni proces Bohanec opredeli 7 aktivnosti, ki se razvrstijo skozi 5 faz tega procesa. Aktivnosti, ki jih vključuje odločitveni proces so (Bohanec, 2006, str. 3):

- spoznavanje odločitvenega problema,
- zbiranje in preverjanje informacij,
- identifikacija alternativ,
- predvidevanje posledic odločitve,
- odločitev – izbira alternative na osnovi zbranih informacij,
- sklep o odločitvi in
- vrednotenje odločitve,

katere se izvedejo v odločitvenem procesu skozi naslednjih 5 faz (Bohanec, 2006, str. 19–25):

- identifikacija odločitvenega problema,
- identifikacija alternativ,
- razgradnja problema in modeliranje,
- vrednotenje, analiza in izbira alternativ in
- realizacija odločitve.

V postopku odločevanja so prisotne različne motnje. Te so objektivne narave kot npr. nepopolni podatki odločitvene domene, slabo opredeljene alternative, neučinkovit odločitveni model ter subjektivne narave kot je pristranskost ali neodločnost. Hatamura (2006, str. 8) navaja neodločnost kot najslabše stanje v odločitvenem procesu. Po sprejetju odločitve odločevalca velikokrat spremlja dvom. Normalno je, da je odločitev sprejetje določene ravni tveganja, vendar lahko dvom omilimo s tem, da izberemo ali izdelamo ter uporabimo učinkovit in zanesljiv odločitveni model. Z učinkovitim modelom zmanjšamo dvome in ostale moteče faktorje, vključno z neodločnostjo. Bolj kot odločevalec verjame v zanesljivost modela, manj bo prisotnih motečih faktorjev.

5.1.1 Večparametrsko odločanje

Odločitve se delijo glede na različne parametre. Ločimo jih po tem, ali je odločanje enkratno ali ponavljajoče, kakšna je zahtevnost, koliko odločevalcev je prisotnih, po ravneh tveganja, po številu kriterijev ipd. V odvisnosti od števila prisotnih kriterijev poznamo enostavnejše enoparametrške (ali enokriterijske) in kompleksnejše večparametrške (ali več večkriterijske) odločitve – MCDM.

Pri enokriterijskih odločitvah je v odločitvenem procesu vsebovan le en kriterij. V odločitvenih procesih, kjer je opredeljen le en kriterij je najpogosteje uporabljen kriterij povezan s financami (Bohanec, 2006, str. 32–33; Pohekar & Ramachandran, 2004, str. 366). Ta je tipično čim večji dobiček, minimalen strošek, želen prihodek, čim boljša donosnost investicije ipd. Redkeje se uporabijo drugi kriteriji, ne da bi se pri tem navezovali še na ostale kriterije, npr. geografska lokacija podjetja.

Kadar je odločitveni model pri izbiri alternativ sestavljen iz več kot enega kriterija je to opredeljeno kot večparametrsko odločanje. Pri tem je neodvisno, ali so kriteriji medsebojno odvisni ali ne. Števila kriterijev in tudi alternativ je pri tem končno mnogo (Triantaphyllou, 2000, str. xxv). Na primer v primera kriterija financ iz prejšnjega odstavka bi bilo pri najemu oblaka smotno dodati kriterije kot so zanesljivost storitve in varnost podatkov. Praktična uporaba večparametrskega odločanja je pogostejša v primerjavi z enoparametrskim odločanjem (Bohanec, 2006, str. 32).

5.1.2 Hierarhični večparametrski model

V praktični uporabi je večparametrsko odločanje kompleksno, saj že »enostavni« problem, kot je nakup osebne ali namiznega računalnika, hitro privede do tolikšnega števila kriterijev, da celotna slika zastavljenih zahtev, kriterijev, parametrov ali lastnosti postane nejasna. V ta namen je smiselno izdelati odločitveni model ali več njih. Odločevalec si v fazi razgradnje problema in modeliranja izdelava model.

Model je sestavljen iz funkcije ovrednotenja alternative. Funkcija za vhodne spremenljivke sprejme seznam opredeljenih lastnosti določene alternative in kot rezultat vrne vrednost, ki vhodno alternativo umesti v urejen seznam vseh alternativ glede na zastavljene kriterije. Odločevalec bo na podlagi dobljenega seznama izbral najboljšo alternativo med vsemi, več njih ali pa urejen seznam uporabil za nadaljnje potrebe. Pri tem Saaty (v Podobnik, 2015, str. 43) opozarja, da izbira najboljših alternativ ni vedno dobra.

Pri linearni strukturi, kjer so kriteriji le naštet v vrstnem redu, lahko tudi razvrščeni po določenem zaporedju, brez medsebojne povezave in že pri majhnem številu kriterijev človeku nerazumljivi, je priporočljivo uporabljati strukturirane odločitvene modele.

Pri analizi problema je njegova razčlenitev na manjše enote glavno orodje znanstvenih raziskav pri testiranju hipotez in reševanju problemov (Saaty & Vargas, 2006, str. 1). Pri strukturiranih odločitvenih modelih razčlenitev problema (kriterije) uredimo v **hierarhijo**. Razmerje nadrejen–podrejen označuje odvisnost med njimi. Element lahko vsebuje več podrejenih kriterijev, ki podrobneje opredelijo nadrejeni element. Primer nadrejenega elementa pri najemu oblaka so stroški. Podrejeni elementi stroškov so: mesečna osnovna najemnina, najemnina za varnostne kopije, strošek v primeru migracije k drugemu ponudniku, strošek najema licenc ipd. Vozlišča, ki imajo podrejene elemente imenujemo

izpeljani, liste drevesa pa osnovni elementi (Bohanec, 2006, str. 104). Osnovni elementi služijo kot vhodne spremenljivke v funkcijo odločitvenega modela za posamezno alternativo. Posebni izpeljani element je koren, ki opisuje končno ovrednotenje alternative.

Model naj bo odločevalni skupini čim bolj razumljiv. V ta namen se pogosto namesto matematičnega zapisa uporablja grafični prikaz najpomembnejših komponent modela (Bohanec, 2006, str. 21–22):

- **struktura odločitvenega problema**, kjer odločevalec skuša problem razgraditi na manjše, človeku razumljivejše (pod-)probleme. S tem se poveča razumevanje problema. Pri tem je treba upoštevati ali so razdrobljene enote med seboj odvisne (zaporedno odločanje) ali so vzporedne in med seboj neodvisne. Dobljene manjše enote je možno opisati hierarhično v večparametrskem odločitvenem modelu.
- V modelu opredelimo **kriterije**, ki služijo pri izbiri alternativ. Pri tem je treba biti pozoren, da so v model vključeni kriteriji, ki so bistveni pri določanju alternative (načelo polnosti). Ob tem je za kriterije pomembno še, da model ne vključuje odvečnih kriterijev, ki so za izbiro alternative nepomembni, da so medsebojno neodvisni in da so merljivi. Kriterijem se določi zaloga vrednosti in opredeli, katere vrednosti so najslabše in najboljše.
- Merljivost kriterijev je odvisna od narave kriterija. Enostavnejše je objektivno merjenje kriterija, kjer je ta opredeljen z določeno vrednostjo in enoto (npr. cena določene storitve ali artikla). Mehke (angl. *fuzzy*) ali opisne kriterije je potrebno sprva ovrednotiti (npr. 1 predstavlja »slabo« in 10 »odlično«). Pri tovrstnih kriterijih ne gre za natančno ocenjevanje, temveč za subjektivno določanje vrednosti, ki služi le za medsebojno primerjanje alternativ – npr. uporaba zaloge vrednosti: slabo, zadovoljivo, odlično (Schniederjans, Hamaker, & Schniederjans, 2010, str. 190).
Tovrstne kriterije Bohanec poimenuje **preference**. Sam termina v svojem delu ne bom uporabljal. Preference obravnavam enako kot ostale merljive kriterije, saj je pomembnost kriterija neodvisna od te lastnosti.
- **Negotovost** ali **tveganje** je komponenta modela, do katere lahko pride pri odločitvi. Odločevalna skupina si mora odgovoriti na vprašanje, ali obstaja pojav ali aktivnost, ki (bistveno) vpliva na njihovo odločitev in nanjo ne more vplivati.

Pri izdelavi hierarhičnega večparametrskega modela je potrebno opredeliti še uteži in funkcijo združevanja ali vrednotenja kriterijev. **Utež kriterija** predstavlja njegovo pomembnost v danem problemu. Uteži je najenostavneje opredeliti z naravnimi števili v naprej določenem intervalu s strani odločevalca. Pomembno je, da so uteži znotraj intervala linearno porazdeljene. Določiti je potrebno tudi, katera vrednost opisuje najpomembnejši in katera najmanj pomemben kriterij.

Funkcija združevanja ovrednoti nadrejeni kriterij na podlagi podrejenih. Pogosto se za združevanje uporablja uteženo vsoto kot prikazuje enačba (2), v kateri $u(A)$ predstavlja

izračunano utež nadrejenega elementa, n število podrejenih kriterijev, w_i utež posameznega podrejenega kriterija in t_i njegovo izmerjeno vrednost.

$$u(A) = \sum_{k=1}^n w_i t_i \quad (2)$$

5.1.3 Analitično hierarhični proces – AHP

AHP metoda je ena najbolj znanih in najbolj uporabljenih metod v večparametrskem odločanju (Bohanec, 2006, str. 119; Sun et al., 2016, str. 43; Garg et al., 2013, str. 1017). Metodo je razvil dr. Thomas L. Saaty v letih 1970 kot pripomoček za reševanje tehniških in menedžerskih problemov.

Saaty in Vargas (2012, str. 1) opredelita AHP kot osnovni pristop k odločanju, kjer je vključena racionalna in intuitivna komponenta za izbiro najboljše alternative. Pri tem odločitveni model temelji na odločevalčevih sposobnostih za parno primerjavo elementov strukture, na pridobljenih informacijah, izkušnjah, znanju in čustvih. Navajata še, da metoda dovoljuje nekonsistenco pri odločitvah in nudi sredstva za njeno izboljšanje.

»AHP (Analytic Hierarchical Process) je metoda večparametrskega modeliranja, pri kateri zajemanje uteži in osnovnih koristnosti alternativ poteka po metodi primerjave po parih. Uteži dobimo z medsebojno primerjavo parametrov, koristnosti alternativ pa z medsebojno primerjavo alternativ. Pri obeh vrstah primerjav definiramo matriko primerjav, v kateri z numeričnimi vrednostmi od 1 do 9 (in njihovimi obratnimi vrednostmi) definiramo pomembnost enega parametra ali alternative v primerjavi z drugimi. Uteži oziroma koristnosti alternativ dobimo bodisi z izračunom lastnih vrednosti matrike bodisi z nekoliko preprostejšim in manj natančnim izračunom, ki temelji na normirani matriki primerjav. Možno je tudi oceniti konsistentnost primerjav.« (Bohanec, 2006, str. 123).

V večparametrskem hierarhičnem odločanju se uporablja različne metode. Glede na naravo in kompleksnost problema, znanja odločevalca ter števila elementov, so nekatere metode bolj učinkovite, druge manj. Nekatere bolj uporabljene metode, ki sem jih zasledil v analizirani literaturi za tovrstne modele poleg AHP so MAUT, SAW in TOPSIS. V svojem delu sem za izdelavo modela za izbiro ponudnika oblaka za IaaS izbral metodo AHP. Razlogi za izbiro te metode, predvsem v primerjavi z ostalimi metodami, so:

- gre za uveljavljeno in preizkušeno metodo.
- Metoda je zaradi svoje fleksibilnosti (Garg et al., 2013, str. 1017) primerna za uporabo v tako heterogenem okolju kot so oblaki.

- Ker je problem možno razdeliti v manjše enote in ga grafično predstaviti s hierarhičnim grafom, daje strukturi preglednost in omogoča lažjo razumevanje problema (Saaty & Vargas, 2012, str. 2).
- Metoda je predvsem učinkovita, kadar je v modelu večje število kriterijev z manj alternativami (Bohanec, 2006, str. 122), kar je za izdelavo mojega modela ugodno.
- Omogoča preverjanje (ne)konsistentnosti odločitvenega modela (Garg et al., 2013, str. 1017; Saaty & Vargas, 2012, str. 1) in s tem morebitno modifikacijo odločitvenega modela (npr. odstranitev nepomembnih kriterijev ipd.).
- Na področju oblaka so nekatere lastnosti težko merljive. Hkrati je neprimerno izvajati neposredno primerjavo številčne vrednosti določenega kriterija, v kolikor ti vrednosti vključujeta še spremno besedo. Torej lastnost oz. kriterij dejansko ni enak, je pa posredno primerljiv. Parna primerjava med dvema elementoma tako omogoča verodostojno izbiro v skladu s celotno vrednostjo (vključno s komentarjem).
- Parna primerjava elementov omogoča primerjavo tako kvantitativnih kot tudi kvalitativnih elementov (Bohanec, 2006, str. 122; Garg et al., 2013, str. 1017). V svojem modelu uporabljam primerjavo obeh vrst elementov.
- Z metodo AHP je mogoče izračunati občutljivost modela glede na morebitne spremembe v strukturi in stabilnost odločitve glede na te spremembe.
- Za podporo metodologiji je na spletu mogoče najti dovolj programskih rešitev za pomoč pri izdelavi modela ter analizo stabilnosti odločitve – grafični prikazi spremembe rezultata kot vpliv spremembe vhodnih podatkov.

Tudi pri analizi obstoječih modelov za izbiro ponudnika oblaka sem zaznal, da je AHP metoda najpogosteje uporabljena metodologija v analiziranih modelih.

Čeprav je AHP učinkovita odločitvena metoda, je njena slabost prisotnost odločevalčeve subjektivnosti pri parni primerjavi alternativ (Whaiduzzaman et al., 2014, str. 6). Bohanec (2006, str. 122–123) navaja, da metoda ni primerna v primeru, ko imamo večje število alternativ, saj kompleksnost modela narašča kvadratno. Dodaja še, da zaradi zahtevnih računskih operacij za reševanje problema potrebujemo za to primerno programsko opremo. Pri tem je ugodno, da je na trgu veliko tovrstne programske opreme, ki je odločevalcem lahko dostopna. Prav tako metoda nudi odstopanja z uporabo enostavnejših matematičnih operacij, ki nudijo dovolj dobre približke za zadoščanje reševanja večine problemov.

Lee in Seo (2013, str. 365) opredelita šest korakov za reševanju kompleksnega problema z uporabo metode AHP, ki so podrobneje opisani v nadaljevanju:

- opredelitev problema in določitev ciljev ter rezultatov,
- razdelitev problema na manjše enote in izdelava hierarhične strukture z elementi odločanja – kriteriji in alternative,
- parna primerjava strukturnih elementov in sestava primerjalne matrike,

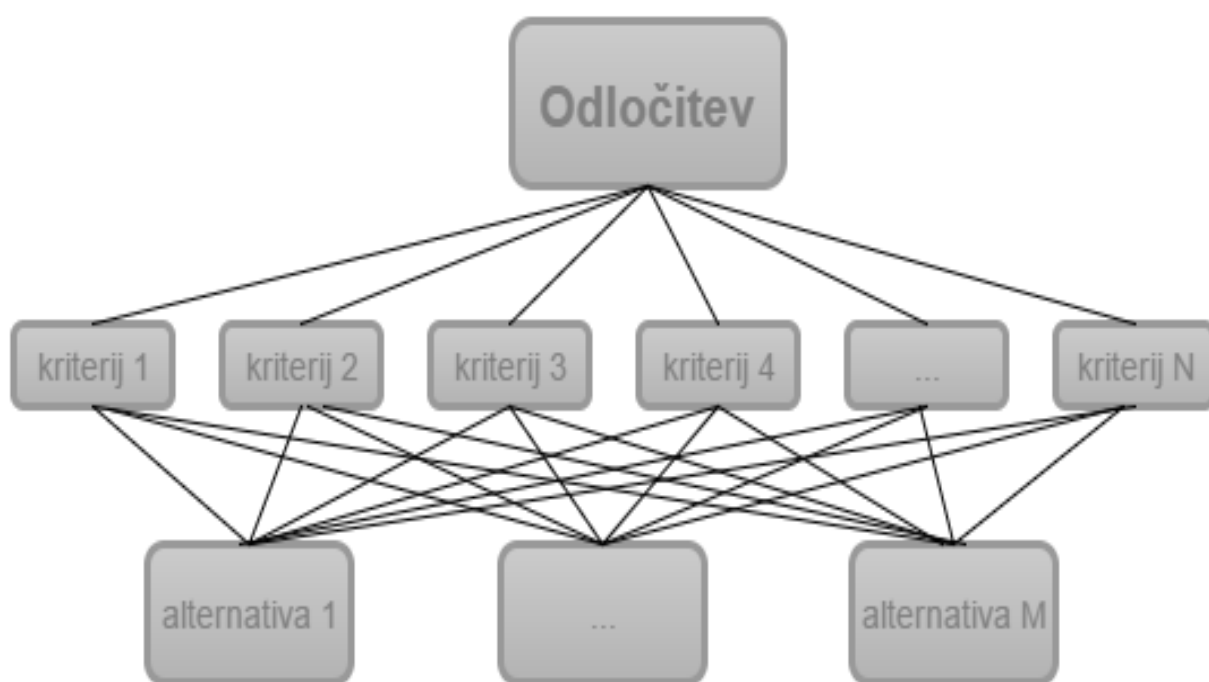
- opredelitev uteži elementom hierarhične strukture s pomočjo lastnih vrednosti,
- preverjanje skladnosti matrik, kot zagotavljanje konsistence presoje odločevalcev,
- združevanje relativnih uteži elementov za določanje skupne ocene posamezne alternative.

5.1.3.1 Korak 1 – opredelitev problema in cilja

Problem, ki ga odločevalec želi rešiti, mora najprej poglobljeno analizirati, ga razumeti in natančno opisati. Pri analizi problema je smotrno problem razdeliti v manjše in lažje rešljive enote.

Eden poglobitnih elementov je jasno zastavljen cilj, ki ga želi doseči s tem modelom. Za doseganje cilja opredelimo kriterije in alternative, ki vstopajo v odločitveni model.

Slika 9: Hierarhični prikaz modela



Vir: T. L. Saaty, *Fundamental of Decision Making and Priority Theory With the Analytic Hierarchy Process*, 1994, str.95.

5.1.3.2 Korak 2 – sestava hierarhične strukture

Pri AHP metodi celotni problem prikažemo s hierarhično strukturo kot prikazuje Slika 9. V korenu drevesa je glavni cilj, kateremu sledi več ravni kriterijev. Kriteriji so enote, ki predstavljajo razdeljen glavni problem. Kriterij na določeni ravni lahko vsebuje podrejene elemente, ki podrobneje opisujejo določen kriterij in jih imenujemo izpeljani kriteriji. Kriteriji, ki so na najnižji ravni kriterijev imenujemo osnovni in predstavljajo lastnosti alternativ. Pri AHP metodi so na najnižji ravni modela navedene alternative, ki se povežejo z vsakim osnovnim elementom predhodne ravni.

Hierarhična predstavitev problema ima naslednje prednosti (Saaty, 1994, str. 98–99):

- s hierarhično strukturo je opis, kako prioritete višjega elementa vplivajo na prioritete nižjih odvisnih elementov, naraven.
- Struktura nudi veliko informacij o strukturi in pomenu nižje ležečih elementov preko višje ležečih elementov. Omejitve višje ležečih elementov se aplicira na nižje ravni.
- Pri hierarhični strukturi gre za naraven prikaz celote.
- Tovrstne strukture so stabilne in prilagodljive. Stabilne, ker imajo majhne spremembe majhne vplive in prilagodljive, ker manjše dopolnitve ne motijo celotnega namena modela.

Pri sestavi hierarhične strukture poznamo dva pristopa (Saaty, 1994, str. 99–100): od spodaj–navzgor (angl. *bottom-up*) in od zgoraj–navzdol (angl. *top-down*). V prvem primeru najprej opredelimo in opišemo alternative ter izvedemo odločitve, šele nato aktivnosti izvedemo nad kriteriji od spodnje ravni proti glavnemu cilju. Ta pristop je uporaben, ko so alternative bolj definirane kot kriteriji. V drugem pristopu je postopek obraten, in sicer izvršujemo odločanje z začetkom pri glavnemu cilju in nato pridemo preko kriterijev do alternativ. Ta pristop se uporablja, ko so kriteriji dovolj dobro definirani ali pa je množica alternativ še neopredeljena.

5.1.3.3 Korak 3 – parna primerjava

Ena izmed prednosti AHP metode je parna primerjava elementov, tako alternativ, kot tudi kriterijev. S tem je omogočena tudi primerjavo sicer na prvi pogled neprimerljivih objektov. Paroma primerjamo vse elemente strukture, ki so povezani. To velja tako za alternative, kot tudi za kriterije.

Kriterije, ki so predmet primerjave, zapišemo v matriko oz. tabelo kot prikazuje spodnja preglednica Tabela 5.

Tabela 5: Parna primerjava elementov

	k_1	k_2	k_3
k_1	1	a_{12}	a_{13}
k_2	$1/a_{12}$	1	a_{23}
k_3	$1/a_{13}$	$1/a_{23}$	1

Pri **primerjavi kriterijev** k_i in k_j se sprašujemo naslednje: kolikokrat je k_i pomembnejši od k_j (Bohanec, 2006, str. 120) in odločitev zapišemo v rezultat a_{ij} . Pri vrednotenju te odločitve upoštevamo skalo kot jo prikazuje Tabela 6. Iz tega sledi, da je primerjava istega elementov vedno 1, tj. $a_{ii} = 1$.

Tabela 6: Temeljna skala za parno primerjanje elementov

Vrednost	Opis
1	Elementa sta enako pomembna
3	Zmerno večja pomembnost prvega elementa v primerjavi z drugim
5	Velika prednost prvega elementa
7	Zelo velika prednost prvega elementa; v praksi prvi dokazano prevladuje nad drugim
9	Skrajna prednost prvega elementa; prednost prvega je najvišja možna

Vir: T. L. Saaty, *Decision making with the analytic hierarchy process*, 2008, str. 86, tabela 1.

Kadar je drugi parameter pomembnejši od prvega, vrednost zapišemo obratno sorazmerno kot $1/3$, $1/5$, $1/7$ in $1/9$. V kolikor odločevalec ne more sprejeti odločitve s podano skalo, lahko uporabi vmesne (sode) celoštevilčne vrednosti: 2, 4, 6, 8. Vmesne vrednosti opisujejo kompromis med osnovnimi vrednostmi. Primer vrednosti 4 bi pomenil zmerno do veliko prednost prvega elementa v primerjavi z drugim. Dovoljeno je uporabljati tudi vmesne ne-celoštevilčne vrednosti (npr. 1,1–1,9), v primeru, da so vrednosti elementov zelo blizu ena drugi (Saaty, 2008, str. 86).

Pri **primerjavi alternativ** je postopek enak, le da parno primerjavo alternativ izvedemo za vsak kriterij ločeno. Pri tem se sprašujemo, kolikokrat je vrednost osnovne lastnosti alternative A1 bolj ugodna kot podana vrednost za alternativo A2. Primerjalne matrike alternativ služijo kot vhodni podatki v funkcijo odločanja oz. v izdelan odločitveni model.

5.1.3.4 Korak 4 – opredelitev uteži

Ko s prejšnjim korakom ovrednotimo vse parne primerjave med kriteriji in alternativami, za vsak osnovni kriterij posebej, vsaki matriki (tabeli) primerjav, izračunamo uteži.

Izračun uteži pri AHP metodi temelji na izračunu lastnih vrednosti matrike primerjav. Kot navaja Bohanec (2006, str. 121) gre za razmeroma zapleten postopek in je v ta namen priporočljivo uporabiti namensko programsko opremo. Prednost takšnega izračuna je v njegovi natančnosti ter omogoča ocenitev konsistentnosti matrik primerjav.

V izogib zapletenim izračunom metoda AHP dovoljuje nekatere poenostavitve. Saaty in Vargas (2012, str. 8) v ta namen predlagata dve alternativni za izračun lastnega vektorja:

- v kolikor je število elementov lastnega vektorja ≤ 3 , izračun normalizacije vrstice sovpada z izračunano lastno vrednostjo matrike,
- druga alternativa je le približek pravi vrednosti, vendar je bistveno preprostejša: sprva vse stolpce v primerjalni matriki normiramo tako, da je vsota stolpca enaka 1, nato po vrsticah izračunamo povprečne vrednosti za vsak element posebej.

Slednja alternativa je kljub približkom pogosto uporabljena različica izračuna. Pri tem pa Saaty in Vargas (1982) opozarjata, da uporaba alternativ kljub matematičnemu približku lahko vodi do preobrata v odločitvenem modelu in odsvetujeta njeno uporabo v pomembnih odločitvenih modelih. Torej je uporaba alternativ odvisna od vrste problema in pomembnosti odločitve.

5.1.3.5 Korak 5 – preverjanje skladnosti

Eden od načel AHP metode je logična skladnost modela (Saaty, 2001). Ocena pove, koliko so primerjave v matriki med seboj skladne. Saaty (2001) opisuje skladnost primerjalnih matrik kot:

- združevanje podobnih objektov in idej v skladu s homogenostjo in njihovo relevantnostjo. Primer: primerjava dveh oseb, moškega in ženske. Po kriteriju živih bitij alternativni sovpadata medtem, ko ju po drugi strani kriterij spol povsem loči.
- Druga vrsta skladnosti predstavlja tranzitivnost med kriteriji. Primer: če je element A 3-krat pomembnejši od elementa B in element B 3-krat pomembnejši od C, je izjava, da je A 9-krat pomembnejši od elementa C, edina skladna. To matematično zapišemo kot: $a_{ik} * a_{kj} = a_{ij}$, kjer a_{ij} predstavlja kriterij v i-ti vrstici in j-tem stolpcu.

Za preverjanje skladnosti primerjalnih matrik najprej izračunamo lastno vrednost matrike kot prikazuje enačba (3). Pri tem spremenljivka m predstavlja dimenzijo matrike A ter w elemente lastnega vektorja matrike A .

$$\lambda = \frac{1}{m} \sum_{i=1}^m \frac{(A * w) * i}{w_i} \quad (3)$$

V primeru popolne usklajenosti je največja lastna vrednost (λ) enaka dimenziji matrike A: $\lambda = m \Leftrightarrow A$, sicer pa je največja lastna vrednost večja od m ($\lambda > m$). Na osnovi tega odstopanja je zgrajen indeks usklajenosti kot prikazuje enačba (4):

$$I = \frac{\lambda - m}{m - 1} \quad (4)$$

V kolikor je $I=0$ ($\lambda = m$), imamo popolno usklajeno primerjalno matriko. Ta indeks primerjamo še s slučajnim indeksom I_R (angl. *random index*), ki ga dobimo iz slučajno generiranih pozitivnih recipročnih matrik enakih dimenzij nad določeno lestvico (Tabela 7). Količnik enačbe (5) opisuje usklajenost primerjalne matrike. Če je količnik manjši od 0,1, je matrika dovolj usklajena, v nasprotnem primeru je matriko potrebno popraviti, saj je matrika nekonsistentna in rezultati modela ne bodo pravilni.

$$\frac{I}{I_R} \quad (5)$$

Tabela 7: Slučajni indeksi v odvisnosti od dimenzije matrike

Dimenzija	I_R
1	0
2	0
3	0,52
4	0,89
5	1,11
6	1,25
7	1,35
8	1,40
9	1,45
10	1,49

Vir: A. Mrvar, Saatyev postopek, b.l., str. 7.

5.1.3.6 Korak 6 – določitev ocene alternative

Pri določanju uteži alternativam metoda omogoča dva načina določanja: idealni in distributivni način.

Idealni način se uporablja za izbor samo najboljše alternative ne glede na ostale. Pri tem načinu za vsak kriterij pri primerjavi alternativ uteži delimo z največjo vrednostjo med njimi, kar pomeni, da med vsemi dominira le tista z največjo vrednostjo – ta dobi končno vrednost 1.

Pogosteje v uporabi je **distributivni način**. Ta se uporablja pri primerjavi odvisnosti med alternativami. Vsota vseh lokalnih uteži kriterijev in alternativ je pri tem načinu enaka 1. Pri distributivnem izračunu končne vrednosti alternative uporabimo funkcije združevanja. Za funkcijo združevanja uporabimo uteženo vsoto in s tem dobimo končno vrednost za vsako alternativo posebej.

Od izbire načina je odvisno, ali odločevalec želi izbrati alternativo, ki je v primerjavi z drugimi boljša glede na vse kriterije, ali tisto, ki je najboljša med vsemi, ne nujno pri vseh kriterijih. Pri slednjem je treba upoštevati, da najboljša alternativa ni vedno dobra (Saaty, 2001).

5.1.3.7 Korak 7 – odločitev in analiza spremembe parametrov

Ko je model izdelan, alternative ovrednotene in čas za odločitev na voljo, je smotrno preveriti njegovo **občutljivost** na spremembe. Analizirati moramo, kako se spreminja rezultat posamezne alternative v odvisnosti spreminjanja podatka o alternativni, bodisi spremembe modela vrednotenja, npr. utež kakšnega kriterija (Bohanec, 2006, str. 95–96). Občutljivost modela je najlažje predstaviti z grafi, kjer s spremembo določene vrednosti od minimuma do maksimuma prikažemo nihanje rezultata. Na spletu je mogoče najti veliko programskih rešitev, ki z različnimi grafi odločevalcu prikažejo spremembo rezultata v odvisnosti od sprememb različnih spremenljivk.

Z analizo spremembe parametrov odločevalec preveri, kako stabilna je najboljša izračunana vrednost ter kako hitro se spreminja rezultat glede na spremembe vhodnih podatkov ali na modelu. Tipičen primer spremembe je nihanje pomembnosti posameznih kriterijev, ki so si po pomembnosti dovolj blizu.

5.2 Opredelitev kriterijev in strukture modela

V tem poglavju sprva opišem zasnovo novo izdelanega modela, sintezo z modeli iz literature in ugotovitvami na študiji primera. Nato sledi opis strukture novo nastalega modela, z opisi kriterijev in pripadajočimi enotami merjenja ter njihovimi medsebojnimi odvisnostmi. Opisi kriterijev so le kratki povzetki že zapisanega v predhodnih poglavjih.

5.2.1 Zasnova modela

Glavni cilj modela je pomagati majhnim in srednje velikim podjetjem pri odločitvi katerega ponudnika IaaS oblaka izbrati. Pri tem sem v model vnesel tudi komponente, ki opredelijo alternativo lastne izvedbe oblaka.

Model, ki sem ga izdelal je rezultat analize obstoječih modelov v literaturi, študije primera ter lastnih večletnih izkušenj na tem področju. Pri tem sem se osredotočil na potrebe majhnih in srednje velikih podjetij, ki za razliko od velikih navadno nimajo lastnega IT oddelka in se želijo razbremeniti skrbi za strojno opremo. Najem strojne opreme je lahko delni, kjer sami še vedno upravljajo s strojno opremo debelih odjemalcev (delovne postaje, lastni dodatni strežnik) ali v celoti, kjer podjetje za poslovanje uporablja le nosilce lahkih odjemalcev (pametne telefone, tablice in prenosnike).

Pri analizi obstoječih modelov in v študiji primera sem zaznal štiri bistvene pomanjkljivosti:

- prevelika delitev kriterijev v podrobnosti, kar poveča kompleksnost modelov, od uporabnika zahteva poglobljeno znanje oblaka ter poznavanje področja sklopa kriterijev (finance, IT, pravo ipd.);
- neupoštevanje medsebojne odvisnosti kriterijev in s tem izgubo posledic njihovega vpliva na odvisnost;
- neupoštevanje, kdo in kakšen položaj v podjetju ima odločevalec oz. oseba, ki vrednoti kriterije (direktor, vodja prodaje, analitik, arhitekt, razvijalec itd.);
- reševanju problema v danem času, brez atributov, ki kažejo vpliv odločitve na prihodnost.

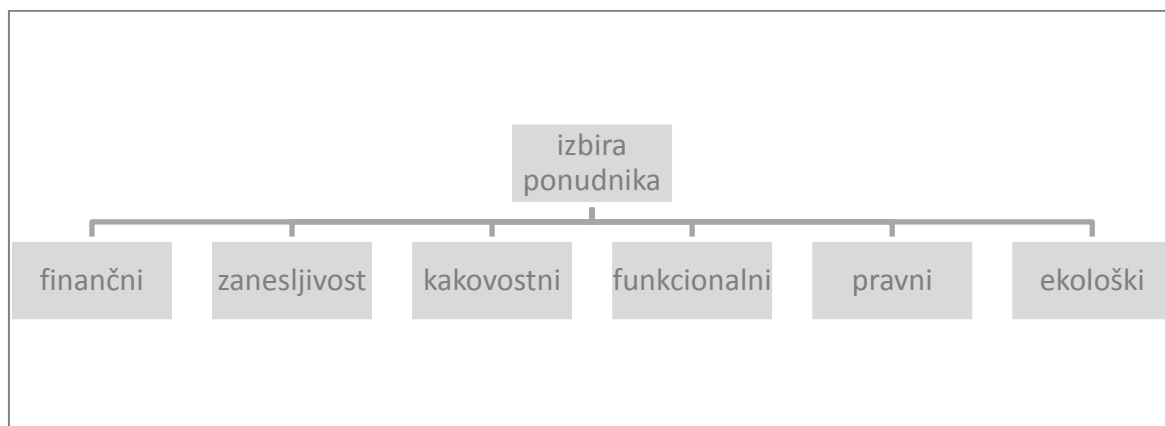
5.2.2 Struktura modela

Struktura modela temelji na hierarhični zasnovi, ki jo uporablja večparametrsko odločanje in AHP metoda. Nastali model je sinteza analizirane literature obstoječih modelov in ugotovitvam v študiju primera. Pri izdelavo modela sem vključil popravke pomanjkljivosti, ki sem jih identificiral med analizo.

Pomanjkljivost neupoštevanja medsebojne odvisnosti rešujem s hierarhično strukturo, katero uporablja AHP metoda. V strukturi opredelim odvisne kriterije in jih medsebojno povežem v logične skupine. S tem sem dobil novo nastali kriterij, ki je odvisen od svojih podrejenih. Pri tem ohranjam kompromis in z njim kompleksnost modela. Pomembnejše abstraktne kriterije sem razdelil le v tolikšni meri, da z odvisnimi kriteriji reprezentativno opišem nadrejeno lastnost. Vse manj pomembne podkriterije za odločitev sem izpustil.

Prva raven kriterijev predstavlja najvišjo abstrakcijo lastnosti alternativ kot prikazuje Slika 10. Strukturo sestavljajo kriteriji financ, funkcionalnosti, kakovosti, prava, ekologije in zanesljivosti.

Slika 10: Osnova struktura modela



Kriterija pravo in ekologija sta hkrati že lista te veje in predstavljata osnovna kriterija. Razlog, da kriterija ostajata na višji ravni abstrakcije, je v njuni (ne)pomembnosti. Ker model ni namenjen organizacijam (banke, zavarovalnice ipd.), ki imajo posebne zakonske uredbe. Podrobnosti **prava** nisem podrobneje opredelil. Ohranitev kriterija na višji abstraktni ravni je smotrna, saj za IT rešitve vseeno veljajo osnovna zakonska določila, kot npr. zakon o varstvu osebnih podatkov ali davčna zakonodaja.

Kriterij **ekologija** vključuje predvsem uporabo obnovljivih virov velikih podatkovnih centrov in racionalno uporabo energije. V primeru alternative z lastno infrastrukturo nima velikega pomena. Kljub temu, da kriterij nima neposrednega vpliva na poslovno odločitev, je po mojem mnenju in po mnenju intervjujancev prav, da je element vključen v model.

Kriterij prava in ekologije imata opisne vrednosti in jih odločevalec vrednoti s svojim subjektivnim mnenjem na podlagi svojega znanja in izkušenj.

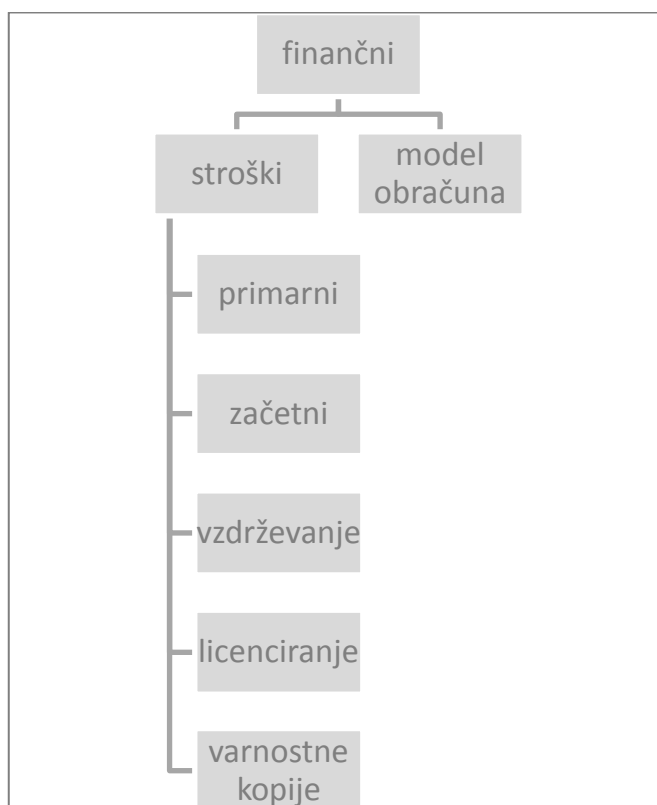
5.2.2.1 Finančni kriteriji

Finančni kriterij je sestavljen iz celotnih stroškov, ki jih ima podjetje s to storitvijo in modela obračuna. Model obračuna je osnovni kriterij in opisuje, na kakšen način ponudnik obračunava najemnino podjetju. Gre za kvalitativni opisni kriterij. Kriterij stroškov vključuje še dodatno raven odvisnih kriterijev:

- najemnina: najemnina za časovno obdobje meseca.
- Začetni stroški implementacije vključujejo šolanje uporabnikov, stroške migracije, stroške nakupa dodatne opreme ipd.
- Vzdrževanje je pomemben kriterij in je navadno vključen v najemnini, vendar to ni nujno. Njegova prisotnost prisili odločevalca v preverjanje, ali je vzdrževanje del najemnine ali predstavlja dodatne stroške.

- Poleg strojne opreme najemodajalec potrebuje določeno programsko opremo za svoje poslovanje. Večina ponudnikov nudi licence za operacijske sisteme, manj pa jih nudi dodatna uporabna orodja (podatkovne baze ipd.).
- Zagotavljanje varnostne kopije je nujno opravilo, ki ga mora ponudnik izvajati. Vendar naj ne bo samoumevno, da je strošek že vključen v osnovno najemnino.

Slika 11: Finančni kriteriji



Vsi finančni kriteriji so kvantitativne narave in jih lahko vrednotimo premo sorazmerno glede na višino stroška. Nemalokrat katerega od kriterijev poleg višine stroška spremlja še spremna beseda, katero je nujno potrebno vključiti v odločevalni proces. Tipičen primer so osnovni stroški, kjer dodaten opis navaja, ali so vzdrževanje, podpora in upravljanje varnostnih kopij vključeni v osnovni najem, ali ne. Strukturo finančnih kriterijev prikazuje Slika 11.

5.2.2.2 Kakovostni kriteriji

Kriteriji, ki opredeljujejo kakovost storitve, so tipično kvalitativne narave, kar pomeni, da jih opisujemo z besedo. Pri tem je prisotna velika subjektivnost odločevalca, saj temelji na njegovih izkušnjah in znanju. Kot prikazuje Slika 12, je struktura kriterija kakovosti razdeljena na tri ravni. Pri tem velja razmisliti, da sklop kriterijev modificiramo glede na to, kdo je v vlogi odločevalca. Za vodilni kader zadnja raven kriterijev morda ni smotrna,

celo nerazumljiva medtem, ko velja za inženirja v vlogi odločevalca nasprotno. V tem primeru se nekateri sestavljeni osnovni kriteriji transformirajo v osnovne.

Opisi kriterijev kakovosti se nahajajo v preglednici Tabela 8.

Slika 12: Kriteriji kakovosti

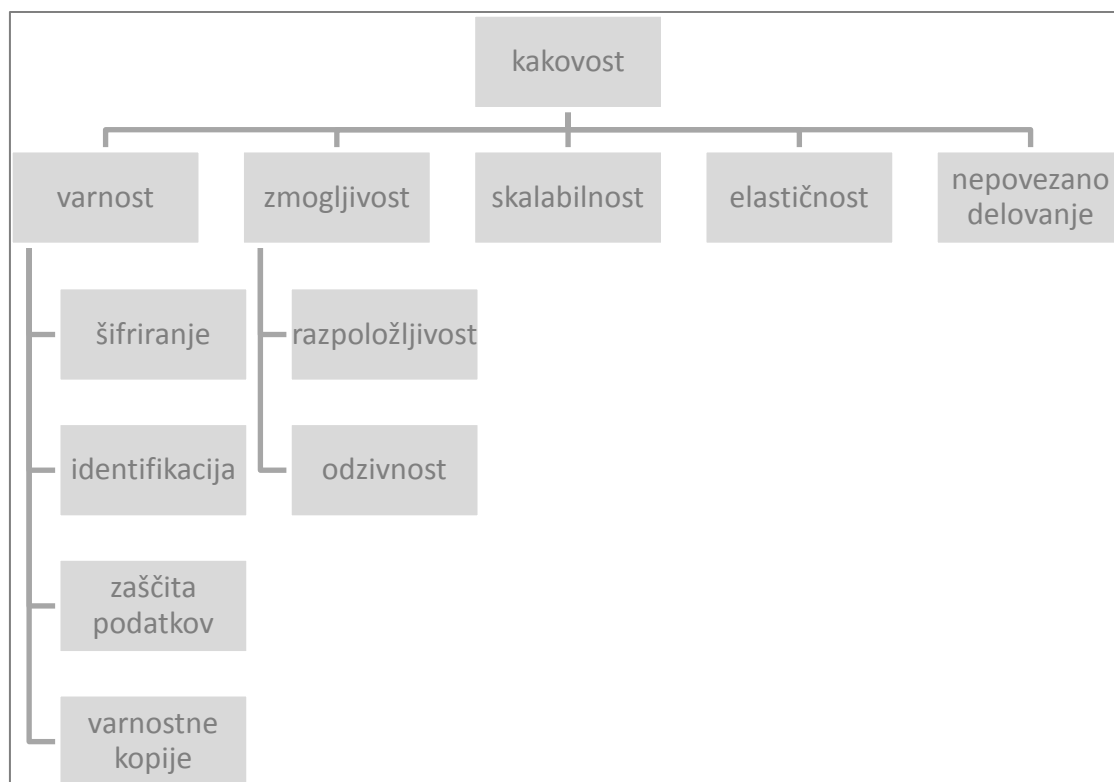


Tabela 8: Kriteriji kakovosti

Sestavljen kriterij	Osnovni kriterij	Opis
Elastičnost		Zmožnost prilaganja virov na potrebne količine zahtev in prometa. Vključeno je tudi upravljanje s prometom.
Skalabilnost		Zmožnost povečanja virov.
Nepovezan način		Delovanje v nepovezanem načinu v primeru izpada internetne povezave. Kriterij vključuje kasnejšo sinhronizacijo podatkov ob ponovni povezavi.

se nadaljuje

Tabela 8: Kriteriji kakovosti (nad.)

Varnost	Šifriranje podatkov	Zmožnost šifriranja podatkov v shrambi podatkov kot tudi za prenos podatkov.
	Identifikacija	Preverjanje pristnosti uporabnikov pri prijavi v sistem.
	Zaščita podatkov	Fizična in programska zaščita podatkov.
	Varnostne kopije	Izdelava varnostnih kopij podatkov – tudi lokacija kopija.
Zmogljivost	Razpoložljivost	Koliko časa je sistem na voljo v določenem časovnem obdobju. Ta kriterij ima standardno lestvico in je zato kvantitetne narave.
	Odzivnost	Čas potreben od podane zahteve uporabnika do prejetega odgovora. Odzivnost je merljiv kriterij, prenosorazmeren z izmerjenim časom.

5.2.2.3 Funkcionalni kriteriji

Podobno kot kvalitativni kriteriji so tudi kriteriji funkcionalnosti opisne narave. Kriteriji tega sklopa so navadno razumljivi vsem profilom zaposlenih in ni potrebe po modifikaciji modela. V Tabela 9 so navedene vse ravni funkcionalnih kriterijev z opisi, iz katerih je razvidna tudi njihova odvisnost.

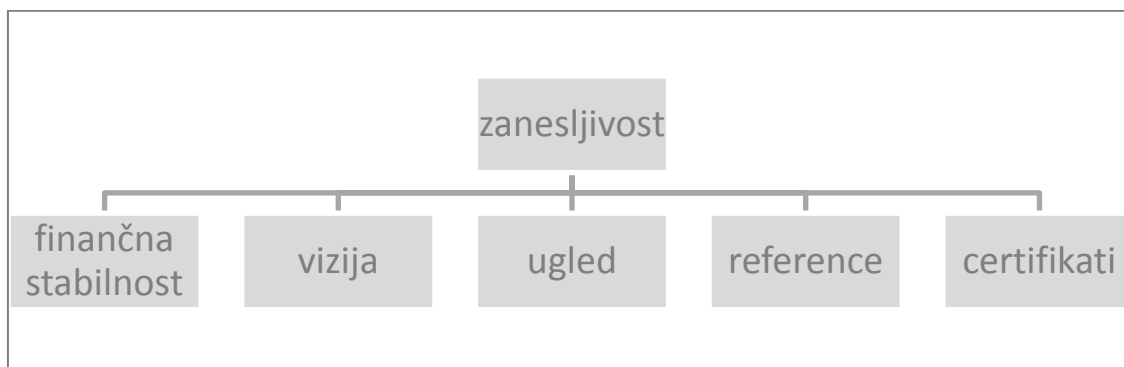
Tabela 9: Funkcionalni kriteriji

Sestavljen kriterij	Osnovni kriterij	Opis
Uporabnost	Namestljivost	Kompleksnost vzpostavitve storitve v podjetju.
	Razumljivost	Čas in napor potreben, da uporabniki razumejo in samostojno uporabijo dano storitev.
Standardizacija	Prenosljivost	Možnost zamenjave ponudnika.
	Integracija	Povezljivost z drugimi sistemi preko standardnih vmesnikov. Uporaba vtičev ponudnika (npr. za identifikacijo uporabnika, sinhronizacijo pri nepovezanem načinu delovanja ipd.).
Licenciranje		Ali ponudnik nudi najem licenc za programsko opremo v nasprotju z nakupom le te.
Vzdrževanje		V kakšnem obsegu je opredeljeno vzdrževanje.
Podpora		Kakšno vrsto podpore ponudnik nudi, tako ob vzpostavitvi sistema kot tudi med njenim delovanjem.
Prilagodljivost		Koliko je ponudnik prilagodljiv na več področjih: finance in obračun, SLA...

5.2.2.4 Kriteriji zanesljivosti

Strukturo kriterija zanesljivosti prikazuje Slika 13, kjer je razvidno, da je raven odvisnih kriterijev samo ena.

Slika 13: Kriteriji zanesljivosti



Uvedba tega sklopa kriterijev rešuje pomanjkljivost vpliva odločitve na prihodnost. V kriterijih zanesljivosti vrednotimo celotno stabilnost ponudnika in s tem določimo oceno tveganja v primeru prenehanja delovanja. V primeru implementacije z lastno infrastrukturo celotni sklop izpustimo, saj je tveganje na podjetju samemu.

Kljub temu, da so kriterije opisni, je večina med njimi na nek način merljiva oz. primerljiva. Finančno stabilnost (bilančno stanje) je mogoče preveriti na osnovi letnih poročil ponudnika ali v javnih finančnih registrih. Prav tako lahko preverimo certifikate, ki jih ponudnik poseduje. Njihovo verodostojnost je mogoče preveriti pri izdajatelju, prav tako morebitni potek certifikata ali kakšnega drugega dokazila. Za seznam referenc želimo, da je javno dostopen, saj je preverjanje zadovoljstva obstoječih strank uveljavljena praksa. Ostane še ugled ponudnika na trgu. Nazadnje lahko tudi ugled ponudnika na trgu preverimo posredno na osnovi uradno izdanih letnih poročil uveljavljenih raziskovalnih hiš (Gartner ipd.) in drugih finančnih institucij.

5.3 Opredelitev pomembnosti kriterijev in njihova uskladitev

Naslednji korak v izdelavi celotnega modela je opredelitev pomembnosti oz. uteži posameznim sestavljenim kriterijem. Določanje pomembnosti kriterija je povsem odvisno od vloge osebe v podjetju, pričakovanj, znanja in poznavanje problema ter izkušenj osebe, ki vrednoti pomembnosti. Pri določanju pomembnosti je ključnega pomena, da ima oseba, ki določa pomembnost kriterija na voljo vse informacije o kriteriju in da podrobno razume njegov pomen. Pomembnost, ki jo odločevalec določi je realna slika prioritet, ki jo želi odločevalec uveljaviti v procesu odločanja.

Določanje pomembnosti sestavljenega kriterija se določi podobno kot pri alternativah. Izdelamo tabelo, kjer stolpci in vrstice predstavljajo podrejene kriterije. Paroma primerjamo vsako vrstico s stolpcem in določimo utež glede na to, kolikokrat je kriterij v vrstici pomembnejši od tistega v stolpcu. Pri tem uporabimo AHP vrednostno lestvico od 1–9 oz. za obratne vrednosti od 1/9–1. Po diagonali tabele, kjer je primerjava kriterija samega s seboj, je vrednost vedno 1.

Pomembnosti kriterijev sem določil na osnovi analizirane literature in pomembnosti, ki so jih določili zaposleni v podjetju študije primera. Nepričakovano vse vrednosti niso konvergirale k istim kriterijem in njihova razvrstitev je bila različna. Največje odstopanje je med akademsko in poslovno sfero. V raziskovalni literaturi kot najpomembnejši kriterij ovrednotijo stroške, visoko uvrstijo še kakovost in funkcionalnost. Nenavadno nizko so ocenili varnost. V nasprotju s tem je v poslovni sferi varnost na prvem mestu. Poslovna sfera uvrsti kriterije kakovosti in zanesljivosti precej pred funkcionalnost in stroške (Repschläger et al., 2011, str. 167–170).

V študiji primera so prav tako odstopanja, vendar v veliko manjšem razponu. Razlog za odstopanje je predvsem zaradi različnih vlog zaposlenih, ki utežijo kriterije. Vodilni kader določi kot najpomembnejši kriterij stroške, katerim skoraj enakovredno sledita zanesljivost in kakovost. Inženirji pa v ospredje postavljajo zanesljivost in funkcionalnost ter tesno za njima še kakovost.

Za pridobitev končnega rezultata pomembnosti kriterijev izdelam primerjalno matriko, kjer vnesem vrednosti parnih primerjav kriterijev. Primerjam kriterij v vrstici s kriterijem v stolpcu in pri tem za pomembnost uporabim v predhodnih poglavjih opisano skalo. Po vnosu vrednosti izračunam usklajenost matrike oz. kriterijev in po potrebi matriko uredim – popravim pomembnosti kriterijev. To ponavljam, dokler matrika ni dovolj usklajena. Po uskladitvi matrike dobim rezultat, ki ga prikazuje Tabela 10. Iz tabele je razvidno, da je npr. finančni kriterij 3-krat pomembnejši od zanesljivosti oz. glede na skalo je finančni kriterij zmerno pomembnejši od zanesljivosti – v tabeli poudarjeno.

Tabela 10: Parna primerjava kriterijev prve ravni

	Finančni	Kakovostni	Zanesljivost	Funkcionalnost	Pravo	Ekologija
Finančni	1	3	3	3	9	9
Kakovostni	1/3	1	1	1/3	7	7
Zanesljivost	1/3	1	1	1/3	5	7
Funkcionalnost	1/3	3	3	1	5	7
Pravo	1/9	1/7	1/5	1/5	1	3
Ekologija	1/9	1/7	1/7	1/7	1/3	1

Primerjalno matriko normaliziram tako, da bo vsota posameznega stolpca enaka 1. Normalizirani matriki po vrsticah izračunam aritmetično sredino in dobljen rezultat predstavlja **prioritetni vektor** (Tabela 11). Prioritetni vektor opisuje pomembnost posameznega kriterija. Iz dobljenih rezultatov vidimo, da je vpliv finančnega kriterija na končni cilj, 40%, skoraj polovičen, temu sledi skoraj polovico manj vpliven kriterij funkcionalnost s 26%, kakovost s 14%, zanesljivost s 13% ter z zelo nizko vplivnostjo

kriterij prava 4% in ekologije s 3%. Glede na dobljene rezultate je finančni kriterij skoraj 2-krat pomembnejši kot funkcionalnost, kateri mu sledi.

Tabela 11: Normalizirana matrika in izračunan prioritetni vektor

	Finančni	Kakov.	Zanes.	Funkc.	Pravo	Eko.	Prioritetni vektor
Finančni	0,44	0,39	0,36	0,60	0,33	0,26	0,40
Kakovostni	0,09	0,13	0,12	0,07	0,26	0,21	0,14
Zanesljivost	0,15	0,04	0,12	0,07	0,18	0,21	0,13
Funkcionalnost	0,22	0,39	0,36	0,20	0,18	0,21	0,26
Pravo	0,05	0,02	0,02	0,04	0,04	0,09	0,04
Ekologija	0,05	0,02	0,02	0,03	0,01	0,03	0,03

Za dobljeno matriko določim še **faktor usklajenosti**, katerega dobim s pomočjo spletne aplikacije za delo z AHP metodo imenovana BPMSG AHP (BPMSG AHP Online System, 2016). Faktor usklajenosti zgornje primerjalne matrike je 6,9%. Ker je faktor usklajenosti manjši kot 10% je matrika dovolj usklajena in pomembnost kriterijev primerna za nadaljnje odločanje (Saaty & Vargas, 2006, str. 53).

Izračunan prioritetnega vektorja podrejenih za sestavljen kriterij predstavlja lokalno pomembnost podrejenih kriterijev. Ta pomembnost opisuje, koliko je določen kriterij pomemben (vpliv odločitve) glede na nadrejen kriterij. Da bi dobili globalni vpliv (podrejenega) kriterija na končno oz. celotno odločitev, je treba prioritetni vektor podrejenih kriterijev normalizirati na vrednost nadrejenega kriterija. Primer spodnje matrike (Tabela 12) prikazuje izračun finančnega kriterija, kjer so stroški 7-krat bolj pomembni od modela obračuna.

Tabela 12: Pomembnost kriterija FINANCE

FINANCE	Model obračuna	Stroški	Lokalni prioritetni vektor	Globalni prioritetni vektor
Model obračuna	1	1/7	0,12	0,05
Stroški	7	1	0,88	0,35

Po normalizaciji matrike dobimo prioritetni vektor **lokalnih pomembnosti**, kjer stroški predstavljajo 88% pomembnost, preostalih 12% pa model obračuna. Te vrednosti označujejo pomembnost glede na nadrejeni finančni kriterij. Da bi izračunali kakšen je **globalni vpliv** stroškov in modela obračuna na celotni odločitveni model, moramo prioritetni vektor normalizirati na pomembnost nadrejenega kriterija finance, t. j. da izračunamo produkt lokalnega vektorja s pomembnostjo nadrejenega vektorja (40%). Globalni vpliv stroškov je tako 35%, za preostalih 5% pa ima vpliv na celotno odločitev model obračuna.

Tabela 13: Model za izbiro ponudnika v oblaku

IZBIRA PONUDNIKA	Raven 1	Raven 2	Raven 3	Pomembnost v %
	PRAVO			4,0
	EKOLOGIJA			3,0
	FINANCE (40%)	model obračuna		5,0
		stroški (35%)	najemnina	11,0
			začetni	1,7
			vzdrževanje	8,5
			licenciranje	7,7
			varnostna kopija	6,1
	FUNKCIONALNOST (26%)	licenciranje		5,9
		vzdrževanje		1,5
		podpora		1,1
		prilagodljivost		6,0
		uporabnost (9,10%)	namestljivost	1,6
			razumljivost	7,5
		standardizacija (2,40%)	prenosljivost	2,1
			integracija	0,3
	KAKOVOST (14%)	elastičnost		3,9
		skalabilnost		0,9
		nepovezan način		0,6
		varnost (6,80%)	šifriranje podatkov	0,4
			identifikacija	2,1
			zaščita podatkov	2,2
			varnostna kopija	2,1
		zmogljivost (1,80%)	razpoložljivost	0,9
			odzivnost	0,9
	ZANESLJIVOST (13%)	finančna stabilnost		6,3
		vizija		2,0
		ugled		3,4
		referenca		0,8
		certifikati		0,5

Celotni odločitveni model za izbiro ponudnika v oblaku sestavljajo 3 ravni, 30 osnovnih kriterijev in 10 prioriternih vektorjev.

Osnovni kriteriji so kriteriji, katere vrednosti je treba opredeliti za vsako alternativo. Pri tem lahko model enostavno skrčimo tako, da zadnjo raven izpustimo. S tem sicer povečamo abstrakcijo modela, vendar je to v nekaterih primerih priporočljivo. V primeru, da je odločevalec vodilni kader, lahko zadnjo raven, ki podrobneje opisuje tehnično lastnost storitve oblaka izpustimo. S tem dobimo tudi manjše število primerjanj.

Postopek izdelave matrike, njena normalizacija, izračun prioritetnega vektorja ter izračun usklajenosti matrike ponovim za vse kriterije. Za sestavljene kriterije pa je potrebno izračunati še globalni vpliv njihovih podrejenih kriterijev. Celotna matrika modela je prikazana v Tabela 13.

V stolpcu Pomembnost so globalne pomembnosti posameznega osnovnega kriterija, predstavljene v odstotkih. Skupna pomembnost sestavljenega kriterija je napisana pod njegovim nazivom in opisuje seštevek vseh pomembnosti podrejenih kriterijev, npr. pomembnost za stroške je 35%.

5.4 Opredelitev alternativ modela

V procesu odločanju je vhodna množica alternativ enako pomembna kot določanje ostalih elementov modela. Pri vključevanju alternativ v večparametrski odločitveni model mora odločevalec paziti, da množica alternativ ne bo prevelika. Za uporabo AHP metode je priporočljivo število alternativ manj kot 10 (Bohanec, 2006, str. 122). V kolikor ima odločevalec na vходу večje število alternativ, je sprva smotrno uporabiti katero od enostavnih odločitvenih metodologij, ki temeljijo na intuitivnem odločanju.

Najenostavnejša je praktični test (angl. *practical test*), kjer odločevalec po svoji intuiciji medsebojno primerja alternative in jih na podlagi lastnosti izloči oz. obdrži (Li, Yang, Kandula, & Zhang, 2010; Sun et al., 2014, str. 143). Še en primer preproste metode je analiza prednosti in slabosti alternativ (Baker et al., 2001, str. 9). Pri tej metodi zberemo najboljše in najslabše lastnosti alternativ in na podlagi primerjav izberemo tisto alternativo, ki ima največ najboljših in najmanj najslabših lastnosti (Bohanec, 2006, str. 84). Neodvisno od izbrane metode je cilj, da se število alternativ omeji.

Ko odločevalec opredeli še končno množico alternativ, uporabi izdelan model za vrednotenje in analizo alternativ (Bohanec, 2006, str. 114). Alternative se nahajajo na najnižji ravni hierarhičnega modela. Vsaka alternativa je povezana z vsemi osnovnimi kriteriji, t. j. z listi odločitvenega drevesa. Za vsak osnovni kriterij izvedemo parno primerjavo alternativ, za vse pare alternativ. Primerjava alternativ za kriterij, ki ga lahko opredelimo kvalitativno je enostavna, saj je rezultat parne primerjave lahko le velikost vrednosti. Tipičen primer takšnega kriterija so stroški. Najemnino kot primarni strošek določimo neposredno iz cenika ponudnika ali po določeni formuli ter jo vnesemo v model za določeno alternativo. Pri tem mora biti odločevalec pozoren, ali je primerjava stroška verodostojna, ali za določeno ceno dobimo za dve alternativni enake storitve, ali je poleg stroška najema potrebno dodati spremno besedo (npr. stroški najemna že vključujejo varnostno kopijo). Za kvalitativne kriterije je odločitev subjektivne narave in temelji na strokovnem znanju, poznavanju področja in izkušnjah odločevalca oz. uporabnika (Sun et al., 2014, str. 148).

Postopek primerjave, normalizacije in izračuna prioritetnega vektorja alternativ je enak kot pri kriterijih (Bohanec, 2006, str. 121). Primerjalna matrika alternativ po ugledu ponudnika na trgu je prikazana v preglednici Tabela 14. Vrednosti se nanašajo na parno primerjavo alternative v vrstici z alternativo v stolpcu. Primer primerjave alternative »lastna« z AWS (v tabeli poudarjena vrednost) daje vrednost 1/9. To pomeni, da je AWS skrajno boljša od alternative »lastna«. Za vsak osnovni kriterij (listi strukture) je potrebno izdelati matriko in alternativam določiti pomembnosti.

Za končno preverjanje svojega modela sem izbral po enega neimenovanega domačega (stolpec »SI«) in nemškega (stolpec »D«) ponudnika ter dva vodilna ponudnika na globalnem trgu (AWS in Microsoft – v tabeli zapisano kot MS). Za potrebe študije primera sem kot eno izmed alternativ dodal izvedbo privatnega oblaka v domeni podjetja s pomočjo partnerjev. Ta alternativa je izvedba oblaka z lastno infrastrukturo in sem jo označil kot »lastna«.

Tabela 14: Parna primerjava alternativ za kriterij "Ugled na trgu"

Ugled	SI	D	AWS	MS	lastna
SI	1	1/3	1/9	1/9	5
D	3	1	1/9	1/9	7
AWS	9	9	1	3	9
MS	9	9	1/3	1	9
lastna	1/5	7	1/9	1/9	1

Način primerjave za kriterij ugled je distributiven. Ko odločevalec izdela primerjalno matriko, nadaljuje po enakem postopku kot pri kriterijih, tj. z normalizacijo matrike in izračunom prioritetnega vektorja. Lokalno dobljeni vektor predstavlja, kolikšen odstotek pomembnosti za ugled ponudnika doprinese posamezna vrednost alternative. Izračunam še globalni prioritetni vektor, tako da normaliziram lokalni vektor glede na vrednost globalne pomembnosti kriterija ugled (3,4%). Za podano matriko dobimo prioriteta vektorja kot prikazuje Tabela 15.

Tabela 15: Prioritetna vektorja za alternative po kriteriju Ugled

Ugled	Lokalni prioritetni vektor	Globalni prioritetni vektor
SI	0,06	0,0021
D	0,10	0,0033
AWS	0,47	0,0159
MS	0,29	0,0100
lastna	0,08	0,0027
skupaj	1,00	0,0340

Parne primerjave, normalizacije in izračun prioriternih vektorjev ponovim za vse kriterije. Kot rezultat dobim tabelo globalnih prioriternih vektorjev za vse kriterije vseh alternativ. Za funkcijo združevanja uporabim uravnoteženo vsoto. Ker globalni prioritetni vektor že predstavlja globalno uravnoteženost, njegove vrednosti za posamezno alternativo le seštejem (enačba (6): n je št. kriterijev; $g_i(a)$ je vrednost globalnega vektorja za kriterij i za alternativo a). Dobljene vsote razvrstijo alternative po primernosti odločitve od največje proti najmanjši.

$$f(lastna) = \sum_{i=1}^n g_i(lastna) \quad (6)$$

5.5 Sprejetje odločitve

Ko z izdelanim modelom ovrednotimo alternative je rezultat urejen vektor alternativ z njihovimi vrednostmi. Med njimi lahko označimo tisto z najvišjo vrednostjo in proces odločanja je končan. Pri tem pa je treba biti previden, saj izbira najboljše alternative ni nujno dobra (Saaty v Podobnik, 2015, str. 43). Ključno je, da je odločevalec pozoren na velikost razlik rezultatov med alternativami. Smotrno je izvesti analizo sprememb modela in vhodnih sprememb in pri tem opazovati posledice, ki se odrazijo na rezultatu. V ta namen predlagam uporabo programskih rešitev za delo z AHP metodo. Tovrstne rešitve nam z grafi prikažejo odvisnosti sprememb modela in vhodnih podatkov ter njihov vpliv na rezultat. Na takšen način lahko odločevalec utrdi in verificira svojo odločitev.

Za potrebe svojega dela končne razvrstitve alternativ ne bom navajal, prav tako ne izbire alternative za študijo primera, saj je izbira določene alternative v domeni odločitvenega problema, ki ga odločevalec rešuje.

SKLEP

Razvoj oblaka ima zelo velik vpliv na IT področje v širokem pomenu besede in s tem predstavlja pomemben faktor v ekonomski in poslovni sferi (Repschläger et al., 2011, str. 163). Ti vplivi niso le tehnološki, temveč predvsem vplivajo tudi na strategije poslovanja in procese delovanja podjetij. V zadnjih letih, ko je razvoj oblaka dozorel, je to postal način, kako podjetja in posamezniki upravljajo z informacijami ter obdelujejo podatke (Dustdar, 2016, str. 13). Tako jim je omogočena široka izbira tehnologije, ki jo potrebujejo za svoje poslovanje. Z najemom storitev si zmanjšajo (začetne) stroške in skrb za obstransko, a nujno potrebno arhitekturo za svoje poslovanje. Čeprav ni pravilo, se v splošnem to toliko bolj odraža v *startup-ih* ter pri majhnih in srednje velikih podjetjih. Pri prvih so primarnega pomena nizki začetni stroški ter elastičnost in skalabilnost glede na

nepredvidljivo rast *startup-a*. V podjetjih, ki navadno nimajo lastnega IT oddelka, kamor spadajo majhna in srednje velika podjetja, je najem arhitekture in s tem prenos skrbi za vzdrževanje in podporo na stran ponudnika oblaka ugodna rešitev.

Čeprav oblak velja za pojav, ki v IT in poslovnem svetu prinaša kopico ugodnosti in prednosti, Misra in Mondal (2011, str. 504) opozarjata, da naj podjetja vzpostavijo objektivni pogled in podrobno preučijo realne vrednosti vključevanja tehnologije oblaka v svoje poslovanje. Pri tem se Liu et al. (2016, str. 1) sprašujejo, ali so oblaki res primerni za vsako podjetje in hkrati navajajo, da po podatkih ENISA 68,1% podjetij meni, da se z uporabo oblaka lahko izognejo investicijskim izdatkom za strojno opremo, IT podporo ipd. (ENISA, 2009).

Ponudba storitev v oblaku predstavlja kompleksno, heterogeno in distribuirano strukturo (Linthicum, 2016, str. 8), kar uporabniku onemogoča pregled in iskanje ustrezne storitve. Ker na tem področju ni standardizacije kriterijev, ki bi lahko omogočala enostavno primerjavo storitev, si podjetja ali posamezniki pomagajo z različnimi odločitvenimi modeli.

Omenjeno dejstvo je bilo povod za izbiro teme magistrskega dela. Kot svetovalec sem bil hkrati vpleten v praktični primer izbire načina implementacije arhitekture in ponudnika oblaka, kar je bila dodatna motivacija za izbrano temo. Namen dela je bilo izdelati model, ki podjetjem pomaga izbrati način implementacije oblaka in predvsem izbiro najustrežnejšega ponudnika oblaka. Pri tem sem se osredotočal na enostavnost in uporabnost modela za širšo uporabo. Za dosego namena sem s pomočjo literature analiziral opredelitev oblaka, opredelitve komponent in storitev, ki spremljajo oblak ter analiziral že obstoječe odločitvene modele, ki so bili razviti za tovrstne namene. Rezultat analize so bili opredeljeni kriteriji modela. Temu je sledila še analiza trga ponudnikov oblaka s poudarkom na IaaS. Dobljene okvirje iz literature sem nato uporabil v študiji primera, kjer sem poiskal sintezo z dobljenimi rezultati in na podlagi ugotovitev izdelal svoj odločitveni model. Pri tem sem skušal odpraviti vse prepoznane pomanjkljivosti obstoječih modelov, katere sem ugotovil, ali s pomočjo literature ali na praktičnem primeru.

Za izdelavo odločitvenega modela sem uporabil večparametrsko metodologijo AHP. Menim, da je odločitev za izbiro te metodologije prava. Uporabljena metodologija ni samo zelo pogosto uporabljena metodologija v večparametrskem odločanju na splošno (Lee et al., 2012; Whaiduzzaman et al., 2014, str. 6), temveč je tudi najpogosteje uporabljena v analiziranih modelih iz literature. S pomočjo te metodologije z uporabo tabel uporabniku pregledno in razumljivo predstavimo problem in odločitveni model, ki ga nato uporabi. Metoda s pomočjo programske opreme, ki jo lahko najdemo na spletu, omogoča kvalitetno in razumljivo analizo dobljenega rezultata s pomočjo grafov in morebitne spremembe glede na modifikacije vhodnih podatkov ali samega modela.

Menim, da je namen dosežen, saj je bil izdelani model že uporabljen na praktičnem primeru in pri tem pozitivno sprejet. Struktura modela omogoča prilagajanje ravni znanja odločevalca in pri tem ohranja svoj namen. Prav tako je model mogoče modificirati v kolikor ima zastavljen problem svoje specifične potrebe. Pri slednjem je sicer potrebno izvesti ponovno računanje pomembnosti elementov modela, kar pa s pomočjo programske opreme ne predstavlja nikakršnih omejitev..

Kljub doseženemu cilju je možnosti za nadaljnje raziskave še veliko. Neodvisno od same zasnove modela je, kot posledica heterogene ponudbe storitev oblaka, večji poudarek na samih kriterijih. Ob vzpostavitvi določenih standardov, bi bila opredelitev vrednosti kriterijev za določeno alternativo in njihova primerjava bolj verodostojna. To je možno doseči z uradno sprejetimi standardi in certifikati, ki dokazujejo določeno lastnost. S tem izločimo subjektivno komponento odločevalca pri mehkih kriterijih in uporabnikom omogočimo iskanje storitve na trgu oblaka s homogeno primerjavo in primernejšo izbiro storitve glede na njegove potrebe.

LITERATURA IN VIRI

1. Ahson, S.A., & Ilyas, M. (2011). *Cloud Computing and Software Services: Theory and Techniques*. New York: CRC Press, Taylor & Francis Group.
2. *Amazon Web Services (AWS) - Cloud Computing Services*. Najdeno 28. marca 2016 na spletnem naslovu <https://aws.amazon.com/>
3. Antonopoulos, N., & Gillam, L. (2010). *Cloud Computing: Principles, Systems and Applications*. London: Springer.
4. Armbrust, M., Fox, A., Griffith, R., Joseph, A. D., Katz, R. H., Konwinski, A., Lee, G., Patterson, D. A., Rabkin, A., Stoica, I., & Zaharia, M. (2009, 10. februar). *Above the Clouds: A Berkeley View of Cloud Computing*. University of California at Berkeley. Najdeno 14. avgusta 2015 na spletnem naslovu <http://www.eecs.berkeley.edu/Pubs/TechRpts/2009/EECS-2009-28.pdf>
5. Babcock, C. (2010). *Management strategies for the cloud revolution : how cloud computing is transforming business and why you can't afford to be left behind*. New York: McGraw-Hill, cop. 2010
6. Badger, L., Grance, T., Patt-Corner, R., & Voas, J. (2012, maj). Cloud Computing Synopsis and Recommendations. *NIST - National Institute of Standards and Technology: Information Technology Laboratory*. Najdeno 4. aprila 2015 na spletnem naslovu <http://nvlpubs.nist.gov/nistpubs/Legacy/SP/nistspecialpublication800-146.pdf>
7. Baker, D., Bridges, D., Johnson, G., Krupa, J., Murphy, J., & Sorenson, K. (2001, december). *Guidebook to decision-making methods*. Najdeno 10. aprila 2016 na spletnem naslovu https://www.researchgate.net/publication/255621095_Guidebook_to_Decision-Making_Methods
8. Baranwal, G., & Vidyarthi, D. P. (2014). A framework for selection of best cloud service provider using ranked voting method. *Advance Computing Conference (IACC), 2014 IEEE International* (str. 831–837). Gurgaon: IEEE.
9. Bengtsson, P. (1999, 9. april). *Multiple Case Studies - not just more data points?!* Najdeno 20. marca 2016 na spletnem naslovu <http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.30.9769&rep=rep1&type=pdf>
10. Bennett, A., & Elman, C. (2010). Case Study Methods. *The Oxford Handbook of International Relations*. Oxford: Oxford University Press.
11. Bilderbeek, P. (2014, 29. oktober). *Top 1000 European Public Cloud Computing Providers*. Najdeno 27. marca 2016 na spletnem naslovu <http://www.themetisfiles.com/2014/10/top-1000-european-public-cloud-computing-providers/>
12. Binning D. (2009, 24. april). Top Five Cloud Computing Security Issues. *Computer Weekly*. Najdeno 19. decembra 2015 na spletnem naslovu <http://www.computerweekly.com/Articles/2010/01/12/235782/Topfive-cloud-computing-security-issues.htm>

13. Bohanec, M. (2006). *Odločanje in modeli*. Ljubljana: DMFA-ZALOŽNIŠTVO.
14. *BPMSG AHP Online System*. Najdeno 11. aprila 2016 na spletnem naslovu <http://bpmsg.com/academic/>
15. Bruce, R. (2014, 15.julij). 5 Benefits of Hardware-as-a-Service (HaaS). *Meridian*. Najdeno 19. februarja 2016 na spletnem naslovu <http://www.whymeridian.com/blog/bid/390744/5-Benefits-of-Hardware-as-a-Service-HaaS>
16. BT launches high speed ISDN access to the internet. (1996). *Computer Law & Security Review*, 12(4), 257.
17. Burrington, I. (2016, 8. januar). Why Amazon's Data Centers Are Hidden in Spy Country. *The Atlantic*. Najdeno 28. Marca 2016 na spletnem naslovu <http://www.theatlantic.com/technology/archive/2016/01/amazon-web-services-data-center/423147/>
18. Chang, C. W., Liu, P., & Wu, J. J. (2012). Probability-Based Cloud Storage Providers Selection Algorithms with Maximum Availability. *2012 41st International Conference on Parallel Processing (ICPP)* (str. 199–208). Pittsburgh, PA: IEEE.
19. Chang, V., & Ramachandran, M. (2016). Towards Achieving Data Security with the Cloud Computing Adoption Framework. *IEEE Transactions on Services Computing*, 9(1), 138–151.
20. Chang, V., & Wills, G. (2016). A model to compare cloud and non-cloud storage of Big Data. *Future Generation Computer Systems*, 57, 56–76.
21. Chellappa, R.K., & Gupta, A. (2002). Managing computing resources in active intranets. *International Journal of Network Management*, 12(2), 117-128.
22. Chen, C., Yan, S., Zhao, G., Lee, B. S., & Singhal, S. (2012). A Systematic Framework Enabling Automatic Conflict Detection and Explanation in Cloud Service Selection for Enterprises. *2012 IEEE 5th International Conference on Cloud Computing (CLOUD)* (str. 883–890). Honolulu: IEEE.
23. Chorafas, D.N. (2011). *Cloud Computing Strategies*. New York: CRC Press, Taylor & Francis Group.
24. Chowdhury, N.M.M.K., & Boutabab, R. (2010). A survey of network virtualization. *Computer Networks*, 54(5), 862-876.
25. *Cisco – Desktops as a Service(Cloud-Based)*. Najdeno 19.februarja 2016 na spletnem naslovu <http://www.cisco.com/c/en/us/solutions/data-center-virtualization/desktop-as-a-service-solution/index.html>
26. *Citrix – Desktops as a Service*. Najdeno 19.februarja 2016 na spletnem naslovu <https://www.citrix.com/products/daas/overview.html>
27. Clark, J. (2012, 7. junij). How Amazon exposed its guts: The History of AWS's EC2. *ZDNet*. Najdeno 20. februarja 2016 na spletnem naslovu <http://www.zdnet.com/article/how-amazon-exposed-its-guts-the-history-of-awss-ec2/>
28. *ComputerWorldUK* (2014, 5. avgust). *How to pay the right price for your Oracle database*. Najdeno 22. marca 2016 na spletnem naslovu

- <http://www.computerworlduk.com/blogs/management-briefing/how-to-pay-the-right-price-for-your-oracle-database-3570808/>
29. CSMIC (2014, julij). *SMI*. Najdeno 3. marca 2016 na spletnem naslovu http://csmic.org/downloads/SMI_Overview_TwoPointOne.pdf
 30. Čehovin, G. (2011, 20. februar). Računalništvo v oblaku: fleksibilnejši dostop do računalniških storitev. *Zavod mladi podjetnik*. Najdeno 1. junija 2015 na spletnem naslovu <http://mladipodjetnik.si/podjetniski-koticek/poslovanje/racunalnistvo-v-oblaku-fleksibilnejši-dostop-do-racunalniskih-storitev>
 31. Daneasa, A. (b.l.). *Disadvantages of using Oracle Database 10g XE (Express ed.)*. Najdeno 22. marca na spletnem naslovu <http://www.webxpert.ro/andrei/2009/03/17/disadvantages-of-using-oracle-database-10g-xe-express-edition/>
 32. Darrow, B. (2012, 17. september). *5 things you need to know about cloud in Europe*. Najdeno 27. marca 2016 na spletnem naslovu <https://gigaom.com/2012/09/17/5-things-you-need-to-know-about-cloud-in-europe/>
 33. Dastjerdi, A. V., Tabatabaei, S. G. H., & Buyya, R. (2010). An Effective Architecture for Automated Appliance Management System Applying Ontology-Based Cloud Discovery. *2010 10th IEEE/ACM International Conference on Cluster, Cloud and Grid Computing (CCGrid)* (str. 104–112). Melbourne: IEEE.
 34. Dustdar, S. (2016). Cloud Computing. *IEEE Journals & Magazines*, 49(2), 12-13.
 35. Eisenhardt, K.M. (1989). Building Theories from case Study Research. *Academy of Management Review*, 14(4), 532-550.
 36. Elgazzar, K., Hassanein, H.S., & Martin, P. (2014). DaaS: Cloud-based mobile Web service discovery. *Pervasive and Mobile Computing*, 13, 67-84.
 37. ENISA (2009, november). *An SME perspective on cloud computing*. Najdeno 22. marca 2016 na spletnem naslovu https://www.enisa.europa.eu/activities/risk-management/files/deliverables/cloud-computing-sme-survey/at_download/fullReport
 38. Eurocloud - Spletna stran Zavoda E-Oblak. Najdeno 28. marca 2016 na spletnem naslovu <http://eurocloud.si/>
 39. Filipe, J. (2008). *E-business and Telecommunications*. Berlin: Springer.
 40. Foo, T. (2011, 4. november). Cloud Computing for Financial Institutions: With the Benefits Come Security Concerns. *MBAF*. Najdeno 16. januarja 2015 na spletnem naslovu <http://www.mbafter.com/newsletters/1841/Cloud-Computing-for-Financial-Institutions--With-the-Benefit.aspx>
 41. Fujitsu (2010). *Three questions for the man who coined the term "cloud computing"*. Najdeno 1. septembra 2014 na spletnem naslovu <http://www.bus.emory.edu/ram/Papers/FujitsuITCOMPASS.pdf>
 42. Garg, S. K., Versteeg, S., & Buyya, R. (2011). SMICloud: A Framework for Comparing and Ranking Cloud Services. *2011 Fourth IEEE International Conference on Utility and Cloud Computing (UCC)* (str. 210–218). Victoria, NSW: IEEE.
 43. Garg, S. K., Versteeg, S., & Buyya, R. (2013). A framework for ranking of cloud computing services. *Future Generation Computer Systems*, 29(4), 1012-1023.

44. Garrison, G., Wakefield, R.L., & Kim, S.(2015). The effects of IT capabilities and delivery model on cloud computing success and firm performance for cloud supported processes and operations. *International Journal of Information Management*, 35(4), 377-393.
45. Gartner (2009, 29. junij). *Highlights Five Attributes of Cloud Computing*. Najdeno 1. septembra 2015 na spletnem naslovu <http://www.gartner.com/newsroom/id/1035013>
46. Gartner (2015, 27. avgust). *Gartner Says Modernization and Digital Transformation Projects Are Behind Growth in Enterprise Application Software Market*. Najdeno 21. marca 2016 na spletnem naslovu <http://www.gartner.com/newsroom/id/3119717>
47. Ghezzi, C., Panzica La Manna, V., Motta, A., & Tamburrelli, G. (2015). Performance-driven dynamic service selection. *Concurrency and Computation: Practice and Experience*, 27(3), 633–650.
48. Gill, R. (2011). Why cloud computing matters to finance. *Strategic Finance Articles*, 92(7), 43–47.
49. Godse, M., & Mulik, S. (2009). An Approach for Selecting Software-as-a-Service (SaaS) Product. *IEEE International Conference on Cloud Computing, 2009. CLOUD '09* (str. 155–158). Bangalore: IEEE.
50. Gupta, J. N. D., Forgionne, G. A., & Mora, T. (2006). *Intelligent Decision-making Support Systems Foundations, Applications and Challenges*. London: Springer.
51. Harker, P. (1988). *The Art and Science of Decision-making: The Analytic Hierarchy Process*. (Working Paper 88-06-03). *Decision Science Department, The Wharton School, University of Pennsylvania, Philadelphia*. Berlin: Springer.
52. Hastings, R. (2014). *Making the most of the cloud:how to choose and implement the best services for your library*. Lanham: Scarecrow Press.
53. Hatamura, Y. (2006). *Decision-Making in Engineering Design Theory and Practice*. London: Springer.
54. He, Q., Han, J., Yang, Y., Grundy, J., & Jin, H. (2012). QoS-Driven Service Selection for Multi-tenant SaaS. *2012 IEEE 5th International Conference on Cloud Computing (CLOUD)* (str. 566–573). Honolulu: IEEE.
55. Herbst, N. R., Kounev, S., & Reussner, R. (2013). Elasticity in Cloud Computing: What It Is, and What It Is Not. *10th International Conference on Autonomic Computing - ICAC'13* (23 - 27). San Jose, CA: USENIX Association.
56. Höllwarth, T. (2012). *POT v oblak*. Ljubljana: Pasadena.
57. Hsiao, J. (2008, april). Amazon.com CEO Jeff Bezos on Animoto. *Animoto blog*. Najdeno 10. oktobra 2015 na spletnem naslovu <http://animoto.com/blog/company/amazon-com-ceo-jeff-bezos-on-animoto/>
58. Hulme, G. V. (2011). NIST formalizes cloud computing definition, issues security and privacy guidance. *CSO - Security and risk*. Najdeno 10. novembra 2014 na spletnem naslovu <http://www.csoononline.com/article/661620/nist-formalizes-cloud-computing-definition-issues-security-and-privacy-guidance>

59. IDC (2014, 3. november). *IDC Forecasts Public IT Cloud Services Spending*. Najdeno 27. februarja 2016 na spletnem naslovu <http://www.idc.com/getdoc.jsp?containerId=prUS25219014>
60. IS Decisions (b.l.). *FISMA/NIST 800-53/FIPS PUB 200 compliance*. Najdeno 14. januarja 2016 na spletnem naslovu <http://www.isdecisions.com/compliance/FISMMA-NIST-800-53-FIPS-PUB-200-compliance.htm>
61. Islam, S., Lee, K., Fekete, A., & Liu, A. (2012). How a Consumer Can Measure Elasticity for Cloud Platforms. *Proceedings of the 3rd ACM/SPEC International Conference on Performance Engineering* (str. 85–96). New York: ACM.
62. Jansen, W., & Grance, T. (2011, januar). Guidelines on Security and Privacy in Public Cloud Computing. *NIST - National Institute of Standards and Technology: Information Technology Laboratory*. Najdeno 4. aprila 2015 na spletnem naslovu <http://nvlpubs.nist.gov/nistpubs/Legacy/SP/nistspecialpublication800-144.pdf>
63. Jula, A., Sundararajanb, E., & Othmana, Z. (2014). Cloud computing service composition: A systematic literature review. *Expert Systems with Applications*, 41 (8), 3809–3824.
64. Jung, G., Mukherjee, T., Kunde, S., Kim, H., Sharma, N., & Goetz, F. (2013). CloudAdvisor: A Recommendation-as-a-Service Platform for Cloud Configuration and Pricing. *2013 IEEE Ninth World Congress on Services (SERVICES)* (str. 456–463). Santa Clara, CA: IEEE.
65. Karim, R., Ding, C., & Miri, A. (2013). An End-to-End QoS Mapping Approach for Cloud Service Selection. *2013 IEEE Ninth World Congress on Services (SERVICES)* (str. 341–348). Santa Clara, CA: IEEE.
66. Kearney, K. T., & Torelli, F. (2011). The SLA Model. V P. Wieder, J. M. Butler, W. Theilmann, & R. Yahyapour (ur.), *Service Level Agreements for Cloud Computing* (str. 43–67). New York: Springer.
67. Kittlaus, H.-B., & Clough, P. N. (2009). *Software Product Management and Pricing*. Berlin Heidelberg: Springer.
68. Koehler, P., Anandasivam, A., Dan, M.A., & Weinhardt, C.(2010). Customer Heterogeneity and Tariff Biases in Cloud Computing. *International Conference on Information Systems (ICIS) proceedings 2010. Paper 106*. St. Louis: Association for information systems.
69. Koulopoulos, T. M., & Champy, J. (2012). *Cloud surfing*. Brookline (MA): Bibliomotion.
70. Kourtesis, D., Alvarez-Rodríguez, J. M., & Paraskakis, I. (2014). Semantic-based QoS management in cloud systems: Current status and future challenges. *Future Generation Computer Systems*, 32, 307–323.
71. Kragelj, P., & Rajkovič, V. (2012). Kako oceniti ponudnika storitev v oblaku. *Uporabna informatika*, 20(4), 244–249.
72. Kwon, H.-K., & Seo, K.-K. (2014). A Fuzzy AHP based Multi-criteria Decision-making Model to Select a Cloud Service. *International Journal of Smart Home*, 8(3), 175–180.

73. Lee, Y.-H. (2014). A Decision Framework for Cloud Service Selection for SMEs: AHP Analysis. *Sop Transactions On Marketing Research*, 1(1), 51–61.
74. Lee, Y.-C., Tang, H., & Sugumaran, V. (2012, 29. julij): A Deployment Model for Cloud Computing using the Analytic Hierarchy Process and BCOR Analysis. *18th Americas Conference on Information Systems (AMCIS)*. Seattle. Najdeno 14. decembra 2015 na spletnem naslovu <http://aisel.aisnet.org/cgi/viewcontent.cgi?article=1108&context=amcis2012>
75. Lee, S., & Seo, K.-K. (2013). A Multi-Criteria Decision-making Model for an IaaS Provider Selection Problem. *International Journal of Advancements in Computing Technology (IJACT)*, 5(12), 363–367.
76. Leong, L., Toombs, D., & Gill, B. (2015, 18. maj). *Magic Quadrant for Cloud Infrastructure as a Service, Worldwide*. Najdeno 27. marca 2016 na spletnem naslovu <https://www.gartner.com/doc/reprints?id=1-2G2O5FC&ct=150519&st=sb>
77. Li, A., Yang, X., Kandula, S., & Zhang, M. (2010). CloudCmp: comparing public Cloud providers. *Proceedings of the 10th ACM SIGCOMM conference on Internet measurement (ACM SIGCOMM)*. New York, NY: Association for Computing Machinery.
78. Limam, N., & Boutaba, R. (2010). Assessing Software Service Quality and Trustworthiness at Selection Time. *IEEE Transactions on Software Engineering*, 36(4), 559–574.
79. Lindič, J. (2007). *Pomen kontinuiranih osredotočenih raznovrstnih inovacij v pogojih elektronskega poslovanja: Primer Amazon.com Inc.* (doktorska disertacija). Ljubljana: Univerza v Ljubljani, Ekonomska fakulteta.
80. Ling, Z., Qi, L., Qianyan, Z., & Wei, C. (2013). The Study of Adjustment and Optimization of Oracle Database in Information System. *2013 Fifth International Conference on Computational and Information Sciences (ICCIS)* (str. 442–445). Shiyang: IEEE.
81. Linthicum, D. S. (2016, 26. februar). Cloud Technology Partners. *IEEE Cloud Computing*, 3(1), 8–11.
82. Linux magazine (2008, 8. december). *The Everything as a Service (XaaS) Model*. Najdeno 18. februarja 2016 na spletnem naslovu <http://www.linux-mag.com/id/7197/>
83. Liu, S., Chan, F. T. S., & Ran, W. (izide v letu 2016). Decision making for the selection of cloud vendor: An improved approach under group decision-making with integrated weights and objective/subjective attributes. *Expert Systems with Applications*, 51–55.
84. Luoma, E., & Nyberg, T. (2011, 10. junij). Four scenarios for adoption of cloud computing in China. *19th European Conference on Information Systems* (Paper 123). Helsinki: ECIS.
85. Macías, M., & Guitart, J. (2016). Analysis of a trust model for SLA negotiation and enforcement in cloud markets. *Future Generation Computer Systems*, 55, 460–472.
86. Madria, S., & Sen, A. (2015). Offline Risk Assessment of Cloud Service Providers. *IEEE Cloud Computing*, 2 (3), 50–57.

87. Mahmood, Z., & Hill, R. (2011). *Cloud Computing for Enterprise Architectures*. London: Springer Science & Business Media.
88. Manvi, S. S., & Krishna Shyamb, G. (2014). Resource management for Infrastructure as a Service (IaaS) in cloud computing: A survey. *Journal of Network and Computer Applications*, 41, 424-440.
89. Maoz, Z. (2002). Case study methodology in international studies: from storytelling to hypothesis testing. *Evaluating Methodology in International Studies*. Ann Arbor: University of Michigan Press.
90. Marks, E.A., & Lozano, R. (2010). *Executive's guide to cloud computing*. New Jersey: John Wiley & Sons, Inc., Hoboken.
91. Martens, B., Teuteberg, F., & Gräuler, M. (2011, 13. julij). Design and implementation of a community platform for the evaluation and selection of cloud computing services: a market analysis. *European Conference on Information Systems 2011 Proceedings* (Paper 215). Helsinki: AIS.
92. Martens, B., & Teuteberg, F. (2012). Decision-making in cloud computing environments: A cost and risk based approach. *Information Systems Frontiers*, 14(4), 871-893.
93. Mather, T., Kumaraswamy, S., & Latif, S. (2009). *Cloud Security and Privacy*. Sebastopol: O'Reilly Media, Inc.
94. Mavrič, M. (2011, 28. september). Oblačna Slovenija. *MonitorPro*, poletje 2011, 36-38.
95. McCallion, J. (2012, 3. september). BSA: Europe lags behind in global cloud uptake. *Cloud Pro*. Najdeno 28. marca 2016 na spletnem naslovu <http://www.cloudpro.co.uk/cloud-essentials/general/4485/bsa-europe-lags-behind-global-cloud-uptake>
96. McDonald, K. T. (2010). *Above the clouds, Managing Risk in the World of Cloud Computing*. Cambridgeshire: IT Governance Publishing.
97. Mell, P., & Grance, T. (2011, september). The NIST Definition of Cloud Computing. *NIST - National Institute of Standards and Technology: Information Technology Laboratory*. Najdeno 4. aprila 2015 na spletnem naslovu <http://csrc.nist.gov/publications/nistpubs/800-145/SP800-145.pdf>
98. Menzel, M., Schönherr, M., & Tai, S. (2013). (MC2)2: criteria, requirements and a software prototype for Cloud infrastructure decisions. *Software: Practice and Experience*, 43(11), 1283-1297.
99. Mill, R. C. (2007). *Resorts: Management and Operation*. New Jersey: John Wiley & Sons.
100. Misra, S. C., & Mondal, A. (2011). Identification of a company's suitability for the adoption of cloud computing and modelling its corresponding Return on Investment. *Mathematical and Computer Modelling*, 53(3-4), 504-521.
101. Mohamed, A. (b.l.). A history of cloud computing. *Computer Weekly*. Najdeno 20. februarja 2016 na spletnem naslovu <http://www.computerweekly.com/feature/A-history-of-cloud-computing>

- 102.Mombrea, M. (2012, 9. november). *When to use cloud platforms vs. dedicated servers*. Najdeno 13. marca 2015 na spletnem naslovu <http://www.itworld.com/article/2832631/cloud-computing/when-to-use-cloud-platforms-vs--dedicated-servers.html>
- 103.Monteiro, L., & Vasconcelos, A. (2013). Survey on Important Cloud Service Provider Attributes Using the SMI Framework. *Procedia Technology*, 9, 253–259.
- 104.Mor, Y. (2016, 18. januar). *Gartner Report Analysis: Can Other IaaS Providers Compete With AWS and Azure?* Najdeno 27. marca 2016 na spletnem naslovu <https://www.cloudyn.com/blog/gartner-report-analysis-can-other-iaas-providers-compete-with-aws-and-azure/>
- 105.Mrvar, A. (b.l.). *Saatyev postopek*. Najdeno 10. maja 2016 na spletnem naslovu <http://mrvar.fdv.uni-lj.si/sola/info2/saaty/saaty.pdf>
- 106.Murthy, M. K. M., Ameen, M. N., Sanjay, H. A., & Yasser, P. M. (2013). Software Licensing Models and Benefits in Cloud Environment: A Survey. *Proceedings of International Conference on Advances in Computing* (str. 645–650). India: Springer.
- 107.Myerson, J. M. (2009, 3. marec). Cloud computing versus grid computing. *IBM developerWorks*. Najdeno 15. marca 2016 na spletnem naslovu <http://www.ibm.com/developerworks/web/library/wa-cloudgrid/>
- 108.Nabi, M., Toeroe, M., & Khendek, F. (2016). Availability in the cloud: State of the art. *Journal of Network and Computer Applications*, 60, 54–67.
- 109.Ngan, L., & Kanagasabai, R. (2012). OWL-S Based Semantic Cloud Service Broker. *2012 IEEE 19th International Conference on Web Services (ICWS)* (str. 560–567). Honolulu: IEEE.
- 110.NIST (2011, 25. oktober). *Final Version of NIST Cloud Computing Definition Published*. Najdeno 4. aprila 2015 na spletnem naslovu <http://www.nist.gov/itl/csd/cloud-102511.cfm>
- 111.Olavsrud, T. (2016, 26. januar). 11 cloud trends that will dominate 2016. *CIO*. Najdeno 29. marca 2016 na spletnem naslovu <http://www.cio.com/article/3026527/cloud-computing/11-cloud-trends-that-will-dominate-2016.html>
- 112.Oracle (b.l.). *Database Installation Guide*. Najdeno 22. marca 2016 na spletnem naslovu https://docs.oracle.com/cd/B28359_01/install.111/b32002/pre_install.htm#i1011296
- 113.Park, J.H. (2012). A Virtualization Security Framework for Public Cloud Computing. *Computer Science and its Applications*, 203, 421-428.
- 114.Patton, M. Q. (1987). *How to Use Qualitative Methods in Evaluation*. Thousand Oaks: Sage Publications.
- 115.Podobnik, K. (2015, 14. december). *Uporaba AHP metode pri upravljanju projektnega portfelja z vidika uporabnika* (magistrsko delo). Ljubljana: Univerza v Ljubljani, Fakulteta za gradbeništvo in geodezijo.

116. Pohekar, S. D., & Ramachandran, M. (2004). Application of multi-criteria decision making to sustainable energy planning—A review. *Renewable and Sustainable Energy Reviews*, 8(4), 365–381.
117. Pohorec, S. (2011, 28. september). Skrite pasti računalništva v oblaku. *MonitorPro*, poletje 2011, str. 38–42.
118. Port, L. (2009, junij). ASP vs. SaaS, Fun with acronyms or a sweeping technological shift? *Peer to Peer, Doing More With Less*. Najdeno 2. januarja 2016 na spletnem naslovu <http://www.jdsupra.com/documents/616a4013-582e-495b-8dea-b8353b625e8e.pdf>
119. Preimesberger, C. (2015, 24. julij). *Big Four IaaS Providers Now Own Half the Market*. Najdeno 27. marca 2016 na spletnem naslovu <http://www.eweek.com/cloud/big-four-iaas-providers-now-own-half-the-market.html>
120. Qu, L., Wang, Y., Orgun, M. A., Liu, L., Liu, H., & Bouguettaya, A. (2015). CCCloud: Context-Aware and Credible Cloud Service Selection Based on Subjective Assessment and Objective Assessment. *IEEE Transactions on Services Computing*, 8(3), 369–383.
121. Quinton, C., Romero, D., & Duchien, L. (2014). Automated Selection and Configuration of Cloud Environments Using Software Product Lines Principles. *2014 IEEE 7th International Conference on Cloud Computing (CLOUD)* (str. 144–151). Anchorage, AK: IEEE.
122. Redkar T. (2009). *Windows Azure Platform*. Berkeley: Apress.
123. Relatively Interesting (2012, 24. februar). *Comparing today's computers to 1995's*. Najdeno 4. aprila 2015 na spletnem naslovu <http://www.relativelyinteresting.com/comparing-todays-computers-to-1995s/>
124. Repschläger, J., Wind, S., Zarnekow, R., & Turowski, K. (2011). Developing a Cloud Provider Selection Model. *Presented at the Enterprise modelling and information systems architectures (EMISA 2011)*, 163–176. Bonn: Gesellschaft für Informatik. Najdeno 3. januarja 2015 na spletnem naslovu <http://cs.emis.de/LNI/Proceedings/Proceedings190/article6455.html>
125. Repschläger, J., Wind, S., Zarnekow, R., & Turowski, K. (2013, 29. julij). Decision Model for Selecting a Cloud Provider: A Study of Service Model Decision Priorities. *19th Americas Conference on Information Systems*. Chicago, Illinois: Association for Information Systems. Najdeno 3. januarja 2015 na spletnem naslovu <http://aisel.aisnet.org/cgi/viewcontent.cgi?article=1108&context=amcis2012>
126. Rhoton, J., & Haukioja, R. (2013). *Cloud Computing Architected: Solution Design Handbook*. b.k.: Recursive press.
127. RightScale. (2016, 9. februar). *Cloud Computing Trends: 2016 State of the Cloud Survey*. Najdeno 19. februarja 2016 na spletnem naslovu <http://assets.rightscale.com/uploads/pdfs/RightScale-2016-State-of-the-Cloud-Report.pdf>

- 128.Rodrigues, T. (2013, 27. avgust). *Top cloud IaaS providers compared*. Najdeno 27. marca 2016 na spletnem naslovu <http://www.techrepublic.com/blog/the-enterprise-cloud/top-cloud-iaas-providers-compared/>
- 129.Rodríguez-Haro, F., Freitagb, F., Navarrob, L., Hernández-Sánchez, E., Farías-Mendozaa, N., Guerrero-Ibáñez, J. A., & González-Potesa, A. (2012). A summary of virtualization techniques. *Procedia Technology*, 3, 267–272.
- 130.Royal Pingdom (2008, 11. april). *Map of all Google data center locations*. Najdeno 18. septembra 2011 na spletnem naslovu <http://royal.pingdom.com/2008/04/11/map-of-all-google-data-center-locations/>
- 131.Saaty, T.L. (1977) A scaling method for priorities in hierarchical structures. *Journal of Mathematical Psychology* 15, 234-281.
- 132.Saaty, T. L. (1994). *Fundamental of Decision Making and Priority Theory With the Analytic Hierarchy Process*. Pittsburgh: RWS Publications.
- 133.Saaty, L. T. (2001). *Decision making for leaders: The analytic hierarchy theory with the analytic hierarchy process*. Pittsburgh: RWS Publications.
- 134.Saaty, T. L., (2008). Decision making with the analytic hierarchy process. *International Journal of Services Sciences*, (1)1, 83 - 98.
- 135.Saaty, T. L., & Vargas, L. G. (1982). *The logic of priorities applications in business, energy, health, transportation*. Boston: Kluwer-Nijhoff Publishing.
- 136.Saaty, T. L., & Vargas, L. G. (2006). *Decision Making with the Analytic Network Process*. New York: Springer Science+Business Media, LLC.
- 137.Saaty, T. L., & Vargas, L. G. (2012). *Models, methods, concepts & applications of the analytic hierarchy process*. New York:Springer.
- 138.Sabi, H. M., Uzoka, F.-M. E., Langmia, K., & Njeh, F. N. (2015). Conceptualizing a model for adoption of cloud computing in education. *International Journal of Information Management*, 36(2), 183-191.
- 139.Saripalli, P., & Pingali, G. (2011). MADMAC: Multiple Attribute Decision Methodology for Adoption of Clouds. *2011 IEEE International Conference on Cloud Computing (CLOUD)* (str. 316–323). Washington, DC: IEEE.
- 140.Sarna, D. E. Y. (2010). *Implementing and Developing Cloud Computing Applications*. Boca Raton: Auerbach Publications.
- 141.Schmelzer, R. (2008, april). What does service performance mean? *TechTarget SearchSOA*. Najdeno 15. marca 2016 na spletnem naslovu <http://searchsoa.techtarget.com/tip/What-does-service-performance-mean>
- 142.Schniederjans, M. J., Hamaker, J. L., & Schniederjans, A. M. (2010). *Information technology investment: decision-making methodology* (2nd ed.). Singapore: World Scientific Publishing Co. Pte. Ltd.
- 143.Schouten, E. (2012, 12. september). *Rapid elasticity and the cloud*. Najdeno 6. marca 2016 na spletnem naslovu <http://www.thoughtsoncloud.com/2012/09/rapid-elasticity-and-the-cloud/>

- 144.Serrano, D., Bouchenak, S., Kouki, Y., de Oliveira Jr., F. A., Ledoux, T., Lejeune, J., ... Sens, P. (2016). SLA guarantees for cloud services. *Future Generation Computer Systems*, 54, 233–246.
- 145.Siegel, J., & Perdue, J. (2012). Cloud Services Measures for Global Use: The Service Measurement Index (SMI). *SRII Global Conference (SRII), 2012 Annual* (str. 411–415). San Jose, CA: IEEE.
- 146.Silas, S., Rajsingh, E. B., & Ezra, K. (2012). Efficient Service Selection Middleware using ELECTRE Methodology for Cloud Environments. *Information Technology Journal*, 11(7), 868–875.
- 147.Smart, W. (2012). Bring on the Broadband Regional Australia. *AQ: Australian Quarterly*, 83(1), 4–35.
- 148.Smith, D. M. (2014, 1. oktober). The Top 10 Cloud Myths. *Gartner*. Najdeno 19. marca 2016 na spletnem naslovu <https://www.gartner.com/doc/2860422/top--cloud-myths>
- 149.Smith, D.M., & Cearley, D. W. (2008, 18. september). Contrasting Perspectives on Cloud Computing. *Gartner*. Najdeno 4.aprila 2015 na spletnem naslovu <https://www.gartner.com/doc/759535/contrasting-perspectives-cloud-computing>
- 150.Son, S., Jung, G., & Jun, S. C. (2013). An SLA-based cloud computing that facilitates resource allocation in the distributed data centers of a cloud provider. *The Journal of Supercomputing*, 64(2), 606–637.
- 151.Stotler, L. (2015, 27. april). *AWS Still Leading IaaS Space with Microsoft Gaining Revenues*. Najdeno 28. marca 2016 na spletnem naslovu <http://www.transformingnetworkinfrastructure.com/topics/hyperscale-data-centers/articles/402301-aws-still-leading-iaas-space-with-microsoft-gaining.htm>
- 152.Studt, C., & Hofstader, J. (2009, 1. maj). CaaS: Communications-as-a-Service. *Dr.Dobb's, The world of software development*. Najdeno 19.februarja 2016 na spletnem naslovu <http://www.drdoobs.com/web-development/caas-communications-as-a-service/217600340>.
- 153.Sun, L., Dong, H., Hussain, F. K., Hussain, O. K., & Chang, E. (2014). Cloud service selection: State-of-the-art and future research directions. *Journal of Network and Computer Applications*, 45, 134–150.
- 154.Sun, L., Ma, J., Zhang, Y., Dong., H., & Hussain, F.K.(izide v aprilu 2016). Cloud-FuSeR: Fuzzy ontology and MCDM based cloud service selection. *Future Generation Computer Systems*, 57, 42-55.
- 155.Sundareswaran, S., Squicciarini, A., & Lin, D. (2012). A Brokerage-Based Approach for Cloud Service Selection. *2012 IEEE 5th International Conference on Cloud Computing (CLOUD)* (str. 558–565). Honolulu: IEEE.
- 156.Tang, C., & Liu, J.(2015). Selecting a trusted cloud service provider for your SaaS program. *Computers & Security*, 50, 60-73.
- 157.*The Case Study as a Research Method*. Najdeno 20. marca 2016 na spletnem naslovu <https://www.ischool.utexas.edu/~ssoy/usesusers/l391d1b.htm>

158. Tinnefeld, C. (2015). *Building a Columnar Database on RAMCloud: Database Design for the Low-Latency Enabled Data Center*. B.k.: Springer.
159. Toeroe, M., & Tam, F. (2012). *Service Availability: Principles and Practice*. Chichester: John Wiley & Sons.
160. Triantaphyllou, E. (2000). *Multi-criteria decision making methods: a comparative study*. Boston: Springer.
161. Tucker, P. (2008, 31. marec). *Bringing the cloud with you - Docs Blog by Google Docs*. Najdeno 27. maja 2015 na spletnem naslovu <http://googledocs.blogspot.com/2008/03/bringing-cloud-with-you.html>
162. ur Rehman, Z., Hussain, F. K., & Hussain, O. K. (2011). Towards Multi-criteria Cloud Service Selection. *Fifth International Conference on Innovative Mobile and Internet Services in Ubiquitous Computing (IMIS)* (str. 44-48). Seoul: IEEE.
163. ur Rehman, Z., Hussain, O. K., & Hussain, F. K. (2013). Multi-criteria IaaS Service Selection Based on QoS History. *2013 IEEE 27th International Conference on Advanced Information Networking and Applications (AINA)* (str. 1129–1135). Barcelona: IEEE.
164. Vaquero, L.M., Roderio-Merino, L., Caceres, J., & Lindner, M. (2008). A break in the clouds: towards a cloud definition. *ACM SIGCOMM Computer Communication Review*, 39 (1), 50-55.
165. Vaughan-Nichol, S.J. (2006). New Approach to Virtualization Is a Lightweight. *IEEE Computer Society*, 39 (11), 12-14.
166. Vazquez-Poletti, J.L., Santos-Muzon, D., Florente, I.M., & Valero, F. (2014). A Cloud for Clouds: Weather Research and Forecasting on a Public Cloud Infrastructure. *Cloud Computing and Services Sciences: International Conference in Cloud Computing and Services Sciences*, 512, 3-11.
167. Wang, J.-J., Lin, Z.-K., & Huang, H. (2008). A Decision Model for Information Systems Outsourcing: Using a Multicriteria Method. *Journal of Service Science and Management*, 1(1), 1–10.
168. Wang, L. (2015, 17. avgust). *Is Oracle Too Expensive?* Najdeno 23. marca 2016 na spletnem naslovu <https://www.linkedin.com/pulse/oracle-too-expensive-leo-wang>
169. Whaiduzzaman, M., Gani, A., Anuar, N. B., Shiraz, M., Haque, M. N., & Haque, I. T. (2014, 13. februar). Cloud Service Selection Using Multicriteria Decision Analysis. *The Scientific World Journal*, 2014.
170. Williams, L. G., & Smith, C. U. (2004). Web Application Scalability: A Model-Based Approach. *Software Engineering Research and Performance Engineering Services*. Najdeno 19. marca na spletnem naslovu <http://www.perfeng.com/papers/scale04.pdf>
171. Willis, B. (2014, 5. julij). The Advantages and Limitations of Single Case Study Analysis. *E-International Relations Students*. Najdeno 19. marca na spletnem naslovu <http://www.e-ir.info/2014/07/05/the-advantages-and-limitations-of-single-case-study-analysis/>
172. Wittern, E., Kuhlenkamp, J., & Menzel, M. (2012). Erratum: Cloud Service Selection Based on Variability Modeling. V C. Liu, H. Ludwig, F. Toumani, & Q. Yu (ur.),

- Service-Oriented Computing* (str. E1–E1). Berlin Heidelberg: Springer. Najdeno 3.marca 2016 na spletnem naslovu http://link.springer.com/chapter/10.1007/978-3-642-34321-6_63
173. Yang, J., Lin, W., & Dou, W. (2013). An adaptive service selection method for cross-cloud service composition. *Concurrency and Computation: Practice and Experience*, 25(18), 2435–2454.
 174. Ye, Z., Bouguettaya, A., & Zhou, X. (2012). QoS-Aware Cloud Service Composition Based on Economic Models. V C. Liu, H. Ludwig, F. Toumani, & Q. Yu (ur.), *Service-Oriented Computing* (str. 111–126). Berlin Heidelberg: Springer. Najdeno 3.marca 2016 na spletnem naslovu http://link.springer.com/chapter/10.1007/978-3-642-34321-6_8
 175. Yeluri, R., & Castro-Leon, E. (2014). *Building the Infrastructure for Cloud Security*. B. k.: Apress.
 176. Yin, R. K. (2009). *Case Study Research: Design and Methods* (4th ed.). Thousand Oaks: Sage Publications.
 177. Yu, Q. (2014). CloudRec: a framework for personalized service Recommendation in the Cloud. *Knowledge and Information Systems*, 43(2), 417–443.
 178. Zeng, W., Zhao, Y., & Zeng, J. (2009). Cloud Service and Service Selection Algorithm Research. *Proceedings of the First ACM/SIGEVO Summit on Genetic and Evolutionary Computation* (str. 1045–1048). New York, NY, USA: ACM.
 179. Zhang, S., Chen, X., Zhang, S., & Huo, X. (2010). The Comparison Between Cloud Computing and Grid Computing. *Zbornik V11. International Conference on Computer Application and System Modeling 2010* (str. V11-72 - V11-75). Taiyuan: North University of China.
 180. Zhang, M., Ranjan, R., Nepal, S., Menzel, M., & Haller, A. (2012). A Declarative Recommender System for Cloud Infrastructure Services Selection. *Proceedings of the 9th international conference on economics of grids, clouds, systems and services (GECON)*. Berlin: Springer.
 181. Zhao, L., Ren, Y., Li, M., & Sakurai, K. (2012). Flexible service selection with user-specific QoS support in service-oriented architecture. *Journal of Network and Computer Applications*, 35(3), 962–973.
 182. Zheng, Z., Wu, X., Zhang, Y., Lyu, M. R., & Wang, J. (2013). QoS Ranking Prediction for Cloud Services. *IEEE Transactions on Parallel and Distributed Systems*, 24(6), 1213–1222.
 183. Zhonglin, H., & Yuhua, H. (2011). A Study on Cloud Backup Technology and Its Development. *Innovative Computing and Information* (str. 1–7). Berlin Heidelberg: Springer.