

UNIVERZA V LJUBLJANI
EKONOMSKA FAKULTETA

MAGISTRSKO DELO

**UPORABA KONCEPTA MANAGEMENTA PRODUKTA
PROGRAMSKE OPREME Z NAMENOM PODPORE
UVEDBI SKALABILNEGA POSLOVNEGA MODELA PRI
PRODAJI INFORMACIJSKE REŠITVE**

Ljubljana, avgust 2016

MARKO BIZJAK

IZJAVA O AVTORSTVU

Podpisani Marko Bizjak, študent Ekonomske fakultete Univerze v Ljubljani, avtor predloženega dela z naslovom Uporaba koncepta managementa produkta programske opreme z namenom podpore uvedbi skalabilnega poslovnega modela pri prodaji informacijske rešitve, pripravljenega v sodelovanju s svetovalko prof. dr. Mojco Indihar Štemberger

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 plagiatstvo – 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 plagiatstvo 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.

KAZALO

UVOD	1
1 KONCEPT MANAGEMENTA PRODUKTA PROGRAMSKE OPREME.....	4
1.1 Produkt programske opreme.....	4
1.1.1 Struktura produkta programske opreme	6
1.1.2 Klasifikacija produktov programske opreme	7
1.1.3 Stopenjski model faz v življenjskem ciklu produkta programske opreme.....	7
1.1.4 Razlikovanje med produktom programske opreme in konkretno namestitveno instanco	8
1.2 Motivacija za vpeljavo koncepta managementa produkta programske opreme.....	9
1.2.1 Definicija managementa produkta programske opreme.....	10
1.2.2 Strategija podjetja v povezavi s produktom programske opreme	11
1.3 Vloge v okviru managementa produkta programske opreme.....	12
1.4 Aktivnosti pri managementu produkta programske opreme	15
1.4.1 Referenčni model managementa produkta programske opreme po Weerdu et al.	16
1.4.2 Model managementa produkta programske opreme po Kittlausu in Cloughu...	22
2 SKALABILNOST PRODAJE INFORMACIJSKIH REŠITEV	30
2.1 Razvoj poslovnih modelov v panogi informacijske tehnologije	30
2.2 Rast prodaje in skalabilnost prodaje.....	32
2.3 Definicija skalabilnosti in komponente skalabilnega poslovnega modela	32
2.3.1 Komponente skalabilnega poslovnega modela	32
2.3.2 Model razvoja in prodaje informacijskih rešitev in masovne prilagoditve	33
2.4 Dejavniki zagotavljanja dolgoročne skalabilnosti	34
2.4.1 Ljudje in njihove vloge.....	34
2.4.2 Organizacijska struktura.....	37
2.4.3 Voditeljstvo	41
3 DELOVNI PROCESI NA INFORMACIJSKI REŠITVI ADINSURE.....	42
3.1 Vizija in strategija podjetja.....	43
3.2 Informacijska rešitev AdInsure	43
3.2.1 Kratek opis informacijske rešitve	43
3.2.2 Pregled načina razvoja informacijske rešitve	45
3.3 Organizacijska struktura	46
3.3.1 Poslovna enota Zavarovalništvo.....	46
3.3.2 Oddelek prodaje in trženja.....	50
3.3.3 Vodstvo podjetja.....	50
3.4 Strategija produkta AdInsure.....	50
4 ANALIZA OBSTOJEČIH PROCESOV IN PREDLOG SPREMEMB GLEDE NA UGOTOVLJENA SPOZNANJA	51
4.1 Izvajanje funkcije analize trga.....	52

4.2 Izvajanje funkcije produktne analize.....	54
4.3 Izvajanje aktivnosti v povezavi s funkcijo strategije produkta	56
4.4 Izvajanje aktivnosti v funkciji planiranja produkta.....	58
4.5 Izvajanje aktivnosti v funkciji razvoja	59
4.6 Izvajanje funkcije trženja	62
4.7 Izvajanje funkcije prodaje in distribucije	63
4.8 Izvajanje funkcije podpore in storitev	65
4.9 Temeljne aktivnosti po vlogah v poslovni enoti Zavarovalništvo	66
4.10 Predlagane spremembe v organizacijski strukturi.....	68
SKLEP.....	69
LITERATURA IN VIRI.....	72

KAZALO SLIK

Slika 1: Management produkta in vloga managerja produkta.....	13
Slika 2: Hierarhija entitet managementa produkta	17
Slika 3: Referenčni model za management produkta programske opreme po Weerd et al.	18
Slika 4: Organizacijska struktura poslovne enote Zavarovalništvo	47
Slika 6: Razdelitev izvajanja aktivnosti v okviru funkcije analize trga za informacijsko rešitev AdInsure	53
Slika 7: Razdelitev izvajanja aktivnosti v okviru funkcije produktne analize za informacijsko rešitev AdInsure	55
Slika 8: Razdelitev izvajanja aktivnosti v okviru funkcije strategije produkta za informacijsko rešitev AdInsure	56
Slika 9: Razdelitev izvajanja aktivnosti v okviru funkcije planiranja produkta za informacijsko rešitev AdInsure	58
Slika 10: Razdelitev izvajanja aktivnosti v okviru funkcije razvoja produkta za informacijsko rešitev AdInsure	60
Slika 11: Razdelitev izvajanja aktivnosti v okviru funkcije trženja za informacijsko rešitev AdInsure	62
Slika 12: Razdelitev izvajanja aktivnosti v okviru funkcije prodaje in distribucije za informacijsko rešitev AdInsure	64
Slika 13: Razdelitev izvajanja aktivnosti v okviru funkcije podpore in storitev za informacijsko rešitev AdInsure	65
Slika 14: Predlagana sprememba organizacijske strukture	69

KAZALO TABEL

Tabela 1: Tip programske opreme na primeru informacijske rešitve AdInsure	9
Tabela 2: Instanca programske opreme na primeru konkretne implementacije informacijske rešitve AdInsure za slovensko stranko	9

Tabela 3: Model managementa produkta programske opreme po Kittlausu in Cloughu	23
Tabela 4: Izvajanje temeljnih aktivnosti po vlogah v podjetju za informacijsko rešitev AdInsure	67

UVOD

Programski produkt ali rešitev gre v svojem življenjskem ciklu skozi veliko razvojnih faz. V določenih fazah je produkt samo enkrat v svojem življenjskem ciklu, določene pa se skozi življenjski cikel večkrat ponovijo. Trajanje faz je različno dolgo. Obstoj programskega produkta ali rešitve je podprt z različnimi procesi, ki pomembno vplivajo na potek vsake faze v življenjskem ciklu produkta in določajo smer, v katero se faza razvija, z njo pa sam programski produkt.

Življenjski cikel programskega produkta v splošnem delimo na fazo planiranja in načrtovanja, fazo implementacije, testiranja in dokumentiranja ter fazo vzdrževanja. Procesi oziroma funkcije znotraj organizacije, ki pomembno vplivajo na vsako od prej omenjenih faz so: analiza trga, na katerem se produkt pojavlja, priprava strategije razvoja in prodaje ter prodaja in trženje.

Vsaka aktivnost v okviru procesov življenjskega cikla programskega produkta ima svoje parcialne cilje. Ti cilji so večinoma definirani v okviru skupine ali oddelka v organizacijski strukturi podjetja. Ker so definirani na nivoju oddelka, predstavljajo optimalno izbiro za to skupino ali oddelek, kar pa ne opredeljuje nujno optimalne izbire za razvoj in potek življenjskega cikla produkta, katerega avtor oziroma lastnik je podjetje.

Aktivnosti, ki so izvajane v okviru posameznega oddelka, imajo posledice tudi izven tega oddelka. Tako se različni oddelki s svojimi aktivnostmi vključujejo v različnih fazah življenjskega cikla programskega produkta in s svojo udeležbo, odločitvami in usmeritvami vplivajo na potek razvoja programskega produkta. Posledice teh vplivov so lahko kratkoročne ali dolgoročne. Cilji oddelka v določeni fazi življenjskega cikla so lahko ozko usmerjeni, parcialni in ne upoštevajo širšega konteksta v življenjskem ciklu produkta.

Z namenom celovitega upravljanja vseh procesov in aktivnosti, povezanih s produktom, sta se razvila zahteva in koncept upravljanja produkta, ki skrbita za to, da so vse aktivnosti, ki lahko vplivajo na potek življenjskega cikla produkta, smiselno povezane in koordinirane. Na tak način se ohranja kvaliteta produkta na dovolj visokem nivoju, s tem pa se dlje časa ohranja vrednost za podjetje.

Prvotno se je koncept managementa produkta razvil in uporabljal v tradicionalnih industrijskih panogah (Gorchels, 2006, str. 5), Kilpi (1997) pa je koncept konec dvajsetega stoletja začel uporabljati tudi v panogi informacijske tehnologije.

Med cilji podjetij, ki se ukvarjajo z razvojem lastnih produktov programske opreme, je eden glavnih ciljev tudi čim daljše ohranjanje vrednosti lastnega produkta. Poleg tega pa je istočasno med glavnimi cilji, predvsem zasebnih podjetij, rast prodaje in povečanje dobička. S skalabilnim poslovnim modelom je ta cilj lažje dosegljiv. Temeljna lastnost skalabilnega poslovnega modela je namreč ta, da rast prodaje in dobiček nista

premosorazmerna, pač pa se dobiček povečuje hitreje. Ta model se v praksi veliko implementira v podjetjih informacijske tehnologije, vendar morajo biti za to vzpostavljeni ustrezni pogoji, ki jih glede na zunanje okoliščine ustvari predvsem vsako podjetje samo.

V podjetjih informacijske tehnologije se beseda skalabilnost velikokrat interpretira ozko gledano na samo implementacijo informacijske rešitve. To pomeni, da se izvajajo analize v smeri iskanja možnosti izboljšave obstoječe informacijske rešitve, kateri bi z zagotovitvijo dodatnih virov omogočili razširitev tako, da bi omogočala večje število uporabnikov z enakim odzivnim časom.

Skalabilnost prodaje se, ozko gledano, obravnava samo kot iskanje možnosti kako je mogoče z malo ali sploh brez kakršnih koli prilagoditev večkrat prodati obstoječo rešitev, ki poveča dobiček z manj vloženimi viri.

Uspešna implementacija skalabilnega poslovnega modela ne temelji samo na tehnologiji. Abbot in Fischer (2015, str. 2) na podlagi konkretnih primerov, v katere sta bila vključena kot vodji ali kot svetovalca, trdita, da se vprašanje implementacije skalabilnosti začne v podjetju samem ter ljudeh, zaposlenih, ki to podjetje sestavljajo. Nato se to vprašanje širi naprej na tehnologijo in procese. Ljudje sprejemajo tako tehnične kot poslovne odločitve. Odločitve se sprejemajo na podlagi informacij, ki so znane in na voljo odločevalcu. Tako se na primer zgodi, da so zaradi slabe informiranosti o tehničnih podrobnostih odločitve glede določenih tehničnih rešitev pri implementaciji napačne, kar se na koncu lahko izraža kot slaba skalabilnost tehnične rešitve oziroma celotnega produkta. Abbot in Fischer (2015, str. 3) nadaljujeta, da se ljudje v organizacijah ne ukvarjajo z razvojem procesov, ki bi pomagali pri učenju iz napak storjenih v preteklosti in včasih vpeljujejo preveč procesov, katerih izvajanje kvečjemu še sili organizacijo v slabe odločitve ali odločitve, sprejete prepozno za pravi učinek.

Abbot in Fischer (2015, str. 3) tudi pravita, da mora biti informacijska rešitev, s katero želimo izboljšati skalabilnost, zasnovana na arhitekturi, ki omogoča skalabilnost. To pomeni, da je potrebno obstoječo rešitev s ciljem povečanja skalabilnosti arhitekturno prilagoditi, kar pa ni vedno mogoče ali stroškovno upravičeno. Zato je pri odločitvi o razvoju obstoječe rešitve v smeri produkta programske opreme nujno analizirati stanje rešitve glede na obstoječe stopenjske modele, ki podajajo odgovor glede smiselnosti in možnosti uvedbe večjih arhitekturnih sprememb v obstoječih rešitvah. Informacijske rešitve tekom razvoja in uporabe namreč prehajajo različne faze, ki opredeljujejo značilnosti rešitve in aktivnosti, izvajane nad rešitvijo, ter možnosti nadaljnjih aktivnosti in smeri razvoja informacijske rešitve. Takšen stopenjski model, ki ga bomo omenili še v nadaljevanju tega dela, sta definirala tudi Rajlich in Bennett (2000).

Izboljšanje skalabilnosti aplikacije in vpeljava skalabilnega poslovnega modela torej nista odvisna samo od arhitekture informacijske rešitve, ampak kot trdita Abbot in Fischer (2015, str. 2–3), predvsem od ljudi in ustreznih organizacijskih procesov.

Namen magistrskega dela je oblikovati izhodišča za opredelitev sprememb v procesih življenjskega cikla obstoječe informacijske rešitve, s pomočjo katerih podjetja v panogi informacijskih tehnologij izboljšajo skalabilnost prodaje informacijske rešitve ter s tem podpirajo skalabilen poslovni model.

Cilji magistrskega dela so v prvem delu:

- podrobna teoretična preučitev koncepta managementa produkta programske opreme;
- analiza praktične uporabe koncepta managementa produkta programske opreme na konkretnih primerih;
- izdelava pregleda literature glede oblik in načinov implementacije skalabilnih poslovnih modelov ter procesov, ki razvoj teh modelov podpirajo;
- ugotoviti dejavnike, ki so pomembni za izboljšanje skalabilnosti.

S temi teoretičnimi osnovami želimo ugotoviti, ali obstaja sorazmerje med uspešnim managementom produkta programske opreme in skalabilnostjo prodaje, ter definirati izhodišča za drugi del magistrskega dela.

Cilj v drugem delu magistrskega dela je aplikacija spoznanj na realnem, praktičnem primeru. Pri tem želimo ugotoviti pomanjkljivosti v trenutnih procesih dela in opredeliti konkretne operativne korake, s katerimi bi v podjetju dosegli višjo oceno stopnje zrelosti v konceptu managementa produkta programske opreme. Z vpeljavo sprememb želimo doseči izboljšanje kvalitete same informacijske rešitve, izboljšanje zadovoljstva strank in izboljšati možnosti za rast prodaje informacijske rešitve AdInsure v podjetju Adacta d.o.o.

Pri izdelavi magistrskega dela bom uporabil znanja, pridobljena tekom študija, ter praktična in teoretična znanja, ki sem jih pridobil v več kot 10 letih dela na projektih razvoja informacijskih rešitev.

V prvem delu je poudarek na teoretičnem preučevanju obstoječe literature s področja managementa produkta programske opreme ter okoliščin in procesov, ki vplivajo na skalabilnost informacijskih rešitev in procesov, ki podpirajo skalabilne poslovne modele, kar bo predstavljalo teoretično osnovo za nadaljnje delo.

Obstoječi način dela pri razvoju informacijske rešitve bom izvedel z lastnim opazovanjem dela in metodo intervjujev. Pridobljene informacije v tem delu bom s pomočjo metode dedukcije in sinteze uporabil za določanje potrebnih korakov in sprememb v načinu dela, s katerimi bi dolgoročno izboljšali kvaliteto informacijske rešitve in njeno prodajno skalabilnost.

1 KONCEPT MANAGEMENTA PRODUKTA PROGRAMSKE OPREME

Management produkta skozi njegov celoten življenjski cikel se je najprej uveljavil v drugih panogah in se nato prenesel v panogo informacijskih tehnologij.

Uporaba managementa produkta kot ekspliciten koncept se je najprej pojavil v podjetju Procter&Gamble leta 1931, ko sta bila dvema konkurenčnima produktoma dodeljena managerja produkta (Gorchels, 2006). Od takrat se je osnovna ideja razširila na različne panoge.

Izraz management produkta programske opreme (angl. *Software Product Management*) prvič uporabi Kilpi (1997), ki koncept managementa produkta aplicira v organizaciji, ki se ukvarja z razvojem programske opreme. Empirične raziskave so v nekaj letih uporabe koncepta v življenjskih ciklih produktov pokazale, da ima uporaba managementa produkta programske opreme pozitivne učinke tako na porabljen čas namenjen produktu znotraj organizacijskih enot kot na področju zaostankov pri dobavi in kvaliteti samega produkta (Maglyas, Nikula, & Smolander, 2011). Prednosti uporabe managementa produkta programske opreme se kažejo tudi v profitabilnosti, boljšem predvidevanju poteka življenjskega cikla produkta in ujemanju s poslovnimi cilji (Maglyas et al., 2011).

Namen koncepta managementa produkta programske opreme je, da se vzpostavi enovit nadzor nad vsemi procesi, ki lahko vplivajo na smer razvoja programskega produkta od povsem začetne faze načrtovanja in vzpostavljanja zahtev do zaključne faze vzdrževanja in končnega zapiranja programskega produkta, ko ta ni več v uporabi.

Ebert in Brinkkemper (2014) trdita, da je management produkta programske opreme ključni dejavnik za uspeh produkta programske opreme, saj je vključen v celoten življenjski cikel produkta. Tako sta prek tega zagotovljeni tako tehnična kot poslovna perspektiva pogleda na produkt programske opreme. Z različnimi pogledi na različne poslovne procese in nosilce teh procesov skozi celotno življenjsko obdobje produkta je management produkta programske opreme glavna gonilna sila in vodilo za vse zahteve, ki jih doseže s svojo konsistentno osredotočenostjo in orientacijo na vrednost produkta.

1.1 Produkt programske opreme

Pojem produkt programske opreme se večinoma uporablja v kontekstu programske opreme, ki je razvita z namenom prodaje in generiranja direktnih materialnih koristi njenega avtorja. S pomočjo definicije produkta v splošnem pridemo tudi do definicije produkta programske opreme.

V literaturi je navedenih več definicij **produkta**, tri izpostavljam v nadaljevanju.

Kotler in Armstrong (2012, str. 224) definirata produkt kot kar koli, kar je mogoče ponuditi na trgu z namenom, da pritegne pozornost, je uporabno in uporabljeno, in s čimer zadosti določeno željo ali potrebo. Pri tem produkti niso le oprijemljivi predmeti, kot so avtomobili ali računalniki, pač pa so v to zajete tudi storitve, dogodki, osebe, kraji, organizacije ali kombinacije vsega naštetega.

Ficker (2012) definira produkt kot dobrino, s katero zadovoljimo skupino ljudi ali organizacij s primerljivimi potrebami, katere uporabo se nadomesti v neki drugi obliki, ki pa ni nujno v obliki plačila.

Kittlaus in Clough (2009, str. 6) definirata produkt kot kombinacijo dobrin in storitev, katere posameznik ali skupina poveže zaradi svojih komercialnih interesov, katerih rezultat prenese na stranko. Nadalje definirata produkt programske opreme kot produkt, katerega primarna dobrina je programska oprema. Ficker (2012) dodaja, da je programska oprema informacijska dobrina, ki je izražena v obliki bitov, kar je karakteristična razlika programske opreme napram drugim produktom.

Vsi ti avtorji opredeljujejo dejstvo, da stranka ne pomeni nujno podjetje ali drugi subjekt, ki je za produkt kaj plačal. Komercialni interes ob kreiranju produkta je še precej širši. Ta interes lahko pomeni vzpostavitev konkurence na trgu, tržne prepoznavnosti ali samo osebnega zadoščenja. Tako Kittlaus in Clough (2009, str. 6) kot tudi Ficker (2012) poudarjajo, da se neka stvar ne imenuje produkt šele, ko je prodana, pač pa jo že sam namen za njeno prodajo, pritegnitev pozornosti ali uporabo definira kot produkt.

Pojem **programska oprema** sta Kittlaus in Clough (2009, str. 5) opisala kot neoprijemljivo ekonomsko dobrino, ki je zaznana po svoji funkcionalnosti. Programska oprema je tako zaznana po uporabniškem vmesniku ali kot sam rezultat operacij programske opreme.

Definicijo **produkta programske opreme** tako Kittlaus in Clough (2009, str. 6) zapišeta kot produkt, katerega temeljna komponenta je programska oprema.

V nadaljevanju bo večkrat omenjen pojem **storitve**, zato podajamo njeno definicijo, kot jo opredelita Kotler in Armstrong (2012, str. 224): »Storitve so oblika produkta, ki je sestavljen iz aktivnosti, koristi ali zadovoljstva in namenjen za prodajo. Storitve so v svojem bistvu neopredmetene in ne rezultirajo v materialni lasti česarkoli.«. Primeri storitev so tako turistične storitve, bančne storitve, komunikacijske storitve, ipd.

Programska oprema je danes nepogrešljiva v vsakdanjem življenju. Njena uporaba in rezultati pomembno vplivajo na ekonomske odločitve tako na nivoju posameznika kot podjetja ali države. Življenja brez nje si v veliki meri sploh ni mogoče več predstavljati, saj nas spremlja na vsakem koraku. Trdimo lahko, da je ustrezna uporaba programske opreme v procesih podjetja ključna za njegovo konkurenčnost in njegov uspeh ali neuspeh na trgu.

V grobem programsko opremo delimo na:

- sistemsko programsko opremo,
- uporabniško programsko opremo.

Sistemska programska oprema je tista, ki podpira samo uporabo naprave – računalnika, in je velikokrat že nameščena na napravo. V to kategorijo spadajo tako operacijski sistemi kot vgrajena (angl. *embedded*) programska oprema, katero najdemo, npr. v avtomobilih, navigacijskih napravah, merilnih napravah ipd.

Uporabniška programska oprema je tista, ki je namenjena izpolnitvi specifične uporabniške potrebe in je nameščena na računalnik ali drugo napravo z že nameščeno sistemsko programsko opremo.

1.1.1 Struktura produkta programske opreme

Fricker (2012) pravi, da obstajajo za področje produkta programske opreme tri temeljna področja interpretacije, ki so povezana s sestavo produkta programske opreme. To so razvojno-inženirska, poslovno-prodajna in pravna interpretacija.

1.1.1.1 Razvojno-inženirska interpretacija

Razvojno-inženirska interpretacija produkta programske opreme se usmerja na sam osnovni produkt, katerega sestavljajo programi, procedure, z njima povezana dokumentacija in nabor podatkov, kar skupaj sestavlja paket, osnovan za končnega uporabnika. Te komponente osnovnega produkta so po navedbi Sneed, Hasitschka in Teichmanna (2004, str. 11) rezultat:

- izhodiščnega načrtovanja,
- zbiranja zahtev,
- arhitekture in oblikovanja,
- kodiranja,
- konfiguracijskih podatkov,
- testnih planov in testnih primerov,
- uporabniške dokumentacije.

1.1.1.2 Poslovno-prodajna interpretacija

Poslovno-prodajna interpretacija produkta programske opreme, kot jo imenuje Fricker (2012), opredeljuje produkt kot zaključen nabor programskih komponent ali storitev. Te programske komponente temeljijo na programski opremi ter vključujejo podporni material in storitve, ki so izdani z namenom prodaje in distribucije na določenem trgu. Pri tem so

podporni materiali mišljeni v obliki marketinških brošur, spletnih strani, uporabniških navodil, izobraževalnih brošur.

1.1.1.3 Pravna interpretacija

Pravno interpretacijo produkta programske opreme Fricker (2012) opredeljuje kot nabor pravic za uporabo storitev in artefaktov produkta programske opreme. Pravna podlaga za uporabo produkta programske opreme je definirana v licenčnem sporazumu, ki je še posebej pomemben v primeru uporabe odprtokodnih rešitev v produktu.

1.1.2 Klasifikacija produktov programske opreme

S stališča namena in načina uporabe produkta programske opreme, kot tudi s stališča časa in energije, porabljene na strani stranke, za odločitev o nakupu, se ti produkti delijo na več skupin. Produkt programske opreme se lahko uporabi kot samostojna končna rešitev ali kot del končne rešitve drugega avtorja, katero ponuja na trgu kot končno uporabniško rešitev ali kot uporabo programske opreme kot storitve (angl. *Software as a Service*), ki je v porastu uporabe z razvojem računalništva v oblaku (angl. *Cloud computing*).

Fricker (2012) deli produkte programske opreme na:

- programsko opremo v paketu (angl. *packaged software*),
- vgrajeno programsko opremo (angl. *embedded software*),
- programsko opremo kot storitev.

Produkti znotraj vsake od teh skupin pa se notranje delijo še na (Fricker, 2012):

- produkte za končne uporabnike,
- industrijske produkte.

Produkti za končne uporabnike so namenjeni za osebno rabo končnih uporabnikov, ki pa jih je mogoče deliti še glede na vloženo energijo in trajanje odločitvenega procesa. Tako se ti delijo na produkte vsakodneвне rabe do takšnih, pri katerih potrošnik pred samim nakupom o tem bolj premisli in preveri alternative produktov.

Stranke industrijskih produktov so podjetja ali druge enote znotraj organizacije. Ti so lahko uporabljeni kot gradniki druge programske opreme ali kot orodje in infrastruktura za gradnjo in uporabo druge programske opreme kot uporabniške programske opreme za podporo izvajanja poslovnih procesov v podjetju.

1.1.3 Stopenjski model faz v življenjskem ciklu produkta programske opreme

Informacijska rešitev ali produkt programske opreme gre v svojem življenjskem ciklu skozi več stopenj. Vsaka stopnja ima svoje značilnosti z vidika funkcionalne ustreznosti

rešitve, obsega programskih napak ter možnosti večjih tehničnih in funkcionalnih predelav same rešitve.

V ta namen sta Rajlich in Bennett (2000) definirala stopenjski model s fazami v življenjskem ciklu produkta programske opreme, v katerem cikel delita v naslednje faze:

- Začetni razvoj – inženirji implementirajo prvo delujočo verzijo informacijske rešitve.
- Evolucija – inženirji razširjajo zmožnosti in funkcionalnosti informacijske rešitve, da ta ustreza zahtevam uporabnikov na več različnih načinov.
- Servisiranje – inženirji izvajajo manjše popravke informacijske rešitve ter enostavne funkcionalne spremembe.
- Faza zaključevanja – podjetje se odloči, da se servisiranje informacijske rešitve ne izvaja več, in poskuša iz rešitve še maksimizirati prihodke.
- Zapiranje – podjetje odstrani rešitev iz trga in usmeri uporabnike na uporabo nadomestne rešitve, če ta obstaja.

Vsaka od teh razvojnih faz ima svoje značilnosti v izvajanih aktivnostih, orodjih, možnostih obsežnejših predelav in poslovnih posledicah.

Rajlich in Bennett (2000) z razvojem tega modela poudarjata, da je vračanje informacijske rešitve, ko ta doseže značilnosti posamezne faze, v predhodno fazo zelo težko, če ne nemogoče.

1.1.4 Razlikovanje med produktom programske opreme in konkretno namestitveno instanco

Za boljšo predstavbo, kaj predstavlja produkt programske opreme, izpostavimo razliko med produktom in konkretno namestitveno instanco tega produkta. Kittlaus in Clough (2009, str. 8) opredelita to kot razlikovanje med tipom in instanco na primeru proizvodnje avtomobilov. Tip produkta predstavlja določen model avtomobila znotraj avtomobilske znamke, za katerega je značilno, da se proizvaja v različnih karoserijskih različicah in z različnimi motorji. Instanca produkta pa je konkretno vozilo s točno določeno barvo, karoserijsko različico in motorjem ter namenjeno točno določeni stranki. Nabor možnih instanc je v tem primeru omejeno število. Tako je manager produkta odgovoren za razvoj in proizvodnjo tipa vozila.

Analogijo iz proizvodnje avtomobilov Kittlaus in Clough (2009, str. 9) preneseta na produkt programske opreme, kjer se razvoj aplikacije izvaja za več različnih operacijskih sistemov, različnih proizvajalcev podatkovnih baz, jezikovnih variant ali lokalnih zakonskih predpisov. V primeru produkta programske opreme velikokrat vse lastnosti instance niso natančno definirane, saj se predvideva, da so pred samo produkcijsko uporabo potrebne še prilagoditve, ki so lahko izvedene s strani proizvajalca programske opreme, stranke same ali zunanega partnerskega podjetja.

V Tabeli 1 je na primeru informacijske rešitve AdInsure, katero bomo v drugem delu magistrskega dela podrobneje predstavili in zanjo iskali izboljšave v okviru managementa produkta programske opreme, prikazan tip produkta programske opreme.

Tabela 1: Tip programske opreme na primeru informacijske rešitve AdInsure

Operacijski sistem:	Windows Server 2008, Windows Server 2012
Jezik:	slovenščina, hrvaščina, srbsčina, makedonščina, ruščina, angleščina
Lokalni zakonski predpisi:	Slovenija, Hrvaška, Srbija, Makedonija, Rusija

Vir: Povzeto po H. Kittlausu in P. N. Cloughu, Software Product Management and Pricing: Key Success Factors for Software Organizations, 2009

V Tabeli 2 je za primer informacijske rešitve AdInsure prikazana še instanca, ki je namenjena točno določeni stranki. Vsaka stranka zahteva ustrezno prilagoditev na lokalne predpise zakonske in druge predpise, jezikovno prilagoditev ipd.

Tabela 2: Instanca programske opreme na primeru konkretne implementacije informacijske rešitve AdInsure za slovensko stranko

Operacijski sistem:	Windows Server 2008
Jezik:	slovenščina
Lokalni zakonski predpisi:	Slovenija

Vir: Povzeto po H. Kittlausu in P. N. Cloughu, Software Product Management and Pricing: Key Success Factors for Software Organizations, 2009

V primeru, kadar so za določeno instanco potrebne prilagoditve tudi v smislu spreminjanja programske kode, kar izvaja ali stranka sama ali zunanji partner, se ob izdaji instance s strani proizvajalca doda dogovor, ki opredeljuje garancijo proizvajalca samo za dele, ki so programsko nespremenjeni. Takšni primeri so izven dosega in vpliva produktnega managerja, zato zanje ni odgovoren.

1.2 Motivacija za vpeljavo koncepta managementa produkta programske opreme

»Zmaga ali poraz podjetja sta odvisna od produktnih managerjev«, povzema Ebert (2006) ugotovitve različnih avtorjev.

Ebert (2006) v analizi več raziskav ugotavlja, da se z bolj temeljitim razumevanjem in izvajanjem vloge produktnega managementa povečuje odstotek uspešnih produktov, hkrati pa tudi zadovoljstvo uporabnikov in njihovega vodstva. Ebert (2006) je v tej raziskavi, v

kateri je izvajal opazovanje vpeljave managementa produkta programske opreme v podjetju Alcatel, ugotovil, da je investicija v okrepitev vloge managerja produkta programske opreme dokaj nizka v primerjavi z dodano vrednostjo, ki jo ta lahko prinese, in sicer je lahko tudi preko 20-kratna. V triletnem opazovanju 178 projektov so bile ugotovljene prednosti vpeljave koncepta:

- čas vpeljave produkta na trg se skrajša,
- kvaliteta izdanega se izboljša,
- ujetje urnika oziroma plana je izboljšano.

Na podlagi dejavnikov, ki povzročijo ta pozitivni vpliv managementa produkta programske opreme, Ebert (2006) postavi sledeče smernice:

- poslovni cilji in odgovornost,
- obvladovanje zahtev,
- upravljanje tveganj in nejasnosti,
- vodenje in timsko delo.

Te ugotovitve so primerna motivacija za študij koncepta in njegovo ustrezno vpeljavo.

Glavni stroški, povezani z vpeljavo koncepta, v Ebertovi (2006) raziskavi nastanejo v:

- izobraževanju zaposlenih za koncept managementa produkta programske opreme,
- uvedbi koncepta,
- povečani komunikaciji med različnimi deležniki v konceptu produktnega managementa programske opreme.

1.2.1 Definicija managementa produkta programske opreme

Kittlaus in Clough (2009) v osnovi management produkta programske opreme definirata kot upravljanje in koordiniranje relevantnih aktivnosti na vseh področjih znotraj in izven podjetja z namenom dolgotrajne optimizacije uspeha produkta. Znotraj podjetja se koncept ukvarja predvsem z načrtovanjem, razvojem, proizvodnjo, trženjem, logistiko in z določanjem cene ter raziskavami trga, na katerem se produkt pojavlja.

Ebert (2006) definira management produkta programske opreme kot disciplino, ki ureja in kontrolira življenjski cikel produkta (rešitve ali storitve) od njegovega nastanka do dobave na trg ali stranki ter izvajanja podpore zanj, in sicer z namenom kreiranja največje možne vrednosti za podjetje. To vključuje vse ključne aktivnosti, povezane s produktom programske opreme, od strateških in prodajnih aktivnosti do razvojnih/implementacijskih aktivnosti in podpore strankam (Maglyas et al., 2011).

Vpeljava koncepta managementa produkta v razvoj programske opreme se od klasičnih serijskih proizvodnih produktov precej razlikuje. Določene aktivnosti, opredeljene v samem konceptu, so za področje razvoja produkta programske opreme bolj pomembne in ustrezno prilagojene, druge pa manj. Tako Kittlaus in Clough (2009) navajata kot primer manj pomembnih področij za produkt programske opreme proizvodnjo in logistiko. Bolj pomembno za področje programske opreme navajata področje zbiranja funkcionalnih in tehničnih zahtev. Ta mora omogočati visoko stopnjo prilagodljivosti v nenehnih spremembah, ki se na področju zahtev dogajajo skozi celotno življenjsko obdobje produkta.

1.2.2 Strategija podjetja v povezavi s produktom programske opreme

Uspešno podjetje ima jasno opredeljeni smisel in določeno pot, kateri želi slediti. Ta smisel lahko opišemo tudi kot vizija, katero podjetje zasleduje. McGrath (2001, str. 4) trdi, da je pomanjkanje vizije v podjetju velikokrat razlog za njegov neuspeh. Zato je pomembno, da ima podjetje jasno vizijo svojega poslovanja in ustrezno vizijo, v katero smer se bo področje, na katerem deluje, razvijalo.

Odgovornost za pripravo strateške vizije podjetja ima vodstvo. Vizija mora biti enostavna, jasna in nedvoumna, izražati mora jasen fokus in mora biti kompletna. Kot pravi McGrath (2001, str. 11–16), kompletna pomeni, da mora odgovarjati na vprašanja:

- Kam hočemo priti?
- Kako bomo tja prišli?
- Zakaj bomo zaradi tega uspešni?

Dobro pripravljena vizija je izhodišče za pripravo strategije. V podjetjih z različnimi področji delovanja razlikujejo Porter, Goold in Luchs (1996) dva nivoja strategije:

- strategijo poslovnih enot,
- strategijo podjetja kot celote.

Strategija poslovne enote opredeljuje, kako ustvariti konkurenčno prednost podjetja na vseh področjih, na katerih podjetje tekmuje za tržni delež in svoj uspeh.

Strategija podjetja kot celote se ukvarja z dvema glavnima vprašanjema: s kakšnimi dejavnostmi se podjetje ukvarja in na kakšen način managirati različna poslovna področja znotraj podjetja.

Z namenom ustvarjanja dobička in dolgoročne stabilnosti podjetja se mora v strategiji podjetja odražati zahteva po vzdržnosti poslovanja. V primeru, da je to podjetje, ki se ukvarja z razvojem informacijskih rešitev za končne stranke, se v strategiji podjetja med drugim kot vzdržnost poslovanja opredeli vzdržnost produkta programske opreme.

Vzdržnost produkta se zagotovi z ustreznim upravljanjem zahtev in aktivnostmi, povezanimi s produktom, ki so usmerjene izven okvirov kratkoročnih ciljev ali funkcionalnih zahtev določene stranke ali uporabnika informacijske rešitve. Uresničevanje strategije lahko podjetje doseže z uvedbo koncepta managementa produkta programske opreme. To pomeni, da mora institucionalizirati mesto managerja produkta, ki bo v okviru strategije izvajanja koncepta uresničeval strategijo podjetja.

1.3 Vloge v okviru managementa produkta programske opreme

Podjetja delujejo v okolju, ki pomembno vpliva na njegovo delovanje in z deležniki okolje neposredno ali posredno sodeluje.

Ideja managementa produkta programske opreme je, da je za produkt kot celoto odgovorna ena oseba, ki s pomočniki deluje v okviru svojega oddelka in ima ustrezna pooblastila za odločanje glede vseh aktivnosti, povezanih s produktom, z namenom maksimalne materialne učinkovitosti produkta.

1.3.1.1 Manager produkta

Ebert (2006) opiše vlogo managerja produkta kot »mini« izvršnega direktorja, ki na nivoju produkta:

- definira strategijo in podpira njeno operativno izvajanje z ustrezno kombinacijo projektov,
- preverja in ocenjuje izdane verzije produkta skozi prizmo prispevkov k celotnemu poslovnemu uspehu,
- s pomočjo procesov življenjskega cikla produkta izvaja reden pregled predhodnih predpostavk in sprejema nadaljnje odločitve.

Slika 1 prikazuje razliko v perspektivi pogleda na produkt med managerjem produkta programske opreme in posameznim projektom, ki se izvaja v življenjskem ciklu produkta.

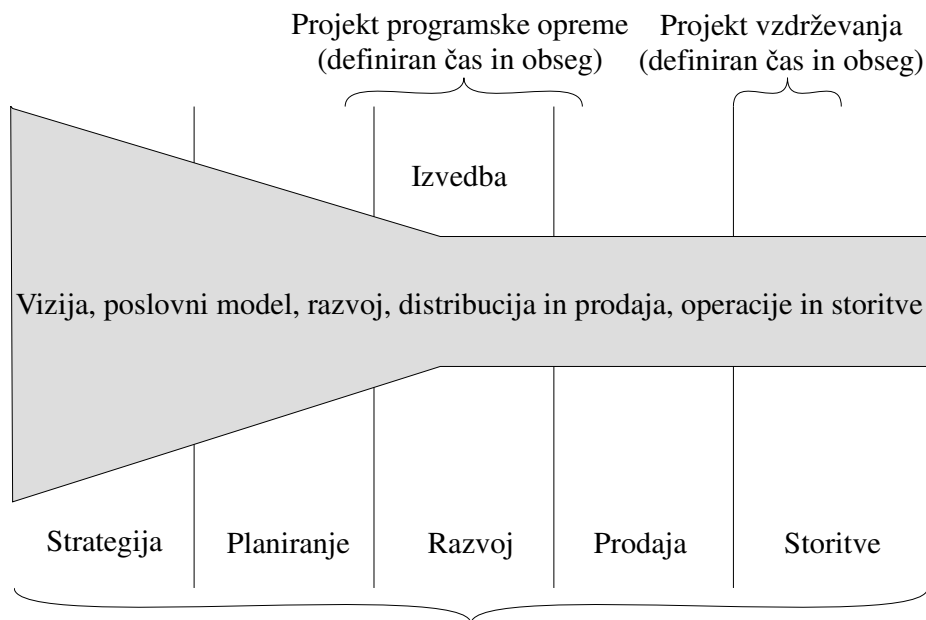
Gorchels (2006, str. 11) opiše managerja produkta kot osebo, ki nadzira vse vidike produkta z namenom ustvarjanja velikega zadovoljstva strank in hkrati dolgoročnega ohranjanja vrednosti produkta za podjetje.

Vloga managerja produkta je pomembna v tem, da povezuje strategijo podjetja s strategijo produkta in jo razvija, prilagaja produkt zahtevam trga ter koordinira aktivnosti na področju razvoja produkta, prodaje in trženja ter ustrezno skrbi za vse aktivnosti vzdrževanja in podpore, vezane na produkt.

Kot pravi Ebert (2006), managerjeva odgovornost ni omejena na posamezni projekt, ki poteka v okviru razvoja produkta programske opreme, saj je odgovoren za uspeh celotnega

portfelja projektov kot celote. Zaradi tega njegovo odločanje poteka na nivoju širšega pogleda, kjer definira, kaj in kako se bodo določene aktivnosti izvajale.

Slika 1: Management produkta in vloga managerja produkta



Vir: C. Ebert, *The Impacts of Software Product Management*, 2006

Ebert (2006) navaja zanimiv konkreten primer razlikovanja med projektnim vodjem in managerjem produkta. Projektni vodja se fokusira na uspeh konkretnega projekta in v primeru, da pride do konflikta med projekti zaradi delitve določenih virov, se projektni vodja načeloma ne ozira na ostale projekte, s katerimi si te vire deli. V takem primeru mora nastopiti manager produkta, ki sprejme odločitev glede virov, katere si delijo različni projekti v okviru istega produkta.

Drugi primer je, ki ga Ebert (2006) navaja, je razlikovanje med managerjem produkta in trženjem in prodajo ter projektnim vodjem v določanju tržnega plana za stranke. Cilj trženja in prodaje je primarno vedno prodati čim več, pogoji katerim mora podjetje ustreči, da je prodaja uspešno izvedena, so za sam oddelek prodaje manj pomembni. Tako prodaja hitro obljubi izvedbo zahtev, ki jih stranka poda, ne glede na realne možnosti izvedbe obljub. Na takšen način se količina zahtev za produkt hitro povečuje in na koncu ima to, brez širše slike in kontrole nad prodanim, negativne posledice za realizacijo. V veliko primerih je realizacija takšnih zahtev neuspešna. Manager produkta je tako na strani uspešnosti produkta in ne samo na strani prodaje. Ker je manager produkta odgovoren za produkt kot celoto, so te odločitve in obveze sprejete s širšim pogledom.

Fricker (2012) podaja opis del in nalog managerja produkta naslednje aktivnosti, katere izvajajo v sodelovanju in ob podpori različnih oddelkov v podjetju:

- razvoj strategije produkta v skladu s strategijo podjetja, katere cilj je doseči dolgotrajen poslovni uspeh produkta;
- planiranje tržnih aktivnosti in preverjanje uspešnosti v sodelovanju z oddelkom trženja;
- definiranje in planiranje obsega funkcionalnosti produkta in nadzor nad tem v sodelovanju z oddelkom razvoja;
- predstavlja produkt znotraj in izven podjetja in koordinira aktivnosti, povezane s produktom med oddelki prodaje ter podporo in vzdrževanjem.

Glavna orodja po Ebertu (2006), ki jih manager produkta programske opreme uporablja, so:

- dolgoročni plan produkta,
- funkcionalne in tehnične zahteve,
- vhodni in izhodni pregledi,
- poslovni primeri, ki jih produkt rešuje.

1.3.1.2 Višji management in vodstvo podjetja

Odgovornost višjega managementa in vodstva podjetja je vzpostavitev jasne vizije podjetja, iz katere vodstvo podjetja definirana tudi strategijo podjetja. Višji management je odgovoren za izvajanje aktivnosti s katerimi uresničuje strategijo podjetja.

Ebert (2006) navaja naloge managementa in vodstva podjetja:

- koordinacija portfelja produktov,
- definicija in priprava metrik uspešnosti,
- analiza rezultatov po definiranih metrikah,
- priprava enotne trženske strategije na nivoju podjetja,
- zagotavljanje virov skozi celoten življenjski cikel produkta.

Cusumano (2003) pravi, da mora vodstvo podjetja, ki se ukvarja tako s prodajo storitev kot z razvojem lastnih produktov programske opreme, dobro vedeti, kaj je njegov glavni poslovni model, saj so organizacijske zahteve za razvoj in prodajo produkta precej drugačne od organizacijskih zahtev prodaje storitev za stranke.

1.3.1.3 Oddelek raziskav

Naloga oddelka raziskav in inovacij je po Weerdu, Brinkkemperju, Nieuwenhuisu, Versendaalu in Bijlsmanu (2006):

- iskanje novih priložnosti za inovacije na produktu,
- iskanje načinov, kako vključiti izboljšave in nove funkcionalnosti v obstoječi produkt.

1.3.1.4 Oddelek trženja in prodaje

Manager produkta programske opreme je za oddelek prodaje in trženja notranji dobavitelj, ki skrbi za dobavo novih ali izboljšanih izdelkov. Oddelek prodaje in trženja v tem primeru predstavlja končno stranko produkta. Kotler (2012, str. 5) definira trženje ali marketing kot proces, s katerim podjetja ustvarjajo dodano vrednost za stranke, s tem da ustvarjajo in vzdržujejo močan stik s stranko z namenom povratnega ustvarjanja dodane vrednosti.

1.3.1.5 Razvojni oddelek

Fricker (2012) pravi, da v okviru priprave strategije produkta in planiranja razvoja produkta sodeluje manager produkta programske opreme z razvojnim oddelkom, s katerim išče priložnosti za inovacije na področju produkta in uporabljene tehnologije. Z njim sodeluje tekom celotne implementacijske faze.

1.3.1.6 Projektni vodja

Fricker (2012) opiše vlogo projektnega vodje kot osebo, ki je odgovorna za uspešno izvajanje projekta, ki vodi do uspešne izdaje produkta. Manager produkta sodeluje z več projektnimi vodji, pri katerih je v vlogi interne stranke, ki definira zahteve in prioritete za posamezno izdajo produkta.

1.3.1.7 Storitve in integracija

Konzultanti oddelka storitev so odgovorni za vpeljava produkta pri stranki. Tako prek stranke in njenih zahtev sporočajo informacije managerju produkta programske opreme.

1.4 Aktivnosti pri managementu produkta programske opreme

Vloga managementa produkta programske opreme je v temelju skrb za dobrine podjetja, v obliki produktov programske opreme in z njimi povezanega znanja, da ostajajo uporabne in zanimive za trg čim daljše časovno obdobje.

Kittlaus in Clough (2009, str. 40) prikažeta funkcijo managerja produkta programske opreme kot osebe, ki se primarno ukvarja ravno z vzdrževanjem vrednosti podjetja v obliki produkta in znanja, povezanega z njim.

Izvajanje aktivnosti v konceptu managementa produkta programske opreme so zelo raznolike, vse pa so usmerjene v smeri dolgoročnega vzdrževanja vrednosti produkta. To pomeni, da v tem okviru lahko določene kratkoročne aktivnosti kreirajo tudi negativno finančno sliko, vendar se te aktivnosti dolgoročno gledano poplačajo. Kot trdita Kittlaus in Clough (2009, str 41), se tako ustvari konflikt med dolgoročnimi cilji managementa produkta in zahtevo po merljivi uspešnosti v periodah, ki so definirane z nivoja višjega

managementa. Konflikt pa se nadaljuje v kombinaciji razvoja produkta in uspešnih zaključkov njegovih razvojnih in ostalih aktivnosti. Te se večinoma ne ujemajo s periodami, določenimi s strani višjega managementa, kateri želi v svojem ciklu preverjati uspešnost delovanja managementa produkta in takrat sprejemati odločitve za korekcije. Tako Kittlaus in Clough (2009, str 43) kot Ebert (2006) in Weerd et al. (2006) ugotavljajo, da delo managerja produkta ni enostavno merljivo, predvsem ne iz perspektive kratkoročnih ciljev.

Opazimo lahko, da se aktivnosti managerja produkta programske opreme včasih precej razlikujejo od projektno in procesno usmerjenih aktivnosti, pri katerih se lahko rezultati oz. prispevki posameznika h končnemu rezultatu izvajanega projekta merijo na določenih krajših časovnih obdobjih oziroma mejnikih. Večino aktivnosti managerja produkta programske opreme je dolgoročnih in ponavljajočih, od katerih so bile določene v preteklosti bolj raziskane kot druge.

Z namenom boljšega strukturiranja aktivnosti in funkcij, ki jih opravlja manager produkta programske opreme, so Weerd et al. (2006) razvili referenčni model z opredeljenimi aktivnostmi in rezultati, ki se od managerja produkta programske opreme pričakujejo. Ta model v glavnem pokriva, kot trdita Kittlaus in Clough (2009, str 44), del aktivnosti, ki jih je lažje povezati v procesno obliko. Kittlaus in Clough (2009, str 44–115) sta na podlagi obstoječega razvila nov model, razvit iz izhodišča, da je glavna naloga managerja zagotavljanje dolgoročno vzdržne vrednosti produkta za podjetje in njegov ekonomski uspeh. Tako njun model vključuje več poudarka na ekonomskem aspektu koncepta managementa produkta programske opreme (Kittlaus & Clough, 2009, str 45).

V nadaljevanju predstavljamo oba modela s poudarkom na slednjem, saj je po mojem mnenju bolj dodelan in bolje predstavlja nabor aktivnosti managerja produkta programske opreme in njihove korelacije z nalogami različnih oddelkov v podjetju.

1.4.1 Referenčni model managementa produkta programske opreme po Weerdu et al.

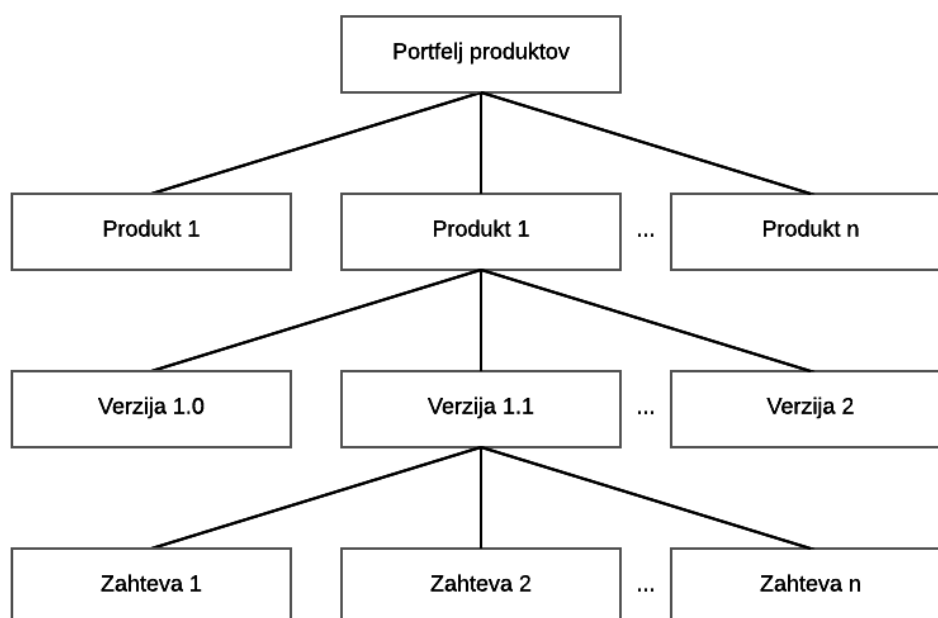
Weerd et al. so z namenom združevanja raziskanega na področju managementa produkta programske opreme definirali model, v katerem so opredelili aktivnosti in korelacije med njimi, ki se izvajajo znotraj managementa produkta programske opreme. Model naj bi služil višjemu managementu za boljšo predstavbo o aktivnostih, ki so potrebne za izvajanje koncepta, in managerjem produktov programske opreme kot ogrodje za njihove aktivnosti. Kittlaus in Clough (2009, str 45) sta prav tako razvila ogrodje, katerega aktivnosti podrobneje opišeta. Na podlagi teh dveh ogrodij bomo opredelili aktivnosti, ki se izvajajo v okviru implementacije koncepta managementa produkta programske opreme.

1.4.1.1 Entitete, na katere so vezane aktivnosti

Weerd et al. (2006) opredelijo v razvitem ogrodju 4 entitete, ki imajo glavno vlogo v konceptu, in okrog katerih se izvajajo različne aktivnosti, ki te entitete povezujejo med seboj. Ogrodje temelji na dejstvu, da je dobro organiziran management produktov dejansko izvajanje aktivnosti, povezanih s funkcionalnimi in drugimi zahtevami, produktom samim ter izdajami oziroma verzijami produkta. Entitete so med seboj hierarhično povezane, kot je prikazano na Sliki 2.

Weerd et al. (2006) v svojem ogrodju delijo aktivnosti v okviru managementa produkta programske opreme v 4 skupine. Središče vsake od teh skupin je ena izmed entitet, predstavljena na Sliki 2. Diagram z razdelitvijo teh aktivnosti je predstavljen na Sliki 3.

Slika 2: Hierarhija entitet managementa produkta



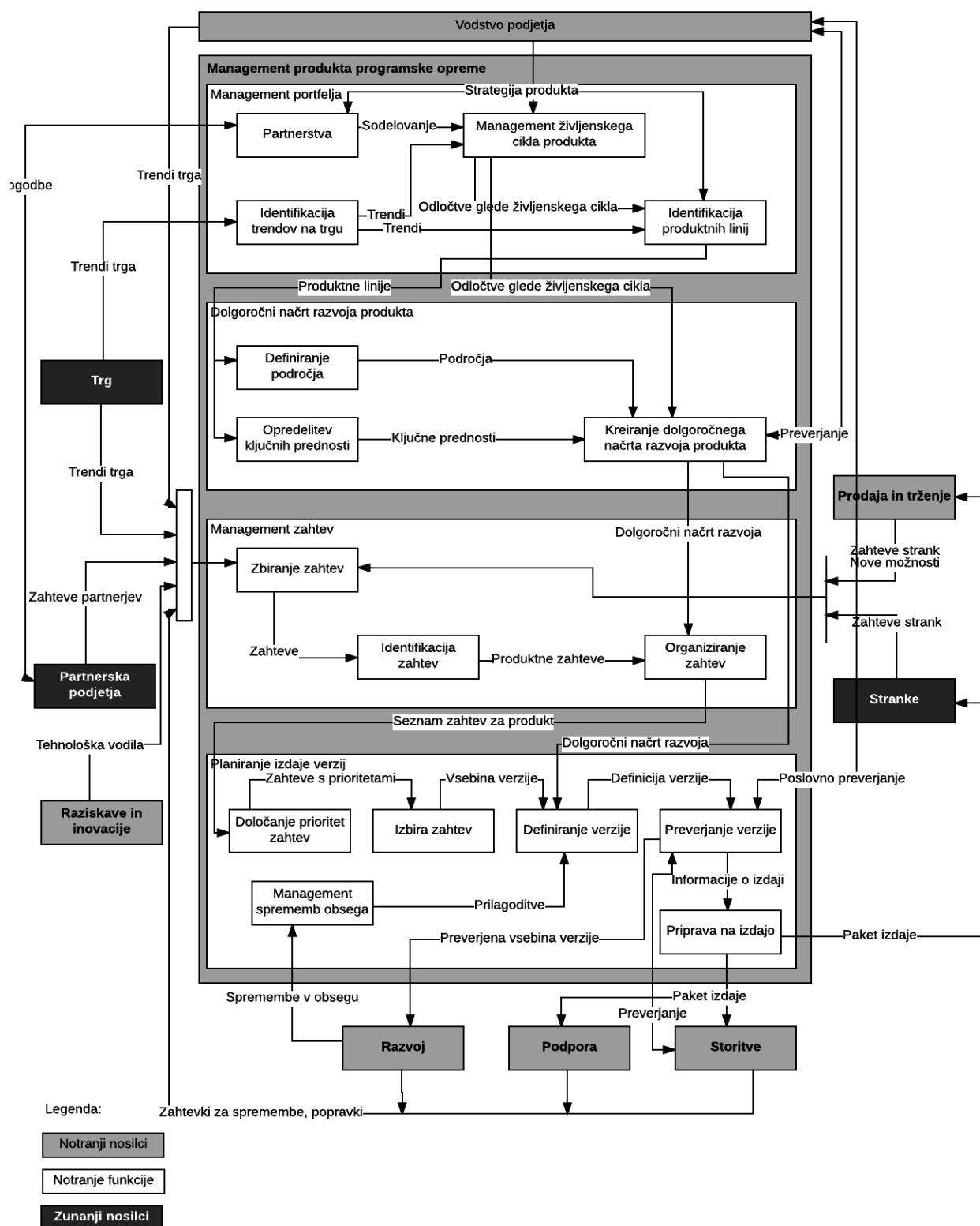
Vir: I. Weerd, S. Brinkkemper, R. Nieuwenhuis, J. Versendaal in L. Bijlsma, A Reference Framework for Software Product Management, 2006

V nadaljevanju predstavimo vsako od teh skupin posebej z opisom aktivnosti, ki se izvajajo v povezavi s posameznimi entitetami.

1.4.1.2 Management portfelja

Aktivnosti v okviru managementa portfelja, kot jih defenirajo Weerd et al. (2006), vključujejo odločitve, povezane z obstoječimi produkti in uvedbo novih produktov v portfelj podjetja. V tem okviru se izvajajo aktivnosti analize trga, v katerega je produkt vključen, aktivnosti odločanja o življenjskem ciklu produkta in vzpostavitev partnerstev.

Slika 3: Referenčni model za management produkta programske opreme po Weerd et al.



Vir: I. van de Weerd, S. Brinkkemper, R. Nieuwenhuis, J. Versendaal in L. Bijlsma. A Reference Framework for Software Product Management, 2006.

Kot vidimo na Sliki 3, se v okviru managementa portfelja izvajajo 4 glavne aktivnosti:

- delo s partnerji in priprava pogodb o sodelovanju,
- management ciklov produkta programske opreme,
- identifikacija trendov trga, na katerem se podjetje s produktom bori za svoj tržni delež,
- identifikacija možnosti vpeljave produktnih linij.

Management ciklov produkta programske opreme Stark (2011, str. 1) definira kot dejavnost upravljanja produktov podjetja skozi ves življenjski cikel produktov, to je od zasnove prve ideje za produkt do njegove upokojitve, ko ni več uporabljen.

Identifikacija možnosti vpeljave produktnih linij je pomembna za iskanje sinergij med obstoječimi produkti podjetja ali možnostjo vpeljave produktne linije, kjer je izhodišče že pripravljen produkt. Kranjc (2001, str. 17) pravi, da je to aktivnost, s katero želimo doseči predvsem ekonomičnost razvoja programske opreme.

1.4.1.3 Dolgoročni načrt razvoja produkta

Po modelu je entiteta okrog katere se definirajo aktivnosti, ki so v domeni managerja produkta programske opreme in katere direktno izvaja on sam ali pa orkestrira aktivnosti drugih oddelkov, produkt. V okviru produkta kot takega so opredeljene aktivnosti:

- definiranje področja,
- opredelitev ključnih prednosti,
- kreiranje dolgoročnega načrta razvoja produkta.

Priprava dolgoročnega načrta razvoja produkta (angl. *roadmapping*) je kompleksen proces analiziranja uporabe znanstvenih in tehnoloških virov, elementov in njihovih povezav za daljše časovno obdobje. Kompleksnost tem aktivnostim daje dejstvo, da je treba upoštevati odvisnosti na obstoječe interne produkte in rešitve, kot tudi rešitve, s katerimi se bo produkt povezoval. Prav tako je treba upoštevati spremembe v tehnologiji.

Kostoff in Schaller (2001) dolgoročni načrt opisujeta kot proces, ki določi pot identifikacije, evalvacije in izbire strateških alternativ, ki so lahko uporabljene za doseg želenega cilja skozi življenjski cikel produkta.

Weerd et al. (2006) povzamejo po Regnelli in Brinkkemperju (2005, str. 301) definicijo načrta produkta kot dokumenta, ki vsebuje predviden načrt izdajanja verzij produkta za obdobje od 3 do 5 let. Dokument je zapisan v okviru pričakovanj in trenutno razpoložljivih sredstev.

Po mnenju Kittlausa in Clougha (2009, str.77) naj bi dolgoročni načrt razvoja produkta prikazoval pogled na razvoj produkta za strateško obdobje do 5 let. Ta načrt vsebuje plan

novih verzij in njihove časovne opredelitve ter ostale pomembne točke, povezane s produktom, ki se jih v tem obdobju pričakuje. Njegov namen je prikaz smeri razvoja produkta interno in eksterno. Njegov interni namen je prikaz korelacij med planom produkta in finančnimi napovedmi. Poleg tega je njegov interni namen pokazati perspektivo novih kariernih priložnosti oseb, ki so trenutno vključene v aktivnosti na razvoju produkta. Eksterni namen dolgoročnega načrta razvoja produkta pa je prikaz sposobnosti izvedbe realizacije produkta, kar večje potencialne stranke pogosto preverjajo pred odločitvijo o nakupu.

V definiranem ogrodju Weerda et al. (2006) so vhodne informacije v dolgoročnem načrtovanju pridobljene iz aktivnosti managementa portfelja, s katerimi se identificirajo glavne prednosti in lastnosti produkta ter motivi za izbiro poti življenjskega cikla produkta.

1.4.1.4 Management zahtev

V managementu zahtev je usmeritev na eniteto posamezne zahteve, ki je v določenem vsebinskem in tehničnem kontekstu združena v verzijo. V okviru tega potekajo po modelu Weerda et al. naslednji sklopi aktivnosti (Weerd et al., 2006):

- zbiranje zahtev,
- identifikacija zahtev,
- organiziranje zahtev.

Kot vire zahtev Weerd et al. (2006) opredelijo naslednje:

- stranke in uporabniki,
- prodaja in trženje,
- razvoj,
- oddelek podpore,
- višji management podjetja.

Poleg teh Kittlaus in Clough (2009, str. 80–87) dodajata še:

- repozicioniranje produkta,
- raziskave trga,
- partnerji, ki produkt prodajajo in integrirajo z zunanjimi sistemi,
- neodvisni konzultanti.

Odločitev glede uravnoveženja zahtev iz različnih virov Kittlaus in Clough (2009, str. 86) opišeta kot eno primarnih nalog managerja produkta, pri kateri se pričakuje veliko konfliktov pri izbiri in določanju prioritet zahtev.

V okviru zbiranja zahtev Regnell in Brinkkemper (2005) povzemata ugotovitve več avtorjev, in sicer da se management zahtev za produkt razlikuje od managementa zahtev za informacijsko rešitev po naročilu. Glavna razlika med njima, ki jo izpostavita, je namreč ta, da je pri razvoju programskih produktov cilj usmerjen v pravočasno lansiranje produkta na trg, medtem ko je v primeru informacijske rešitve po naročilu fokus usmerjen v izvedbo zapisanih funkcionalnih zahtev in zahtev pogodbe.

Kittlaus in Clough (2009, str. 79) menita, da je ena glavnih nalog managerja produkta programske opreme delo na sprejemanju, evalvaciji in določanju prioritet zahtev. Pri tem navajata druge vire, ki kažejo na korelacijo med neuspešnimi projekti in pomanjkanjem pravih zahtev ali njihovim neustreznim managiranjem. Med izvajane aktivnosti v okviru managementa zahtev navajajo Weerd et al. (2006) aktivnosti, kot so:

- zbiranje zahtev,
- modeliranje in analiziranje zahtev,
- določanje prioritet zahtev,
- komuniciranje zahtev,
- poenotenje posameznih zahtev,
- delo na razvoju zahtev.

Schienmann (v Kittlaus & Clough, 2009, str. 79) opredeli 3 dimenzije managementa zahtev, ki so:

- management zahtev stranke,
- management zahtev projekta,
- management zahtev produkta.

Manager produkta programske opreme se primarno ukvarja s produktno dimenzijo managementa zahtev. Zahteve za produkt so tipično pridobljene preko projekta zbiranja zahtev, ki predstavlja projektno dimenzijo.

Schienmann (v Kittlaus & Clough, 2009, str. 87) opredeljuje perspektivo pogleda na management zahtev tudi prek različnih aktivnosti in procesov, ki se odvijajo tekom zbiranja zahtev s pomočjo:

- orientiranosti zahtev glede na stranko, ki prek produkta pričakuje rešitev določenih svojih problemov,
- orientiranosti zahtev glede na produkt, ki rešuje probleme in vsebuje rešitve zahtev zbranih iz različnih virov,
- orientiranosti zahtev glede na projekt, v okviru katerega so izvedene verzije produkta iz dobro definiranih zahtev in implementirane v ustreznih časovnih okvirih.

1.4.1.5 Planiranje verzij

Produkt je z izdajo verzije na voljo uporabnikom. Fokus je v tem primeru na verziji produkta in v okviru te skupine sklopov aktivnosti planiranja verzij Weerd et al. (2006) navajajo kot glavne aktivnosti:

- določanje prioritet zahtev,
- izbiranje zahtev,
- definiranje verzije, v katero bo zahteva vključena,
- preverjanje verzije in vključenosti in ustreznosti implementacije zahteve,
- management sprememb obsega,
- priprava na izdajo.

V ogrodju navajajo kot začetno aktivnost določanje prioritet zahtev produkta, kar ni naloga samo managerja produkta, ampak pri tem sodelujejo vsi deležniki. V glavnem obsegu so to tisti, ki so v okviru aktivnosti managementa zahtev predstavljali vire teh zahtev.

Po odločitvi o prioritetah se določijo zahteve za produkt, ki bodo vključene v naslednjo verzijo produkta, pri kateri se upošteva vire, ki so na razpolago za pripravo in izdajo verzije.

Ob določitvi zahtev za naslednjo verzijo se pripravi dokument o verziji, katerega potrdijo različni deležniki in tudi višji management oziroma vodstvo podjetja.

1.4.2 Model managementa produkta programske opreme po Kittlausu in Cloughu

Izhodišče modela funkcij in aktivnosti znotraj funkcij, ki sta ga razvila Kittlaus in Clough (2009, str 43–115), je dolgoročen ekonomski uspeh produkta, zato ekonomski aspekti koncepta v tem modelu odražajo pomembnejšo vlogo.

V Tabeli 3 je prikazan model, v katerem so v prvi vrstici opredeljene funkcije, v okviru katerih se delo managerja produkta programske opreme vrti. V stolpcih pod posamezno funkcijo so opisane aktivnosti, ki se v okviru te funkcije izvajajo. Aktivnosti so tako razdeljene na temeljne aktivnosti managerja in aktivnosti, dodatne opsijske aktivnosti, povezane z določanjem cenovne politike produkta, ter aktivnosti, pri katerih manager produkta programske opreme sodeluje in/ali jih orkestrira. Prvi stolpec opredeljuje razlikovanje med aktivnostmi, povezanimi na nivoju produkta ali produktne linije oziroma na nivoju podjetja.

V nadaljevanju podajamo podrobnejše opise vsake od aktivnosti, zapisane v modelu.

Tabela 3: Model managementa produkta programske opreme po Kittlausu in Cloughu

	Analiza trga	Produktna analiza	Strategija produkta	Planiranje produkta	Razvoj	Trženje	Prodaja in distribucija	Podpora in storitve
Ponavadi na nivoju podjetja			Management portfelja	Dodeljevanje virov	Dodeljevanje virov	Tržni strateški plan	Plan prodaje in distribucije	
Nivo produkta ali družine produktov	Raziskava trga Določanje velikosti trga Ocena tehnološke Analiza konkurenčnosti	Učinkovitost produkta Zadovoljstvo strank Analiza zmag in porazov Priložnosti	<u>Pozicioniranje</u> Model distribucije Model definiranja cen Definiranje cen Poslovni primer Izdelati samostojno ali kupiti Pravni pogoji Zaščita intelektualne lastnine	Dolgoročni načrt razvoja produkta Plan verzije Management zahtev Funkcionalna specifikacija	Projektni plan Tehnična specifikacija Management projektnih zahtev Implementacija Kontrola kvaliteta Tehnologija Inovacije	Plan izdaje Analiza strank Management partnerjev Operativno trženje Materiali	Priprava prodajnih kanalov Operativna prodaja Operativna distribucija Materiali	Podpora strankam Tehnična podpora Tržna podpora Podpora prodaji Priprava storitev Provizioniranje operativnih storitev

Legenda:

A – temeljne aktivnosti managementa produkta programske opreme,

B – temeljne aktivnosti določanja cen,

C – aktivnosti, v katerih manager produkta programske opreme sodeluje ali jih orkestrira

Vir: H. Kittlaus in P.N. Clough, *Software Product Management and Pricing : Key Success Factors for Software Organizations*, 2009, str. 46

1.4.2.1 Analiza trga

Manager produkta programske opreme mora neprestano vrednotiti stanje na trgu, na katerem je prisoten njegov produkt, in spremljati razvoj trga. Pri tem si lahko pomaga z več različnimi viri (Kittlaus & Clough, 2009, str. 47):

- sodelavci iz oddelkov trženja, prodaje, podpore strankam, storitev in razvoja,
- neposrednimi stiki s člani produktnega ekosistema in ostalimi deležniki na trgu,
- notranje raziskave trga,
- zunanje organizacije za raziskovanje trga.

Izkušen manager produkta programske opreme vzdržuje notranje in zunanje povezave z namenom dobrega pregleda stanja na trgu ter smeri njegovega razvoja. Pri tem mu lahko pomagajo zunanje organizacije, katerih primarna poslovna dejavnost je raziskovanje trga. Sicer se vse trudijo izpostavljati svojo neodvisnost od posameznih igralcev na trgu, vendar so za pridobivanje potrebnih informacij prisiljene v sodelovanje s ponudniki programske opreme, posledično pa kaj kmalu lahko pride do pristranskosti in naklonjenosti posameznemu igralcu na trgu oziroma njegovi rešitvi. Bolj znane organizacije, ki izvajajo in za plačilo ponujajo rezultate raziskav trga, so Gartner, IDC, Forrester Research (Kittlaus & Clough, 2009, str. 47–49). Specializirana raziskovalna institucija tega tipa na področju finančnega in zavarovalniškega trga pa je Celent.

1.4.2.2 Produktna analiza

Ob lansiranju produkta na trg mora manager produkta programske opreme z ekonomske perspektive spremljati odzive na trgu in te dobro poznati. Kot trdita Kittlaus in Clough (2009, str. 50), mora pri tem upoštevati eksplicitne metrike, ki so predvsem finančne narave, in mehkejške metrike, ki velikokrat prej pokažejo na probleme v zvezi s produktom.

Eksplicitne metrike so:

- prihodek – na izdano verzijo ali časovno obdobje,
- stroški – razvoja, vzdrževanja, podpore, trženja, prodaje, na izdano verzijo ali časovno obdobje,
- dobiček – na izdano verzijo ali časovno obdobje,
- število licenc – prodanih ali nameščenih na verzijo ali časovno obdobje ali stranko,
- število napak – na izdano verzijo ali časovno obdobje.

Informacije za mehkejške metrike pridobiva manager produkta programske opreme iz različnih virov, kot so:

- zadovoljstvo strank,
- povratne informacije iz analiz trga,

- povratne informacije od prodaje,
- povratne informacije iz oddelka podpore in storitev.

Na podlagi rezultatov teh metrik manager produkta programske opreme poišče korektivne ukrepe. V primeru, da so rezultati pod planiranimi v katerem koli pogledu, mora poiskati pravi naslov za izvedbo korektivnih ukrepov in njihovo izvedbo preverjati. V primeru pozitivnih informacij, rezultati nad predvidenimi plani, je prav tako dolžan preveriti morebitne potrebe po korektivnih ukrepih. Pozitiven rezultat lahko namreč kasneje privede tudi do negativnih odzivov. Primer tega je prodaja nad planom – ob izvajanju podpore ali storitev, povezane s produktom, je ta podhranjena. V takem primeru je treba ekipo podpore in storitev ustrezno okrepiti.

1.4.2.3 Strategija produkta

Investicija v produkte programske opreme v večini primerov ni majhna. Prav tako ti niso razviti za uporabo na kratko časovno obdobje, npr. teden ali mesec. Zaradi tega je treba pri načrtovanju produkta in kasneje velikokrat sprejemati dolgoročne odločitve, ki imajo tudi velike finančne posledice.

Z jasno strategijo podjetja in strategijo produkta, ki upošteva strategijo podjetja, je produkt lahko uspešen. Manager produkta programske opreme je odgovoren za ustrezno definiranje strategije produkta, ki opredeljuje cilje in njegove načine doseganja za obdobje od 2 do 5 let. Strategija mora vključevati odgovore glede ciljev in načinov doseganja ciljev na naslednjih področjih produkta (Kittlaus & Clough, 2009, str. 53–54):

- funkcionalni obseg produkta ,
- ciljni trgi in možni segmenti,
- način distribucije produkta,
- pozicioniranje produkta,
- poslovna pričakovanja, kot je razvoj trga, razvoj tržnega deleža, prodaja,
- plan obsega investicije in plan porabljenih virov,
- dolgoročni načrt razvoja produkta.

Področja, vključena v strategijo produkta, so med seboj tesno povezana. Vsako odstopanje med pričakovanimi viri in viri, ki so na voljo na enem od teh področij, zahteva prilagoditve tudi na drugih področjih. Tako na primer, če se v časovnem obdobju pripravljene strategije pričakuje vstop konkurenčnih produktov na trg, je, ali treba povečati vire za hitrejši vstop na trg ali določiti obseg funkcionalnosti, s katerimi bo mogoče produkt drugače pozicionirati. V pripravi strategije produkta je ob načrtovanju virov nujno upoštevati usklajenost na nivoju podjetja, enako velja tudi za obseg funkcionalnosti, ki si jih lahko deli več produktov v produktni liniji.

Naloga vodstva je priprava vizije in strategije podjetja, kar je treba upoštevati tudi ob pripravi strategije produkta. Manager produkta programske opreme je odgovoren za pripravo strategije produkta. Ob pripravi strategije, ki vsebuje trenutno stanje in pričakovanja razvoja v prihodnje, se mora v primeru ugotovitve velikih nasprotij med strategijo podjetja in strategijo produkta, ki jo pripravlja, kritično vprašati, ali je strategija podjetja še ustrezna glede na produkt, za katerega skrbi. V takšnih primerih lahko manager produkta programske opreme sproži preverjanje ustreznosti strategije podjetja.

Možnost vpliva na spremembo strategije podjetja je tesno povezano z njegovim ugledom in argumenti na nivoju celotnega podjetja. Zaradi tega je pomembno, da vodstvo podjetja jasno izpostavi teže funkcije managerja produkta programske opreme, saj so s tem zahteve s strani managerja produkta vzete bolj resno in imajo večjo težo.

V okviru izvajanja funkcije strategije produkta Kittlaus in Clough (2009, str. 54–76) izpostavljata področja, na katera so v tej funkciji opredeljene tudi aktivnosti. Področja poleg že omenjene vizije in strategije podjetja zajemajo še:

- management portfelja produktov,
- pozicioniranje produkta na trgu,
- poslovni pogled na produkt,
- način distribucije produkta do končnih strank in model za določanje cene,
- pravni pogoji prodaje in uporabe produkta,
- zaščita intelektualne lastnine.

Za management portfelja pravita Kittlaus in Clough (2009, str. 54–59), da naj se izvaja na nivoju vodstva podjetja. Investicija v produkt je za podjetje naložba, ki se obrestuje šele po uspešni realizaciji in prodaji produkta. Zaradi tega se portfelj produktov, katerih lastnik je podjetje, lahko obravnava kot portfelj investicij, za katere mora vodstvo podjetja dobro skrbeti. Management portfelja tako pomeni, da se za vsak produkt posebej ocenjuje in spremlja velikost in rast ciljnega trga za produkt, potrebno investicijo ter tveganja, povezana s produktom na tem trgu.

Pozicioniranje produkta na trgu zahteva odgovore na vprašanja, povezana s ciljnim trgom, katerega del je vprašanje glede dodane vrednosti, ki jo produkt na ta trg prinaša.

Poslovni pogled na produkt pojasnjuje poslovni primer, ki je primerjava stroškov, povezanih s produktom ter ekonomskimi učinki ali vrednostjo, ki jo produkt prinaša. Kittlaus in Clough (2009, str. 62) pravita, da je poslovni primer v določenih primerih treba določiti za vsako večjo funkcionalnost ali vsaj za vsako izdano verzijo. Razmerje med investicijo v novo verzijo in prihodki ali drugimi prednostmi, ki jih prinaša, mora biti v prid tem pridobljenim vrednostim.

Način distribucije produkta opredeljuje način, po katerem bodo končni uporabniki funkcionalnosti produkta uporabljali. Tako je bolj tradicionalen način nakup namestitvenega paketa na pomnilnem mediju ali po internetni distribuciji, kjer končna stranka po plačilu stroškov licence dobi pravice uporabe, ali pa v obdobju zadnjih nekaj let vse bolj razširjen model programske opreme kot storitve. Na osnovi načina distribucije je določen tudi model za določanje cene. Ta definira ceno in obseg vanjo vključenih storitev.

Pravni pogoji opredeljujejo pravila, pod katerimi je produkt pravno veljavno uporabljen s strani končnih uporabnikov. V večini je to po licenčni pogodbi ali v primeru programske opreme kot storitve, storitvene pogodbe, v kateri so opredeljeni pogoji, pod katerimi končni uporabnik produkt uporablja ali ga spreminja in takega naprej distribuira.

Zaščita intelektualne lastnine se razume kot način zaščite inovacij, vključene v produkt programske opreme, ki za podjetje predstavlja konkurenčno prednost. Primer takšne zaščite je prijava patentov, s katerimi se oteži uporabo teh inovacij v konkurenčnih produktih. Kittlaus in Clough (2009, str. 73) pravita, da na področju programske opreme to vrsto zaščite uporabljajo predvsem večja podjetja. Prijava namreč zahteva obsežne opise rešitve, kar velikokrat pomeni opis rešitve, ki jo konkurenca z legalnimi spremembami uporabi in tako lažje ponudi alternativno rešitev patentirane inovacije.

1.4.2.4 Planiranje produkta

Planiranje produkta programske opreme prištevamo med temeljne aktivnosti v okviru razvitega modela. Manager produkta programske opreme v okviru te funkcije pripravlja plan produkta, katerega uporablja in promovira na nivoju podjetja. Ta plan vsebuje plan trženja in prodaje ter pričakovan proračun ter vire in je osnova oziroma predpogoj za izvedbo planiranja aktivnosti na nivoju podjetja.

Planiranje na nivoju podjetja je dejansko iskanje kompromisa med parcialnimi interesi posameznih managerjev produktov ter ostalih interesov. Po navadi je izvedba planiranja na nivoju podjetja kombinacija pristopa od zgoraj navzdol ter od spodaj navzgor. Managerji produktov in ostali udeleženci planiranja pripravijo vsak za svoj del, produkt ali projekt, plan od spodaj navzgor. Vodstvo podjetja v okviru razpoložljivih finančnih in drugih virov te plane zmanjša oziroma jih omeji. Ker pa se v primeru zmanjšanja razpoložljivih virov plan na nivoju produkta do določene mere podre, ga je treba prilagoditi in ponovno preveriti ustreznost s plani na nivoju podjetja. Tako se ta cikel ponavlja, dokler se ne doseže konsenza v smislu najboljšega za podjetje kot celoto, ki bi moral biti cilj vseh udeleženi.

Kittlaus in Clough (2009, str. 76) ugotavljata, da referenčni model po Weerdu et al., opisanem v 1.4.1, v glavnem vključuje aktivnosti, ki spadajo v funkcijo planiranja produkta, katerega glavni elementi so priprava dolgoročnega načrta razvoja produkta, planiranje izdaje verzij in urejanje zahtev.

Aktivnosti v okviru funkcije planiranja produkta so temeljne aktivnosti, katerih izvajalec je, razen dodeljevanja virov, primarno manager produkta programske opreme. Te aktivnosti so:

- priprava in preverjanje izvajanja dolgoročnega načrta razvoja produkta,
- priprava plana verzije,
- management zahtev,
- funkcionalna specifikacija.

Dolgoročni načrt razvoja produkta programske opreme je opredeljen že v poglavju 1.4.1.3, saj ga predvideva tudi referenčni model po Weerdu et al opisan v 1.4.1.

Plan verzije mora odgovarjati na dve glavni vprašanji Kaj in Kdaj (Kittlaus & Clough, 2009, str. 76). Ob pripravi plana je treba upoštevati faktorje:

- ocene časa, potrebnega za razvoj in pripravo verzij,
- trženjska področja, povezana z verzijo in njeno izdajo,
- odvisnosti med zahtevami verzije in drugimi produkti.

Specifikacije in management zahtev spadajo po mnenju Kittlausa in Clougha (2009, str. 79) med pomembnejše naloge managerja produkta programske opreme. Management zahtev smo podrobneje opisali v 1.4.1.4, v okviru predstavitve referenčnega modela po Weerdu et. al.

1.4.2.5 Razvoj

Kittlaus in Clough (2009, str. 94) predlagata, da se managementu produkta programske opreme v okviru podjetja določi vlogo internega naročnika za oddelek razvoja. To vključuje tako določanje finančnih sredstev, ki so na voljo za razvoj, kot vsebinske zahteve ter prevzemanje oziroma potrjevanje izdelka. Finančna sredstva, ki so na voljo, morajo biti prav tako kot definiranje zahtev in planiranje verzij zagotovljena v okviru funkcije planiranja produkta.

Potrjevanje pravilnosti implementiranih zahtev in testiranje je izvedeno v okviru razvojnega oddelka. Rezultati testiranja so osnova za prevzem produkta, katerega mora potrditi manager produkta programske opreme.

Ločitev odgovornosti v smislu internega naročnika med managerjem produkta in razvojnim oddelkom zahteva veliko napora. Razvojni oddelek na ta nadzor po navadi gleda kot na nepotrebno kontrolo, poslovni del pa velikokrat prevzema del odgovornosti za rezultate razvojnega oddelka. Kittlaus in Clough (2009, str. 94) trdita, da je ta delitev potrebna z namenom bolj eksplicitne delitve vlog in odgovornosti, ki na koncu prispevajo k boljšemu sodelovanju.

V času razvoja produkta se vedno dogajajo spremembe zahtev. Za razvojni oddelek bi bilo sicer idealno, da se spremembe zahtev tekom razvoja ne dogajajo, vendar se je v praksi temu nemogoče izogniti. Razvojni oddelek mora na to računati in uporabljati agilne pristope projektnega vodenja, ki to predvidevajo. Za managerja produkta programske opreme to pomeni bolj zahtevno spremljanje stanja razvoja, saj se s spremembo zahtev spremenijo tudi plani, kako bo končni produkt izgledal in katere funkcionalnosti bo vseboval.

1.4.2.6 Trženje

V manjših podjetjih, kjer ne obstaja ločen oddelek za trženje produktov, prevzame to vlogo za svoj produkt manager produkta programske opreme. Glavna naloga managerja produkta programske opreme v tej funkciji je zagotavljanje, da se produkt na ciljnem trgu pojavi ob pravem trenutku, s pravimi funkcionalnostmi in ustrezno ceno (Kittlaus & Clough, 2009, str. 97). V okviru te funkcije je treba zagotoviti, da je produkt ustrezno zastopan in upoštevan v prodajni in tržni strategiji podjetja ter da je ustrezno vključen ter podprt v tržne poti s podporo vodstva podjetja. Strategija prodaje in trženja produkta mora namreč biti del strategije prodaje in trženja celotnega podjetja.

Manager produkta programske opreme ni oseba, ki bi bila direktno odgovorna za trženjske in prodajne aktivnosti, pač pa mora za to obstajati drug oddelek v podjetju, s katerim manager produkta programske opreme sodeluje in lahko vpliva na njihovo motiviranost k trženju produkta.

1.4.2.7 Prodaja in distribucija

Izbira prodajnih poti je tesno povezana z izbranim ciljnim trgom in značilnostmi produkta, vendar strategija prodaje in distribucije ne more temeljiti samo na enem produktu, ampak mora vključevati zahteve prodajne strategije na nivoju podjetja. Tako mora manager produkta programske opreme dobro razumeti prodajno strategijo podjetja in organizacijsko strukturo povezano s tem, da na osnovi tega pripravi strategijo prodaje in distribucije za produkt. Kot menita Kittlaus in Clough (2009, str. 105), je treba ob pripravi strategije prodaje produkta razmisliti tudi o morebitnih prodajnih in distribucijskih strukturah za potrebe samo enega produkta ali produktne linije, vendar to le ob ustrezni podpori v strategiji podjetja.

Manager produkta programske opreme se tipično aktivno vključuje le pri pozicioniranju produkta v strategijo prodaje in distribucije. Na operativnem nivoju njegove aktivnosti potekajo v povezavi s predprodajnimi aktivnostmi in predstavitvijo pri pomembnejših strankah.

1.4.2.8 Podpora in storitve

Z namenom ohranjanja zadovoljstva strank in njihove lojalnosti je naloga managerja produkta programske opreme, da vzpostavi in nenehno vzdržuje ter izboljšuje strukturo podpore in storitev. To vključuje vse aktivnosti s katerimi podjetje zagotavlja podporo trženju in prodaji ter vzdrževanju in podporo strankam (Kittlaus & Clough, 2009, str. 112).

Z zagotavljanjem izobraževanj tako trženju kot prodaji ter podpornim vzdrževalnim službam znotraj podjetja kot izobraževanjem zaposlenih na strani stranke skrbi manager produkta programske opreme za ustrezno tehnično in vsebinsko širjenje znanja. Naloga managerja produkta je, da v okviru vključenih oddelkov skrbi, da imajo ti v okviru svojih oddelkov na voljo sredstva, da se te aktivnosti lahko izvajajo.

Večje in kompleksnejše produkte je redko mogoče vpeljati v delovanje brez dodatnih storitvenih aktivnosti. To so lahko enostavne storitve vzdrževanja ali kompleksnejše prilagoditve tako na strani produkta kot na strani sistemov v organizaciji stranke, s katerimi se produkt povezuje. Te storitve se izvajajo v okviru podjetja samega ali s pomočjo partnerskih podjetij ali neodvisnih svetovalnih podjetij.

2 SKALABILNOST PRODAJE INFORMACIJSKIH REŠITEV

Podjetja v različnih industrijskih panogah iščejo načine, kako zmanjšati stroške ali povečati prodajo z namenom povečanja dobička. S poslovnim modelom, ki ob enakih stroških omogoča povečevanje prodaje, je ta cilj mogoče doseči. Povečevanje skalabilnosti prodaje tako pomeni povečevanje razmerja med stroški in prihodki od prodaje v prid prihodkov prodaje. Poslovni modeli, ki omogočajo povečanje tega razmerja, se imenujejo skalabilni poslovni modeli.

S tehnološkimi in poslovnimi inovacijami, ki so v panogi informacijske tehnologije pogoste, je vpeljava skalabilnih poslovnih modelov lažja, kot v katerih bolj tradicionalnih in proizvodno zahtevnejših panogah. Z nenehnim razvojem panoge v zadnjih 30 letih je resnično uspešen skalabilni poslovni model mogoče vpeljati tako na poslovnem kot na tehnološkem področju z upoštevanjem določenih pravil in zakonitosti.

V tem poglavju bomo raziskali, katere so temeljne zakonitosti, s pomočjo katerih bomo poiskali možne izboljšave za povečanje skalabilnosti prodaje na konkretnem primeru.

2.1 Razvoj poslovnih modelov v panogi informacijske tehnologije

Baden-Fuller in Haefliger (2013) razvoj novih poslovnih modelov in razvoj tehnologije sicer ločujeta, kar pomeni, da inovacija novega poslovnega modela ni nujno povezana z razvojem novih tehnologij in da inovacija novih tehnologij ne pomeni nujno tudi razvoja novega poslovnega modela. Vendar pa potrjujeta, da z razvojem novih tehnologij obstoječi

poslovni modeli dobijo novo dimenzijo. Tehnološka inovacija namreč pozitivno vpliva na odzivnost in hitrost v procesih obstoječega poslovnega modela.

Z razvojem informacijskih tehnologij v zadnjih desetletjih so podjetja skupaj s tehnološkimi inovacijami dodajala nove dimenzije obstoječim poslovnim modelom in razvijala nove poslovne modele, s katerimi so uspela zagotoviti svojo rast. Tehnološki razvoj in prepoznavanje njegovih prednosti, kot ugotavljata Baden-Fuller in Haefliger (2013), sta omogočila uporabo obstoječih poslovnih modelov na nov način in s tem zagotovila precej višjo rast panoge informacijske tehnologije v primerjavi z ostalimi, tradicionalnimi industrijskimi panogami.

Tehnološki razvoj se ne ustavlja, s čimer odpira možnosti za nove poslovne modele, hkrati pa poslovne modele, ki so še nedavno predstavljali novost, postavlja v položaj neustreznega in zastarelega. Kot ugotavlja Cussumano (2008) so se podjetja v panogi informacijske tehnologije v začetku 90-ih let prejšnjega stoletja, ko je povprečna rast panoge konkretnije začela odstopati od rasti bolj tradicionalnih panog, v večji meri ukvarjala z razvojem in prodajo produktov, vendar se je trend ves čas od takrat obračal v smeri prodaje storitev. Iz tega lahko sklepamo, da so storitve tiste, na katerih se temelji in bo v prihodnje temeljila rast prodaje v tej panogi. Programska oprema pa predstavlja osnovo za izvajanje teh storitev.

V zadnjem desetletju se je uveljavilo nekaj poslovnih modelov, katerih uspeh temelji na širši dostopnosti interneta. Temeljijo na tem, da je vsa programska oprema nameščena na strežnikih, dostopnih po internetu, uporabniku pa je dovoljena uporaba povsem zastonj ali zastonj, z uporabo oglasnega prostora čemer se kreira prihodek ali plačila pristojbine. Primeri teh modelov, kot jih navaja Cussumano (2008), so programska oprema kot storitve ali »zastonj, vendar ne zastonj« (angl. *»free, but not free«*).

Uporaba teh novejših poslovnih modelov je enostavnejša v primeru nujenja storitev, ki so splošno uporabne, brez zahtevnih prilagoditev posameznemu uporabniku, oz. kjer uporabnik v zameno za brezplačno uporabo teh niti ne pričakuje. Z večanjem kompleksnosti prilagoditev, ki za podjetje predstavljajo konkurenčno prednost, se možnost uporabe standardnih informacijskih rešitev v obliki storitev sicer zmanjša, vendar še vedno se v okviru kompleksnih rešitev najdejo moduli, ki so lahko uporabljeni v obliki storitev. Cussumano (2008) ugotavlja, da se v podjetjih, ki so več kot 20 let primarno prodajala produkte programske opreme in so prilagajala svoj poslovni model tudi prodaji storitev, razmerje med prodajo storitev in prodajo programskih izdelkov ustali pri 50 % na vsako. Ugotavlja torej, da se padanje prodaje produktov programske opreme v večjih, ustaljenih podjetjih ne zmanjšuje proti 0, ampak se ustali na približno 50 %.

Razvoj in prodaja produktov programske opreme, kot ugotavljajo Cussumano (2008) in Entrepreneur (2016) ter The Economic Times (2016), predstavljata skalabilen poslovni

model. Prav tako lahko kot skalabilen poslovni model opredelimo model masovnih prilagoditev.

2.2 Rast prodaje in skalabilnost prodaje

Z razvojem tehnologije in razvojem novih poslovnih modelov se pri večini uspešnih podjetij, od katerih so bila nekatera šele ustanovljena pred desetletjem ali malo več, kaže, da za njihov uspeh ni bila dovolj samo dobra rast prodaje, ampak je za takšen preskok potrebno več. Odgovor na to je skalabilnost.

Uspešna internetna podjetja, kot so Google, PayPal, eBay, v svojih poslovnih modelih vključujejo vrsto formuliranih elementov, ki jim omogočajo način, na katerega dosegajo takšno rast. Rast pomeni dodajanje virov v enakem tempu, kot rastejo prihodki, ker dodatni viri zagotavljajo večanje prihodkov. Skalabilnost na drugi strani pomeni eksponentno rast prihodkov, medtem ko je rast virov linearna (Growth vs. Scaling, 2016).

Za visoko skalabilne poslovne modele označimo tiste, ki zagotavljajo eksponentno rast dobička. Za razliko od linearnih modelov se pri njih ob rasti prodaje stroški ne povečujejo v enakem obsegu, kar sčasoma prinaša večji dobiček organizacije (Entrepreneur, 2016).

2.3 Definicija skalabilnosti in komponente skalabilnega poslovnega modela

V tehničnem pomenu lahko skalabilnost opišemo kot sposobnost sistema za obvladovanje rastočega obsega dela v vzdržnih okvirih oziroma zmožnost razširljivosti sistema z namenom obvladovanja rasti obsega dela (Bondi, 2000).

Skalabilnost prodaje lahko definiramo kot zmožnost vzdrževanja ali povečevanja prodajne marže, medtem ko se obseg prodaje povečuje (The Economic times, 2016).

2.3.1 Komponente skalabilnega poslovnega modela

Vpeljava skalabilnega poslovnega modela je kompleksen proces. Ne obstajajo namreč splošno določeni kriteriji, z upoštevanjem katerih bi podjetju garantirano zagotovilo uspešen prehod iz obstoječega poslovnega modela na takšnega, ki bi omogočal skalabilnost prodaje. Skalabilen poslovni model je zelo odvisen od vrste posla, s katerim se podjetje ukvarja. Joung (2015) na osnovi obstoječih uspešnih implementacij skalabilnih poslovnih modelov izpostavi 4 komponente, ki prispevajo osnovo za uspešno implementacijo takega modela:

- standardizacija,
- skladnost v upravljanju,
- uporaba obstoječih sredstev na novih področjih in trgih,

- grupiranje strank po lastnostih.

Vpeljava standardov v proces dela omogoča lažjo komunikacijo med različnimi deležniki v procesu. Z uporabo standardov v razvoju produkta programske opreme je sistem bolje obvladljiv, fokus na končni produkt pa večji.

Ljudje z znanjem so ena največjih vrednosti podjetij v panogi informacijske tehnologije. S skladnostjo v upravljanju dosežemo enostavnejše prehode ljudi iz enega projekta na drugega, s tem pa tudi boljši pretok znanja med projekti. To dosežemo s tem, da veljajo enaka pravila in se uporabljajo enaki procesi dela v različnih timih znotraj podjetja.

Z uporabo obstoječih sredstev v primeru razvoja informacijskih rešitev, programskih modulov ali infrastrukture je čas, potreben za prehod na tržišče, krajši.

Grupiranje strank glede na lastnosti omogoča, da se v procesu dela ne ukvarjamo z vsako stranko na individualnem nivoju, ampak vsako stranko glede na njene lastnosti uvrstimo v določeno skupino. Na nivoju teh skupin imamo definirane procese, prilagojene strankam z določenimi lastnostmi.

2.3.2 Model razvoja in prodaje informacijskih rešitev in masovne prilagoditve

V literaturi je poslovni model prodaje produktov programske opreme in model masovnih prilagoditev (angl. *mass customization*) opredeljen kot skalabilen poslovni model. Pri prodaji produktov je namreč potrebna visoka investicija v začetku, ob razvoju produkta, sama distribucija pa naj ne bi predstavljala omembe vrednih stroškov.

Cussumano (2008) je analiziral razvoj podjetij, ki so se v preteklih desetletjih intenzivnega razvoja informacijske tehnologije ukvarjala predvsem z razvojem in prodajo zaključenih informacijskih rešitev ali produktov programske opreme, katere uporabnik samostojno ali s pomočjo plačanih storitev namesti na računalnik. Storitve, izvedene v okviru prodaje teh produktov, nudijo podjetja, ki so proizvajalci samega produkta programske opreme ali zunanja partnerska podjetja. Pri tej raziskavi je Cussumano (2008) ugotovil, da so se v preteklosti veliki proizvajalci produktov programske opreme izogibali prodaji storitev za svoje izdelke, saj je bila prodaja samih produktov dovolj uspešna. Izvajanje storitev, povezanih s prilagoditvami prodanih produktov, so prepuščali partnerskim podjetjem. V obdobju vse večjih zahtev po prilagoditvah produktov programske opreme, kjer programski izdelek kot tak, brez prilagoditev, ni več dovolj, se je investicija v same storitve začela povečevati in tako v veliki večini presega investicijo v sam produkt programske opreme.

Povečanje obsega teh storitev dejansko predstavlja masovne prilagoditve obstoječih produktov.

Podjetja, proizvajalci produktov programske opreme so tako začela povečevati obseg prodaje storitev za svoje obstoječe izdelke. Prihodek podjetij iz vira storitev se je iz leta v leto povečeval. Vendar je Cussumano (2008) v svoji raziskavi ugotovil, da se v podjetjih, ki se tradicionalno ukvarjajo z razvojem produktov programske opreme, razmerje prihodkov med prodajo produktov in prodajo storitev ustali približno pri 1:1, pri čemer privzemamo, da so te storitve povezane s samim izdelkom. Torej na podlagi prodanega izdelka so uspešno prodane in izvajane tudi storitve. Storitve v zvezi z izdelkom lahko delimo na storitve, povezane z infrastrukturo in integracijo izdelka v obstoječe sisteme stranke, ter na storitve, ki prilagajajo funkcionalnosti obstoječega produkta potrebam stranke.

2.4 Dejavniki zagotavljanja dolgoročne skalabilnosti

Dejavnikov, ki vplivajo na tehnično skalabilnost informacijskih rešitev, skalabilnost podjetja ali na skalabilnost prodaje je več. Skupne točke skalabilnosti na teh treh področjih so po mnenju Abbota in Fischerja (2015, str. 3):

- ljudje in njihove vloge,
- organizacijska struktura podjetja,
- ustrezno voditeljstvo.

2.4.1 Ljudje in njihove vloge

Abbot in Fischer (2015, str. 10) trdita, da so ljudje edini najpomembnejši faktor, katerega je potrebno ustrezno obravnavati, če želimo zagotoviti skalabilnost produkta ali procesa. Ljudje so namreč tisti, ki definirajo sistem, na katerem temelji skalabilen poslovni model. Ljudje ga razvijajo in konfigurirajo, uporabljajo in prodajajo ter sprejemajo pomembne odločitve, povezane z njim. Ljudje tudi definirajo procese, ki jih izvaja podjetje in ki zagotavljajo identifikacijo trenutnih in prihodnjih vprašanj, povezanih s skalabilnostjo.

Ljudje kot najpomembnejši faktor pri zagotavljanju skalabilnosti morajo biti ustrezno organizirani, da je izvajanje aktivnosti učinkovito. Organizacijska struktura podjetja mora omogočati čim lažje vključevanje in izključevanje posameznikov ali manjših timov v procese dela. Težko je namreč povečati količino izvedenega dela, če je vključevanje novih ljudi ali timov v procese dela kompleksno in dolgotrajno (Abbot & Fischer, 2015, str. 12).

Organizacijska struktura mora omogočati vzpostavitev metrik, na podlagi česar je mogoče določiti učinkovitost posameznikov ali timov. Rezultati merjenja predstavljajo podlago za odločanje o potrebnih spremembah z namenom povečanja učinkovitosti. Prav tako je pomembno, da organizacija omogoča timom samostojno določanje ciljev in daje možnost, da so ti cilji uresničljivi, kajti v takšnih timih je tudi bolj prisotna pozitivna morala in navdušenje. Organizacijska struktura mora omogočati pozitivne, kognitivne konflikte

znotraj organizacije, saj razširjajo nabor strateških možnosti. Kognitivni konflikti so lažje dosegljivi v multifunkcionalnih timih (Abbot & Fischer, 2015, str. 13–14).

Za uspešno skalabilno organizacijo velja, da morajo biti vloge in odgovornosti jasno definirane in razdeljene. Z jasno definirano odgovornostjo posameznika ali tima se izognemo situacijam, ko dva posameznika izvajata isto delo zaradi tega, ker odgovornost razdelitve dela ni bila jasna. Drugi primer je ta, da se šele v trenutku, ko se rezultati nekega dela že pričakujejo, izkaže, da njegovo izvedbo nihče ni razumel kot del svojih zadolžitev. Takšne situacije bistveno prispevajo k slabi morali v timu, zato jih je treba pravočasno preprečiti (Abbot & Fischer, 2015, str. 23–24).

Abbot in Fischer (2015, str. 25–38) predlagata, da podjetje v svoji organizacijski strukturi jasno definira vsaj naslednje vloge in odgovornosti, ki jih poimenujeta po tradicionalnih vlogah, značilnih za večja podjetja; vendar vloge, lahko z drugim poimenovanjem, morajo biti zasedene v vsakem podjetju s ciljem povečanja skalabilnosti:

- glavni izvršni direktor (CEO),
- finančni direktor (CFO),
- vodja poslovnega področja ali enote,
- direktor informatike,
- odgovornost arhitektov sistema,
- odgovornost razvojnih inženirjev,
- odgovornost razvojno-infrastrukturnih in integracijskih inženirjev,
- odgovornost sistemskih inženirjev,
- odgovornost testiranja in zagotavljanja kvalitete.

V naslednjih podpoglavjih podrobneje opišemo vsako od zgoraj naštetih vlog.

2.4.1.1 Glavni izvršni direktor (CEO)

Glavni izvršni direktor je v primeru vseh vprašanj, povezanih s skalabilnostjo, končni odločevalec. Imeti mora sposobnosti osnovnega razumevanja tehnologije, da z ustrezno postavljenimi vprašanji dobi ustrezne odgovore za sprejemanje odločitev. Njegova naloga je med drugim tudi, da poskrbi za kulturo v podjetju, v kateri bo vprašanje skalabilnosti postavljeno visoko na prioritetni lestvici.

2.4.1.2 Finančni direktor (CFO)

Finančni direktor mora v podjetju predvsem poskrbeti za dovolj visok proračun, ko gre za vprašanja in investicije v skalabilnost sistema. Hkrati mora poskrbeti, da proračun ni pretiran, kar doseže z ustreznim preverjanjem kakovosti napovedanih planov potreb po povečevanju skalabilnosti.

2.4.1.3 Vodja poslovnega področja ali enote

Vodja poslovnega področja je najbolj odgovoren za pravilno napoved rasti rešitve ali produkta tako v poslovnem smislu kot v smislu zahtevanih tehničnih kapacitet. Projekcije glede performančnih zahtev in povečevanja povpraševanja so glavno orodje za določanje točk, ki bodo zahtevale izboljšavo v smislu tehnične skalabilnosti. Preko tega pa skrbi je proračun, namenjen skalabilnosti, realen. Projekcije so na tem mestu pomembne tudi zato, ker se v primeru njihove odsotnosti odgovornost za skalabilnost prestavni na tehnični del organizacije, ki večinoma nima enakega vpogleda v poslovna vprašanja in so odločitve z njene strani lahko precej slabe.

Vodja poslovnega področja je prav tako odgovoren za vzpostavljanje kulture skalabilnosti v podjetju. Z ustreznimi vprašanji tako svojim podrejenim kot nadrejenim ter pomočjo in podporo podrejenim pri reševanju vprašanj povezanih s skalabilnostjo.

2.4.1.4 Direktor informatike

Direktor informatike je v podjetju odgovoren za tehnične procese in tehnološko organiziranost, da bo ta omogočala skalabilnost. Zagotoviti mora, da je skalabilnost v podjetju s tehnološkega stališča stroškovno učinkovita in podprta. Njegova naloga je definiranje strategije informatike podjetja, ki mora vključevati aspekte skalabilnosti.

Z vključevanjem konkretnih, agresivnih, merljivih in dosegljivih ciljev v strategijo mora hkrati zagotoviti, da so posamezniki in ekipe, katerim so naloge doseganja teh ciljev dodeljene, ustrezno strokovno in moralno podprti.

2.4.1.5 Odgovornost arhitektov sistema

Arhitekti, ki načrtujejo strukturo produkta, morajo zagotoviti v času načrtovanja in kasneje ob spremljanju implementacije takšen načrt, ki ima možnosti razširitve v smislu skalabilnosti. Pri tem je treba poudariti, da morajo imeti v vidu, kar se tiče skalabilnosti, tudi možnosti, ki še niso opredeljene kot zahteva, vendar se lahko pojavijo šele po implementaciji, ko se, če se, za to pojavi zahteva s strani vodje poslovnega področja.

Naloga arhitektov je tudi vzpostavitev tehničnega upravljanja standardov in postopkov dela ob razvoju produkta ter skrb, da se vsi ti postopki in standardi izvajajo v praksi.

2.4.1.6 Odgovornost razvojnih inženirjev

Razvojni inženirji so tisti, ki v glavnem vsa pravila, postopke, standarde in zahteve konkretizirajo s tem, ko implementirajo načrtovan produkt. Vsak razvojni inženir mora pri svojem delu imeti v mislih vprašanje skalabilnosti produkta, ki ga gradi.

2.4.1.7 Odgovornost razvojno-infrastrukturnih inženirjev

Razvojno-infrastrukturni in integracijski inženirji izvajajo aktivnosti, povezane tako z aktivnostmi razvoja informacijske rešitve kot z aktivnostmi za sistemsko infrastrukturo, kjer bo rešitev nameščena. V angleščini se je za skupek teh aktivnosti uveljavil izraz DevOps, in sicer zlasti z uvedbo agilnih pristopov v razvoju programske opreme. Izraz je bil prvič omenjen na konferenci Agile 2008, kjer sta Shafter in Debois predstavljala svoje poglede na agilno infrastrukturo (Wikipedia-DevOps, 2016). Ideja skupne odgovornosti razvojno-infrastrukturnih inženirjev je, da se v primeru agilnega pristopa k razvoju programske opreme vključi tudi infrastrukturo ter s tem zagotovi boljšo povezanost med razvojem programske opreme in pripravo infrastrukturnega okolja, v katerem se ta izvaja.

Delo razvojno-infrastrukturnih inženirjev je močno povezano z okoljem, v katerem teče razvita informacijska rešitev. Odgovorni so tako za razvojna kot testna okolja, pripravo namestitvenih skript in dnevno spremljanje delovanja sistema. Tako imajo dober vpogled v trende odzivnosti in so obenem primerni za pripravo predlogov in rešitev, povezanih s skalabilnostjo.

2.4.1.8 Odgovornost infrastrukturnih inženirjev

Infrastrukturni inženirji, kamor štejmo skrb za podatkovne baze, omrežje ipd., so v manjših podjetjih ločeni od ostalih timov in večinoma zagotavljajo infrastrukturno podporo več razvojnim timom. V okviru skalabilnosti produkta je njihova glavna naloga skrb za sisteme, na katerih produkt teče in sisteme uporablja, da so ti ustrezno upravljani in nadzirani.

2.4.1.9 Odgovornost testiranja in zagotavljanja kvalitete

Posamezniki, odgovorni za zagotavljanje testiranja in kvalitete, preverjajo in skrbijo, da produkt zagotavlja rezultate, ki so bili planirani za vsako posamezno razvojno fazo, v smislu funkcionalnosti. Odgovorni so tudi za testiranje informacijske rešitve v kontekstu skalabilnosti. Z dodajanjem novih funkcionalnosti v obstoječo informacijsko rešitev ali dodajanjem drugih informacijskih rešitev na isto okolje se večinoma zahtevnost do okolja poveča. Z znanjem o vseh spremembah na sistemu bi v idealnem primeru skrbeli za opisovanje vpliva vsake spremembe na skalabilnost. Pomembno je, da so obveščeni in se zavedajo vsake spremembe tako na informacijski rešitvi kot na sistemu, da se, predvsem z avtomatskim testiranjem, pripravijo na testiranje skalabilnosti.

2.4.2 Organizacijska struktura

Postavitev organizacijske strukture, s katero posameznike razdelimo v time in skupine timov, ima pomemben vpliv na ustrezno skalabilnost podjetja. Skozi organizacijsko strukturo dejansko zagotavljamo ustrezno komunikacijo, učinkovitost, standarde, kvaliteto in osebno angažiranost posameznika (Abbot & Fischer, 2015, str. 41).

Ustrezna komunikacija in koordinacija sta ključni pri izvajanju nalog, v katero je vpetih več oseb. Slaba komunikacija v primeru razvoja produkta programske opreme ima lahko za posledico nekvaliteten in funkcionalno nedelujoč produkt, povečanje pritožb strank, slabo moralo med posamezniki, vpetimi v delo razvoja ipd.

Patrashkova-Volzdoska, McComb, Green, in Compton (2003) ugotavljajo, da je pomemben tudi način in količina komunikacije. Tudi prepogosto informiranje posameznikov o vprašanjih, ki se jih ne tičejo, niso primer ustrezne komunikacije. V takih primerih se namreč zgodi, da se zaradi velike količine nepotrebnih informacij posamezniki več ne odzivajo niti na informacije, ki so za njihovo delo pomembne. Na drugi strani redka komunikacija povzroči premalo informacij za uspešno delo posameznika ali skupine.

Abbot in Fischer (2015, str. 42) trdita, da se organizacijska učinkovitost izboljšuje takrat, kadar organizacijska struktura poenostavlja potek dela in poslabšuje takrat, kadar so projekti ukalupljeni v strukturo, ki zahteva povečanje komunikacije za uspešno opravljeno delo. Komunikacija mora biti kratka in hitra, da so odgovori na vprašanja čim prej odgovorjeni. V nasprotnem primeru se hitro lahko zgodi, da posameznik čaka na odgovor, povezan z njegovim nadaljnjim operativnim delom, za katerega se zahteva kompleksno usklajevanje, medtem pa čakajoči na odgovor nima opcije izbire drugega učinkovitega dela.

Organizacijska učinkovitost je dosegljiva tudi s pomočjo definiranja ter uporabe standardov. Ignoriranje standardov in pravil se odraža tako na zmanjšanju kvalitete in učinkovitosti kot na povečanju tveganja za kasnejše produkcijske težave. Posledično neupoštevanje standardov pomeni večje stroške z vzdrževanjem, ko je produkt v produkcijski uporabi.

Po mnenju Abbota in Fischerja (2015, str. 44) organizacijo dela v osnovi definirata velikost timov ali skupin in njihova struktura.

2.4.2.1 Velikost tima

Ustrezna velikost tima je odvisna od več faktorjev, ki vplivajo na možnost različnih velikosti timov tudi znotraj posameznega podjetja. Povzeti po Abbotu in Fischerju (2015, str. 44–49) so glavni faktorji naslednji:

- izkušnost projektne vodje,
- stalnost tima,
- izkušnje članov tima,
- obseg nalog projektne vodje,
- zahteve poslovnega okolja.

Prvi trije faktorji kažejo v prid temu, da je število članov v timu manjše, zadnji s poslovnimi zahtevami, ki prinašajo prihodek, to uravnava.

Izkušnost projektnega vodje je pomembna v tem, da je treba upoštevati dejstvo, da manj izkušeni projektni vodja potrebuje za opravljanje dodeljenih nalog več časa kot njegov izkušenejši kolega.

Stalnost tima pomeni časovni obseg obstoja tima. Izkušenejši posamezniki, združeni v tim, ki skupaj deluje že dlje časa, potrebujejo manj vodenja s strani projektnega vodja ter manj interne komunikacije za izvedbo zastavljenih nalog. Posledično takšen tim lahko šteje večje število članov.

Izkušnje posameznikov v timu pomembno vplivajo na izbiro velikosti tima. Tim, sestavljen iz zelo izkušenih posameznikov, bo izvedbo zastavljenih nalog opravil hitreje, hkrati pa je komunikacije v smislu podpore pri reševanju nalog manj, kot pa to velja za tim, sestavljen iz različno izkušenih posameznikov.

Okvir obsega nalog, ki so dodeljene vlogi projektnega vodje, v podjetju prav tako predstavlja faktor pri določanju ustrezne velikosti tima. V opisu del in nalog projektnega vodje je lahko več nalog, ki so lahko precej administrativne narave – od periodičnih razgovorov s člani tima do urejanja kadrovskih nalog za člane tima. Istočasno se projektni vodja lahko ukvarja z vsemi vsebinskimi vprašanji projekta. Daljši kot je ta seznam nalog, za katere je projektni vodja zadolžen, manjša je priporočena velikost tima.

Cilji vodje poslovne enote ali managerja produkta so v glavnem povečanje prihodkov in dobička, izboljšanje konkurenčne prednosti in povečevanje nabora strank. To pa pomeni vedno bolj kompleksne funkcionalne zahteve, prilagojene strankam, ter hiter razvojni cikel s čim hitrejšim prehodom v produkcijo. Vse te zahteve določajo obseg velikosti tima, katerega bo projektni vodja še lahko koordiniral, da bo ta dovolj fleksibilen in se prilagajal zahtevam poslovnega okolja.

2.4.2.2 Organizacijska struktura tima

Eden največjih problemov v razvoju informacijskih rešitev, kot meni Vilet (2007, str. 93), je koordinacija aktivnosti med člani tima. To se še posebej izkaže v primeru, kadar se projekt širi in zahteve postajajo vse bolj kompleksne. Koordinacija aktivnosti je v prvi fazi mogoča s pomočjo ustrezne komunikacije, ki jo lahko formaliziramo s pomočjo rednih sestankov, formalnih preverjanj izvajanja aktivnosti ipd.

Vilet (2007, str. 93) trdi, da je še vedno neformalna medosebna komunikacija glavni način, s pomočjo katere se širijo informacije, zato je pri definiranju organizacijske strukture treba to upoštevati.

Po mnenju Vileta (2007, str. 97–103) se je razvilo več različnih struktur, s katerimi so podjetja, ki se ukvarjajo z razvojem informacijskih rešitev, organizirana. Od teh so nekatere pogostejše uporabljene kot druge:

- hierarhična organizacija,
- matrična organizacija,
- tim vodilnega programerja,
- SWAT tim (angl. *SWAT – Skilled With Advanced Tools*),
- agilni tim,
- odprtokodni način razvoja.

Od zgoraj naštetih Abbot in Fischer (2015, str. 51–68) kot primerne organizacijske strukture za skalabilno organizacijo navajata le:

- hierarhično oziroma funkcijsko organizacijo,
- matrično organizacijo,
- agilno organizacijo.

Hkrati poudarjata, da je od teh najprimernejša slednja, torej agilna organizacija. Zato v nadaljevanju opišemo samo prednosti in tudi slabosti teh organizacijskih struktur.

Izvirna ideja agilne organizacijske strukture je učinkovito medfunkcijsko sodelovanje, usmerjeno v zadovoljitev potreb stranke. V hierarhični ali funkcijski organizaciji se kot slabost izkaže komunikacija med različnimi funkcijami, ki so potrebne za uspešen razvoj informacijskih rešitev. Matrična organizacija z dodano projektno odgovornostjo na obstoječo funkcijsko strukturo to komunikacijo sicer izboljša, vendar s tem doda tudi slabost delitve odgovornosti in problem določanja zahtev in ciljev posameznika. Posledično se lahko zgodi, da so zahteve funkcionalnega dela strukture v nasprotju zahtevam projektnega dela strukture (Abbot & Fischer, 2015, str. 51–59).

Agilen pristop k razvoju programske opreme je nastal kot alternativa tradicionalnim pristopom, katerih strukture so opisane zgoraj. Pristop je opredeljen na podlagi 12 principov, ki so jih definirali Beck idr. (2001). Med drugimi sta definirana tudi principa, ki govorita o komunikaciji, o organizacijski strukturi: »Najboljše arhitekture, zahteve in oblikovanje programske opreme izhajajo iz samoorganiziranih timov« ter »Najbolj učinkovit in uspešen način posredovanja informacij v tim in znotraj tima je pogovor iz oči v oči«.

Abbot in Fischer (2015, str. 59–60) pravita, da je ideja o samoorganiziranem timu podala možnost nove oblike organizacijske strukture, ki ne temelji na različnih vlogah znotraj tima in je usmerjena na zadovoljevanje potreb stranke. Pri tem je posameznik in tim kot celota obravnavan kot storitev za stranko, storitev kot taka pa kvalitetna in zanesljiva. S

pomočjo svojih praktičnih izkušenj sta prišla do ugotovitev, da v večini primerov problem, povezan s skalabilnostjo podjetja in njegovih informacijskih rešitev, ni tehnične narave. ampak organizacijske, ter da so od organizacijske strukture tima odvisne kvaliteta, skalabilnost in zanesljivost rešitev. Tako prideta do sklepa, da pravi skalabilni sistemi nastanejo v kombinaciji arhitekture, organizacije in procesov.

Agilna organizacijska struktura tako zahteva avtonomne in samozadostne time, ki za stranke samostojno zagotavljajo storitev skozi njen celoten življenjski cikel. Takšen tim združuje posameznike z različnimi vlogami in odgovornostmi, kot so razvijalci, testerji, razvoj in operacije, projektno vodenje in infrastruktura. Ti delujejo vzajemno, zaradi vključenih različnih funkcij dela omogočajo večjo inovativnost (Abbot & Fischer, 2015, str. 61–65).

Agilna organizacijska struktura ima tako svoje prednosti kot tudi slabosti, vendar sta Abbot in Fischer (2015, str. 66–68) mnenja da, je ta način organizacijske strukture najprimernejši za podjetje, usmerjeno v skalabilnost. Prednosti, ki jih navajata, so povečanje inovativnosti tima s pomočjo kognitivnih konfliktov, povečanega vpliva in moči tima in zmanjšanja funkcijsko-organizacijskih mej. Kot slabosti navajata težave v primeru prekrivanja lastništva funkcionalnosti ali izvirne kode, kjer naj bi dva ali več timov skrbelo za iste funkcionalnosti, saj v tem primeru tim ne more delovati avtonomno. V tem primeru predlagata rešitev v obliki delovanja po odprtokodni strukturi, kjer določen tim ostalim zagotavlja storitve v obliki knjižnic in so kot takšne uporabljene v ostalih timih.

2.4.3 Voditeljstvo

Definicijo voditeljstva (angl. *leadership*) kot vloge v procesu razvoja programske opreme je, kot ugotavljajo de la Barra, Galdames, Crawford, Soto in Crawford (2015), precej težko opredeliti, saj se definicije iz leta v leto na novo ustvarjajo. Različnim definicijam sta, kot ugotavljajo, skupni dve karakteristiki, in sicer usmerjanje in vpliv. Povzamejo, da vodja izvaja ti dve aktivnosti tako, da postavlja smer in cilj ter vleče ali vpliva na ostale z namenom doseganja zastavljene smeri in cilja. Abbot in Fischer (2015, str. 72) definirata voditeljstvo kot »vplivanje na obnašanje organizacije ali osebe z namenom dosega specifičnega cilja«.

De la Barra et al. (2015) ugotavljajo, da se voditeljstvo spreminja v bolj razpršeno obliko v situacijah, v katerih iz vloge posameznika, vodje, to prehaja na več posameznikov. S takšno razpršitvijo se v primeru hitrih sprememb v specifičnih situacijah omogoči ustrezno hitro reagiranje, saj vlogo vodje prevzame kompetenten posameznik za določeno področje.

Ustrezno voditeljstvo je za skalabilna podjetja izrednega pomena. Kot trdita Abbot in Fischer (2015, str. 71), se je v praksi pomanjkanje pravega voditeljstva večkrat izkazalo kot usodno za nizko rast podjetja ali njegove informacijske rešitve in celo za propad podjetja. Podjetja pri razvoju informacijskih rešitev namreč velikokrat kot najvišjo

prioriteto postavljajo implementirane funkcionalnosti, kar posledično pomeni pomanjkljivosti na drugi strani. To je manjši fokus na stabilnosti rešitve ter ustrezno performančno podhranjenost, ki ima lahko za posledico, ob povečanju količine uporabniških akcij, upočasnitev ali celo ustavitev sistema.

Od vodje se pričakuje, da poleg vizije definira natančne, merljive, realne in dosegljive cilje, ki so časovno opredeljeni in jih je ob definiranih mejnikih mogoče preverjati. Na podlagi rezultatov preverjanja se vodja odloči o prilagoditvah zastavljene poti (Abbot & Fischer, 2015, str. 72).

3 DELOVNI PROCESI NA INFORMACIJSKI REŠITVI ADINSURE

Podjetje Adacta d.o.o. je bilo ustanovljeno pred več kot 25 leti in je vodilni Microsoft Dynamics in Qlik partner v regiji Adriatik ter eno vodilnih podjetij z rešitvami za zavarovalniško industrijo v centralni in vzhodni Evropi. Podjetje zaposluje več kot 370 ljudi v 6 različnih državah: v Sloveniji, Rusiji in Srbiji ter na Hrvaškem, Češkem in Cipru. Podjetja v različnih državah so združena v Adacta Holding s sedežem v Sloveniji.

Naloga podjetij v teh državah je, da skrbijo za implementacijo, integracijo in storitveno podporo strankam podjetja na teh lokalnih trgih. Hkrati skrbijo za razvoj poslov, poslovni stik z lokalnimi strankami ter prodajo in iskanje novih strank in poslovnih priložnosti na lokalnem trgu.

Organizacijsko je podjetje razdeljeno v poslovne enote, razdeljene glede na vertikale trga in tehnološke rešitve, za katere nudi podjetje storitve prek partnerskih pogodb:

- poslovna enota Dynamics NAV,
- poslovna enota Dynamics CRM,
- poslovna enota Dynamics AX,
- poslovna enota Poslovna Inteligenca,
- poslovna enota Zavarovalništvo.

Razvoj in podpora informacijske rešitve AdInsure je tako vključeno v poslovno enoto Zavarovalništvo.

V nadaljevanju bomo analizirali in poiskali povezave ter predlagali spremembe na procesih, povezanih z razvojem in prodajo informacijske rešitve AdInsure. Tako se bomo usmerili predvsem na delo in procese dela znotraj poslovne enote Zavarovalništvo. Delovne procese na nivoju podjetja bomo le omenili v primerih, kadar bi se tudi ti morali prilagoditi notranjim spremembam v sami poslovni enoti.

3.1 Vizija in strategija podjetja

Poslanstvo podjetja, zapisano v Adacta (2016), je »... da z rešitvami izboljšujemo poslovanje naših strank«.

Ugotavljamo, da vizija in strategija podjetja ni jasno zapisana in splošno znana, čeprav je v internih dokumentih večkrat omenjena (Adacta, 2009).

Vendar iz delovanja podjetja in usmeritev dela po posameznih poslovnih enotah lahko povzamemo, da je vizija podjetja postati vodilni ponudnik informacijskih rešitev za zavarovalništvo v vzhodni Evropi in širiti prodajo na zavarovalniško manj razvite trge, širjenje baze strank v jugovzhodni Evropi na področju Microsoft ERP in CRM rešitev ter širjenje baze strank na področju informacijskih rešitev poslovne inteligence.

V poglavju 1.2.2 smo zapisali ugotovitve McGratha (2001, str. 4), da je pomanjkanje vizije in strategije podjetja slabost, ki je v določenih situacijah za podjetje tudi usodna. Zato je smiselno, da vodstvo podjetja poskrbi za jasno zapisano vizijo in strategijo in njeno promocijo tako znotraj podjetja kot navzven.

3.2 Informacijska rešitev AdInsure

Podjetje Adacta je razvilo lastno rešitev AdInsure, ki z implementacijami za stranke ustvari velik delež prihodkov podjetja. Vsebinsko je rešitev zelo kompleksna, posledično pa so rešitve zahtev zapletene. Z namenom boljše predstave funkcionalnega, organizacijskega in tehničnega obsega rešitve jo na kratko predstavljamo.

Informacijska rešitev AdInsure je bila razvita kot lastna rešitev podjetja za podporo vsem procesom dela na področju zavarovalništva. Razvijala se je po modulih in v začetni fazi je bila precej prilagojena zahtevam prvih strank tako na funkcionalnem kot na poslovnem področju. Posledice prilagajanj so se skozi kasnejši razvoj večkrat pokazale kot slabost, katere je bilo mogoče v določenih primerih izločiti, v določenih primerih pa se te posledice čuti še danes.

3.2.1 Kratek opis informacijske rešitve

Informacijska rešitev AdInsure predstavlja kompletno programsko orodje za podporo izvajanju procesov v zavarovalniških družbah. Rešitev sestavlja več funkcijskih modulov, ki so namenjeni podpori procesov posameznih oddelkov znotraj zavarovalniške družbe. Moduli so med seboj neodvisni, vendar dobro povezljivi, kar predstavlja prednost za stranko. Z eno informacijsko rešitvijo je namreč mogoče podpreti vse temeljne zavarovalniške procese, hkrati pa ni potrebna hkratna uvedba rešitve na vseh funkcijskih področjih stranke. Tako je rešitev sestavljena iz modulov:

- upravljanje poslovnih partnerjev,

- produktno jedro,
- prodajni portal in portal za zastopniško mrežo,
- upravljanje s policami,
- knjiženje in računovodstvo,
- registracija in upravljanje škod,
- obračunavanje in zaračunavanje,
- pozavarovanje.

Informacijska rešitev temelji na enotnem podatkovnem modelu in enotni programski kodi. Z leti razvoja se je zahtevnost vzdrževanja in razvoja na enotni programski kodi izredno povečala. Treba je namreč zagotavljati enotno reševanje specifičnih zahtev strank, kar je večkrat vodilo tudi v konflikte med zahtevami.

Z jasno ločitvijo temeljnih funkcionalnosti, ki jih informacijska rešitev ponuja, ter specifičnimi zahtevami posameznih strank se to večanje kompleksnosti vzdrževanja ustavlja. Do ločitve vseh temeljnih funkcionalnosti informacijske rešitve in funkcionalnosti, ki so bile v preteklosti implementirane za specifične potrebe določene stranke, je potrebno strukturo določenim delom informacijske rešitve preurediti ali jo napisati ponovno.

V poglavju 1.1.3 smo opisali stopenjski model s fazami, v katerih se lahko posamezna informacijska rešitev nahaja. Informacijske rešitve AdInsure kot celote ni mogoče enostavno umestiti v eno od teh faz. Zaradi funkcionalno modularne zasnove je smiselno v faze stopenjskega modela umestiti posamezne module. Moduli informacijske rešitve so bili namreč razviti iz različnih izhodišč, kot so:

- vsebinsko predznanje,
- jasnosti funkcionalnih zahtev,
- splošnosti funkcionalnih zahtev ali specifičnosti za posamezno stranko,
- infrastrukturno-tehnične zahteve,
- tehnično znanje in kvaliteta podpore izkušenejših posameznikov,
- časovne omejitve za prehod modula v celoti ali večjih funkcionalnih delov v produkcijo,
- opredelitve pričakovanega števila hkratnih uporabnikov,
- ipd.

Tako lahko module informacijske rešitve razporedimo v te faze, s čimer ugotovimo, kateri so primerni za uporabo in nadaljnji razvoj tudi v smislu arhitekturnih sprememb ali pa so primerni za delni prepis ali prepis v celoti. Arhitekturne spremembe so namreč potrebne za izboljšanje skalabilnosti celotne rešitve.

3.2.2 Pregled načina razvoja informacijske rešitve

Kot že rečeno, je bila informacijska rešitev razvita na osnovi zahtev prvih strank, kar pomeni, da so bile določene funkcionalnosti pisane na kožo njihovim zahtevam, ki so se v kasnejših fazah izkazale za neustrezne za splošno rešitev. V začetku je bilo tudi veliko razvoja izvedenega na principu prototipov, ki so se z določenimi prilagoditvami uporabili v produkcijske namene. Kittlaus in Clough (2009, str. 30) navajata, da praktične pretekle izkušnje kažejo na to, da se razvoj splošne produkcijske rešitve za večje število strank iz implementiranih prototipov v večini primerov ne izplača, saj je vložek za prehod iz prototipa na splošno uporabno rešitev previsok. To se izkazuje tudi v primeru razvoja informacijske rešitve AdInsure. V tem primeru se je več sklopov funkcionalnosti razvilo najprej kot prototip in se kasneje prilagodilo za potrebe prvih strank, prehod na splošno rešitev pa je postal prezahteven. Veliko teh funkcionalnosti se vzdržuje v takšni obliki ter raziskuje možnosti vsaj delnega prepisa na novo, določene pa se že prepisujejo na novo.

Procesi dela v razvoju informacijske rešitve se v zadnjih 10 letih niso ustrezno organizacijsko prilagajali spremembam v številu strank ter obsegu funkcionalnosti same rešitve.

Zaradi povečanih potreb strank se je v proces razvoja in vzdrževanja informacijske rešitve vključevalo nove razvijalce, sama organizacijska struktura pa se povečevanju števila ljudi v procesu razvoja ni ustrezno prilagajala. Prav tako se z večanjem števila, vključenih ljudi v razvoj in vzdrževanje, ni uredilo drugih področij, ki bi zagotavljala dolgoročno vzdržen razvoj in dolgoročno vzdrževanje vrednosti produkta, katerega opisujemo v drugem poglavju. Hkrati pa s tem niso bile zagotovljene komponente skalabilnega poslovnega modela.

V procese dela ni bila vključena standardizacija postopkov, ki bi zagotavljala enovit način strukturiranja programske kode, zapisovanja funkcionalnih in sistemskih zahtev, preverjanja implementiranih funkcionalnosti, razlikovanja med temeljno funkcionalnostjo produkta ter specifično zahtevo stranke, razlikovanja zahtevkov strank med napake in nove funkcionalne zahteve itd. Z intervjuji in opazovanjem vzrokov napak ter načinov reševanja napak in razvoja dodatnih funkcionalnosti smo ugotovili glavne težave, s katerimi se sooča razvojna ekipa. Te so:

- neenotna struktura programske kode, ki rešuje probleme preveč parcialno;
- neuporaba ali uporaba neenotnih programskih vzorcev, ki bi zagotavljali hitrejše in enostavnejše razumevanje programske kode;
- povečano drobljenje odgovornosti za posamezne sklope funkcionalnosti, pri čemer določene sklope funkcionalnosti pozna samo razvijalec, ki jih je implementiral, če je ta sploh še vključen v delo na produktu;
- težave pri identificiranju povezave med implementirano zahtevo ter verzijo produkta, v katero je ta vključen;

- težave pri komunikaciji s stranko in zaračunavanjem izvedenih storitev v fazi vzdrževanja;
- slabo dokumentirane funkcionalne zahteve in nepovezanost funkcionalnosti z zahtevami, na podlagi katerih je določena funkcionalnost implementirana;
- slaba tehnična in uporabniška dokumentacija, ki otežuje predajanje znanja novim članom razvojnih ali podpornih timov ter onemogoča vključevanje partnerjev v proces prodaje in podpore.

Razdeljevanje nalog velikokrat ni bilo razdeljeno projektno, z jasnimi izhodišči ter definiranimi cilji projekta. Naloge so se izvajale na podlagi trenutne prioritete, večinoma je te določala eskalacija stranke. V takšnih primerih izrednega stanja se je začelo iskati najkompetentnejše ljudi, ki bi lahko rešili težave, katere so povod za eskaliranje, kar za te posameznike pogosto pomeni spremembo fokusa, posledica tega pa veliko število nedokončanih področij. S skladnim upravljanjem bi se nabor ljudi, primernih za reševanje težav na posameznem področju, razširil, hkrati pa bi se s tem izboljšala morala večje skupine posameznikov, ko bi ti dobili občutek večje vključenosti in sigurnosti, da imajo znanje in podporo, da sami več prispevajo k uspehu informacijske rešitve.

Vse večje nezadovoljstvo strank in tudi zaposlenih je vodilo vodstvo podjetja v odločitev za korenitejšo spremembo v organizaciji načina dela v poslovni enoti Zavarovalništvo. Ta je bila izvedena v aprilu 2015.

3.3 Organizacijska struktura

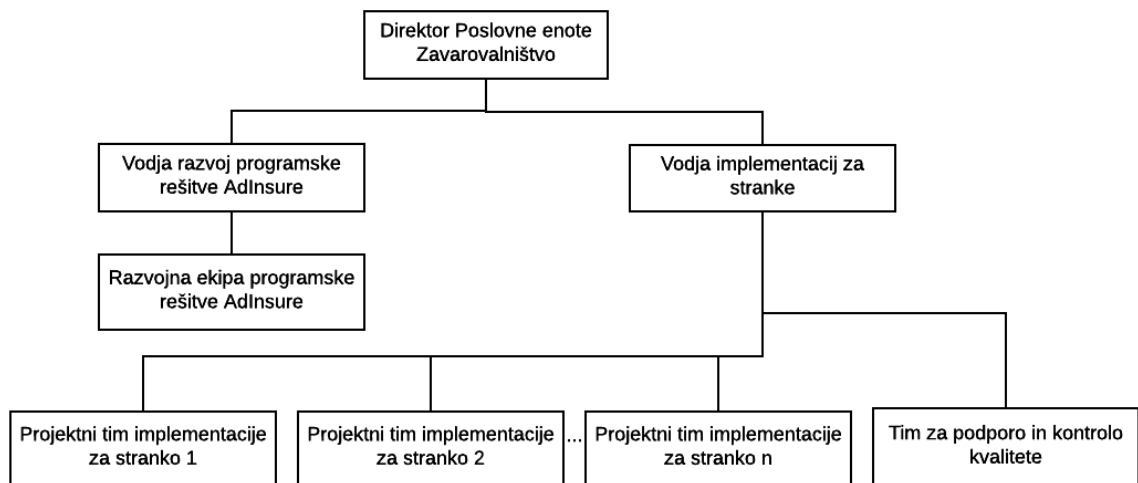
V okviru managementa produkta programske opreme je treba organizacijsko strukturo v poslovni enoti gledati širše. Sama poslovna enota namreč ni samostojna in neodvisna od vplivov drugih oddelkov znotraj podjetja.

3.3.1 Poslovna enota Zavarovalništvo

Cilj sprememb organizacijske strukture v poslovni enoti Zavarovalništvo je bila jasna vzpostavitev projektnih timov, ki so postavljeni na podlagi vsebinskih področij in kategorij funkcionalnosti.

Razvojna ekipa se je razdelila na projekte v okviru temeljnih funkcionalnosti informacijske rešitve in projekte v okviru implementacij za posamezno stranko. Izhodišče za reorganizacijo je bilo to, da se izboljša kakovost same informacijske rešitve, ki se prodaja kot produkt, in se na njeni osnovi izvaja storitve za prilagajanje potrebam posamezne stranke.

Slika 4: Organizacijska struktura poslovne enote Zavarovalništvo



3.3.1.1 Direktor poslovne enote zavarovalništvo

Ključne odgovornosti direktorja poslovne enote Zavarovalništvo (Adacta, 2009) v podjetju so skrb:

- za kakovost, strokovnost in rezultate dela,
- za razvoj organizacije na področju dela,
- za usklajenost in nemoteno delovanje delovnega procesa,
- za stroške poslovanja,
- za poslovanje v skladu z veljavno zakonodajo,
- za uveljavljanje vrednot družbe,
- za pravilen in pravočasen potek informacij znotraj podjetja,
- za zadovoljstvo zaposlenih.

Glavne aktivnosti in naloge, katere direktor poslovne enote pri svojem delu izvaja, so tako (Adacta, 2009)

- sodelovanje pri analizah trga in analiziranje rezultatov zunanjih analiz trga;
- izgradnja in razvoj odnosov s ključnimi kupci na nivoju uprave;
- izvajanje predprodajnih in prodajnih aktivnosti v sodelovanju z oddelkom prodaje in direktorji lokalnih hčerinskih podjetij;
- aktivno sodelovanje pri načrtovanju verzij informacijske rešitve in definiranju posameznih zahtev;
- aktivno sodelovanje pri definiranju prioritet funkcionalnosti;
- operativno izvajanje prodajnih aktivnosti ter izgradnja in razvoj odnosov s ključnimi kupci na nivoju uprave;
- sodelovanje pri dodeljevanju virov v okviru razvoja;

- dodeljevanje virov za investicije v informacijsko rešitev;
- sodelovanje pri planiranju implementacijskih projektov in projektov podpornih storitev.

3.3.1.2 Vodja razvoja informacijske rešitve AdInsure

V domeno razvoja rešitve AdInsure spadajo vse aktivnosti načrtovanja in implementacije ter testiranja. Rezultat teh aktivnosti je izpolnitev funkcionalnih in tehničnih zahtev produkta, ki so opredeljene kot funkcionalnosti produkta in na voljo za samostojno uporabo ali kot osnova, uporabljena v razširitvah in prilagoditvah v primeru implementacije za posamezno stranko.

Ključne odgovornosti vodje razvoja informacijske rešitve AdInsure v sistemizaciji delovnih mest v podjetju, opredeljene kot skrb (Adacta, 2009):

- za kakovost, strokovnost in rezultate dela,
- za razvoj organizacije na področju dela,
- za usklajenost in nemoteno delovanje delovnega procesa,
- za stroške poslovanja,
- za poslovanje v skladu z veljavno zakonodajo,
- za uveljavljanje vrednot družbe,
- za pravilen in pravočasen potek informacij znotraj podjetja,
- za zadovoljstvo zaposlenih.

Aktivnosti, ki jih izvaja vodja razvoja produkta AdInsure (Adacta, 2009):

- pripravlja plan razvoja produkta po posameznih modulih,
- pripravlja in koordinira pripravo tehničnih in funkcionalnih zahtev,
- skrbi za tehnično dokumentacijo,
- skrbi za zagotavljanje in kontrolo kvalitete funkcionalnosti produkta,
- podpora pri prodaji,
- sodelovanje pri oblikovanju cenovne politike družbe.

Ekipo za razvoj informacijske rešitve trenutno šteje okrog 40 članov, ki formalno niso razdeljeni v posamezne time. Zaradi tega je vodenje ekipe zelo zahtevno in težko obvladljivo. Vodja razvoja mora namreč pokrivati zelo široko tehnično in vsebinsko področje same informacijske rešitve.

Na tehničnem nivoju so sicer inženirji vodeni s strani izkušenejših inženirjev, ki jim nudijo podporo pri njihovem delu in dajejo navodila in zahteve za vsakodnevno delo. Vendar je vodja razvoja kljub temu še zelo intenzivno vpet v vsakodnevne aktivnosti z vsakim posameznim inženirjem.

Smiselno bi bilo razmisliti o razdelitvi dela na vsebinske ali tehnične sklope, tako da bi inženirje lahko organizirali v manjše time z določenim projektnim vodjem, ki bi prispeval k ustrezni komunikaciji in preverjanju učinkovitosti izvajanja načrtanih nalog.

3.3.1.3 Vodja implementacij za stranke

Rešitev AdInsure je za posamezno stranko samostojno uporaben le v redkih primerih. Vedno se pričakuje, da se v primerih novih strank izvajajo prilagoditve ali razširitve obstoječe rešitve, ki pokrivajo razlike med zahtevami stranke ter obstoječimi tehničnimi in predvsem funkcionalnimi rešitvami, ki so del informacijske rešitve AdInsure.

Vzroki za potrebne storitve dodelave so: ali v manjkajočih večjih sklopih funkcionalnosti, katerih sama informacijska rešitev ne vsebuje, ali prilagoditvah specifikam načina dela ali okolja, v katerem stranka deluje. Vzroki za te vrste dodelav ali prilagoditev so na primer v:

- funkcionalnosti in načinu dela, ki za zavarovalnico predstavlja konkurenčno prednost,
- integraciji z obstoječimi sistemi stranke,
- zadostitvi zakonskim in regulatornim zahtevam,
- lokalizacijski prilagoditvi.

Vodja implementacij za stranke ima v sistemizaciji delovnih mest (Adacta, 2009) enak zapis ključnih odgovornosti kot vodja razvoja informacijske rešitve AdInsure. Te smo zapisali v 3.3.1.2. V nadaljevanju opisujemo aktivnosti in odgovornosti, povezane s storitvenimi projekti za implementacijo prilagoditev in razširitev funkcionalnosti informacijske rešitve za potrebe konkretnih strank.

Vodja implementacij za stranke je odgovoren za vse aktivnosti, povezane z izvajanjem teh storitev dodelav in prilagoditev za različne končne stranke. Trenutno je v tem oddelku odprtih 9 aktivnih projektov. Za vsak projekt je organiziran samostojen projektni tim z dodeljenim projektnim vodjem. Projektni timi štejejo od 5 do 10 članov.

Aktivnosti, ki jih izvaja vodja AdInsure implementacij, so:

- priprava in nadzor priprave in izvajanje funkcionalne in tehnične analize zahtev novih in obstoječih strank;
- priprava in odgovornost za pripravo funkcionalnih specifikacij z opredeljenimi razlikami glede na informacijsko rešitev AdInsure;
- skrb za pripravo projektnih planov implementacij;
- skrb za izvajanje implementacijskih projektov;
- skrb in izvedba koordinacije med planiranimi verzijami informacijske rešitve AdInsure ter zahtevami in pričakovanji implementacijskih projektov za stranke;
- skrb za izvajanje vsebinske in tehnične podpore strankam;

- skrb za kontrolo kvalitete tako implementacij za stranke kot temeljnih funkcionalnosti same informacijske rešitve;
- podpora pri prodaji.

3.3.2 Oddelek prodaje in trženja

V podjetju je oddelek prodaje in trženja razdeljen po rešitvah, katere podjetje ponuja. Veliko samih prodajnih aktivnosti in podpora prodaji se intenzivno izvaja znotraj v okviru aktivnosti v samih poslovnih enotah.

Področje prodaje informacijske rešitve AdInsure v največji meri pokriva direktor poslovne enote Zavarovalništvo ob podpori direktorjev hčerinskih podjetij, ki delujejo na trgu, na katerem deluje tudi potencialna stranka. Koordinacije med oddelkom prodaje in trženjem ter poslovno enoto ni veliko, saj prodaja vsebinsko tega dela poslovanja podjetja ne pokriva. Direktor poslovne enote tako veliko svojega časa posveča predprodajnim aktivnostim in predstavitvam informacijske rešitve potencialnim strankam.

3.3.3 Vodstvo podjetja

Vodstvo podjetja se približno enakovredno ukvarja z organizacijskimi vsebinami v vseh poslovnih enotah. V poglavju 1.3.1.2 navajamo Eberta (2006), ki pravi, da naj bi se vodstvo oziroma višji management med drugim ukvarjal tudi z definicijo strategije in njenim uresničevanjem ter definicijo in pripravo metrik uspešnosti produktov, kjer to niso le metrike v finančnem smislu.

Vodstvo se v izvajane aktivnosti v okviru poslovne enote Zavarovalništvo predvsem vključuje na področju dodeljevanja virov, plana prodaje in distribucije ter tržnih strateških planov. Tako lahko rečemo, da se te aktivnosti precej dobro prekrivajo z aktivnostmi, opredeljenimi v modelu po Kittlausu in Cloughu.

3.4 Strategija produkta AdInsure

Podobno kot velja za pripravo strategije podjetja velja tudi za strategijo produkta, da mora izhajati iz vizije produkta (McGrath, 2001, str. 18). Za poslovno enoto Zavarovalništvo in informacijsko rešitev AdInsure ugotavljamo, da je vizija ohlapna in v glavnem poudarja izboljšanje informacijske rešitve in povečevanje njene prodaje, natančno pa ne izraža načina oziroma smeri, v kateri bi samo informacijsko rešitev izboljšali.

Posledično je z nejasno vizijo informacijske rešitve tudi strategija nejasna, zato se smer razvoja informacijske rešitve in aktivnosti v povezavi z njo spreminja in bolj prilagaja zahtevam novih strank. Z jasno strategijo bi se aktivnosti, povezane s informacijsko rešitvijo, izvajale v bolj začrtani smeri, katere izvajanje je mogoče preverjati in po potrebi prilagajati. Zahteve strank so vsekakor pomembne in jih je treba upoštevati, naloga managerja produkta programske opreme je, da skupaj s prodajo najde skupno pot. Cilj

prodaje je namreč vedno povečanje prodaje, kar včasih pomeni, da se prodaja tudi funkcionalnosti, katerih v določenem trenutku informacijska rešitev ne podpira in so planirane v kasnejši fazi, kar za časovni okvir stranke ni sprejemljivo. Posledično to za sam razvoj pomeni, da se mora ta prilagajati do te mere, da usmeritve, zapisane v strategiji, nimajo več prave teže.

4 ANALIZA OBSTOJEČIH PROCESOV IN PREDLOG SPREMEMB GLEDE NA UGOTOVLJENA SPOZNANJA

Razvoj in prodaja produktov programske opreme je po definiciji v poglavju 2.3 skalabilen poslovni model. Za razvoj produkta je potrebna določena količina dela, ki je tudi glavni strošek produkta, ta produkt pa se kasneje v enaki obliki ali s prilagoditvami prodaja večkrat različnim strankam. S prodajo zadostnega števila produktov, ko se doseže točka preloma med stroški razvoja ter prihodki od prodaje, pride v tem poslovnem modelu do izraza skalabilnost prodaje. Investicija v pripravo in distribucijo namestitvenih paketov je zanemarljiva v primerjavi s samimi stroški razvoja produkta programske opreme in prihodki od prodaje posamezne instance.

Storitve za prilagoditve ali nadgraditve produkta programske opreme, ki so tesneje povezane s specifičnimi zahtevami stranke, so več ali manj dodatno plačane in se v stroških razvoja samega produkta ne upoštevajo.

Poslovni model, ki se ga podjetje uporablja v primeru informacijske rešitve AdInsure, deluje bolj v smeri masovnih prilagoditev, saj informacijska rešitev AdInsure predstavlja osnovo, na podlagi katere se izvaja implementacija, prilagojena posamezni stranki. Tako je cena implementacije za stranko sestavljena iz cene za določeno število licenc informacijske rešitve in cene storitev, s katerimi se implementirajo prilagoditve in razširitve za stranko. Razlogi za zahteve prilagoditev in razširitev informacijske rešitve so:

- funkcionalnosti, ki predstavljajo konkurenčno prednost stranke,
- lokalizacija in regulatorne, zakonske zahteve trga, na katerem stranka deluje,
- oblikovanje aplikacije in izhodnih dokumentov v skladu z grafično podobo stranke,
- integracija z obstoječimi sistemi pri stranki,
- migracija in podpora migraciji podatkov.

Iz naštetega lahko ugotovimo, da je za uspešno izvedbo implementacije za posamezno stranko potrebno dobro poznavanje lokalnega poslovnega okolja, na katerem bo implementacija delovala. Lokalno poslovno okolje najbolje poznajo subjekti in posamezniki, ki že delujejo v tem okolju. Posledično to pomeni, da bi se prilagoditve in razširitve v okviru implementacije najučinkoviteje izvajale neposredno s strani subjektov, ki to okolje poznajo.

Podjetje lahko doseže povečanje skalabilnosti prodaje s povečano prodajo licenc informacijske rešitve, česar posledica ne sme biti v enaki meri povečanje števila zaposlenih, ki bo izvajala samo implementacijo. S širitvijo prodaje na tuje trge, kjer podjetje še ne deluje, zaradi tega pa slabše pozna lokalno okolje in zahteve, povezane s tem, bi bilo v primeru samostojne izvedbe implementacije potrebno več napora, kot če se implementacija izvaja za okolje, katerega podjetje že pozna.

Tako ugotovimo, da je primeren način povečanja skalabilnosti prodaje ta, da podjetje razvija informacijsko rešitev v smeri, da bo implementacija mogoča s strani lokalnih partnerjev, ki bodo imeli dovolj znanja za njeno izvedbo za stranke na tem lokalnem trgu. Podjetje bi v takem primeru povečalo število prodanih licenc, ne bi pa v enaki meri povečalo stroškov, povezanih s tem, saj bi lokalni partnerji sami izvajali storitve, potrebne za implementacijo.

Iz vsega tega lahko povzamemo, da mora biti cilj podjetja razvoj informacijske rešitve na način, da bo ta neodvisna od lokalnih zahtev, hkrati pa bo podpirala možnost doimplementacije teh zahtev s pomočjo storitev za nadgradnje in prilagoditve, ki se izvajajo v fazi implementacije za posamezno stranko.

Informacijska rešitev mora biti pripravljena na ustrezen način, ki bo dajal rezultate v obliki, ki bo razumljiva tudi subjektom z manj detajlnega vsebinskega in tehničnega znanja.

Rešitev mora zagotavljati modularen, verzioniran, za nazaj kompatibilen sistem, ki je ustrezno dokumentiran. Takšno rešitev je mogoče brez podrobnejšega znanja o načinu implementacije uporabiti in na njej v obliki izvajanja storitev za prilagoditev in razširitev graditi končno rešitev za posamezno stranko.

Z namenom povečanja skalabilnosti prodaje se mora podjetje prek poslovne enote Zavarovalništvo usmeriti v razvoj informacijske rešitve AdInsure v obliki produkta programske opreme. Produkt je mogoče prodajati s pomočjo partnerskih subjektov na različnih trgih. Pot k cilju izvedbe produkta programske opreme, ki bi to omogočal, pa vodi tudi prek uporabe koncepta managementa produkta programske rešitve.

Zato bomo v nadaljevanju analizirali trenutne načine dela, jih povezali in predlagali prilagoditve, da se bodo te ujemale s funkcijami in aktivnostmi, opredeljenimi v modelu managementa produkta programske opreme po modelu Kittlausa in Clougha.

4.1 Izvajanje funkcije analize trga

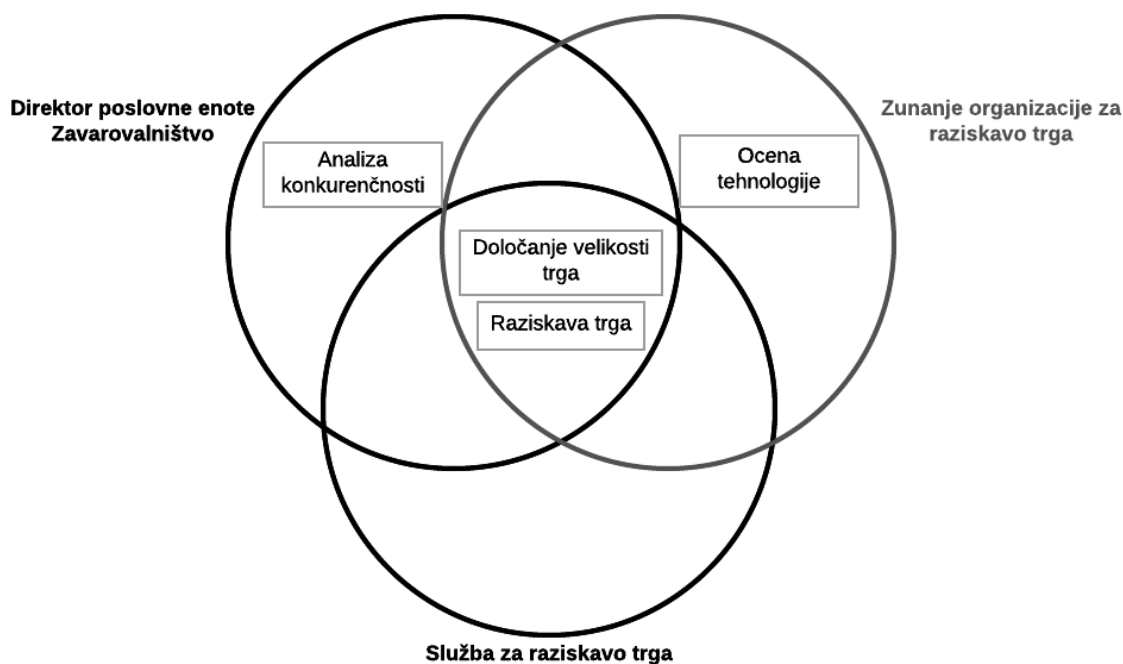
Spremljanje stanja na trgu programskih rešitev za podporo procesom v zavarovalništvu se v podjetju oziroma njegovi poslovni enoti Zavarovalništvo konstantno izvaja s pomočjo kontaktov pri obstoječih strankah, aktivne udeležbe na strokovnih konferencah s področja

zavarovalništva in informacijskih tehnologij, spremljanja rezultatov analiz trga, katere objavlja družba Celent – organizacija za izvajanje in vrednotenje trga in informacijskih rešitev zanj – ter vključenostjo produkta v Celentovo vrednotenje informacijskih sistemov za zavarovalniško industrijo (Adacta Group, 2016).

Na podlagi tega je znanje o stanju na trgu v podjetju dobro prisotno, posledično se hitro identificirajo potencialne nove zahteve. V tržni komunikaciji glede informacijske rešitve s tržiščem je podjetje s svojo prilagodljivostjo v predstavitvah potencialnim strankam ali zunanjim organizacijam za analizo trga zelo uspešno. Med drugim je tudi zaradi tega informacijska rešitev pri analitikih trga dobro ocenjena.

Analiza trga v podjetju se ne izvaja samo za trge, kjer je podjetje že prisotno in zato bolje pozna lokalne regulatorne zahteve, ampak se aktivno usmerja v tržišča s potencialom rasti, kjer trg na tem področju še ni toliko razvit in je potencial za rast visok. Težava s prodorom na te trge je ravno v nezadostnem poznavanju lokalnih pravil in regulacij, kar bi uspešno rešili z vključitvijo lokalnih partnerjev.

Slika 6: Razdelitev izvajanja aktivnosti v okviru funkcije analize trga za informacijsko rešitev AdInsure



Na Sliki 6 je prikazano izvajanje aktivnosti funkcije analize trga po modelu Kittlausa in Clougha (2009, str. 47–49), katerega smo predstavili v poglavju 1.4.2 za primer informacijske rešitve AdInsure.

Aktivnosti v funkciji analize trga ne spadajo med temeljne aktivnosti managementa produkta programske opreme. V primeru informacijske rešitve AdInsure te aktivnosti izvaja v določeni meri direktor poslovne enote Zavarovalništvo in interna služba za raziskavo trga. V pomoč pri raziskavah so podatki in informacije zunanjih konzultantskih podjetij, specializiranih za raziskave trga.

Na podlagi vsega tega lahko sklenemo, da je poznavanje trga v podjetju zelo dobro in ustrezno izvajano za podporo potrebnim odločitvam v zvezi z razvojem informacijske rešitve AdInsure in je s tem ta funkcija v modelu managementa produkta informacijske rešitve dobro pokrita.

4.2 Izvajanje funkcije produktne analize

Informacijska rešitev AdInsure v trenutni obliki ne predstavlja produkta, ki se distribuira preko različnih kanalov v pravi obliki produkta z določeno verzijo. Zato analize produkta s pomočjo eksplicitnih metrik ni smiselno izvajati v obliki, kot jo navajata Kittlaus in Clough (2009, str. 50). Za produktno analizo se uporabljajo predvsem mehke metrike ob izvajanju poslovne analize funkcionalnih zahtev implementacije za posamezno stranko. V tem procesu se poslovne zahteve stranke uparjajo z obstoječimi funkcionalnostmi informacijske rešitve in se tako identificirajo manjkajoče specifične funkcionalnosti in druge zahteve, ki jih stranka pričakuje.

Na podlagi teh razlik se identificirajo tudi zahteve do same informacijske rešitve, ki so ocenjene kot dovolj splošne in ustrezne za vključitev v eno od prihodnjih verzij informacijske rešitve AdInsure. V tem primeru je projekt implementacije za stranko notranji naročnik projekta verzije informacijske rešitve.

Vsekakor bi bilo v okviru produktne analize smiselno definirat eksplicitne metrike, katere je lažje analizirati. Vendar v načinu dela, pri katerem uporaba informacijske rešitve brez izvedbe implementacijskih storitev ni mogoča, je treba eksplicitne metrike poiskati na relaciji povratne informacije, pridobljene iz implementacijskih projektov. Možni primeri eksplicitnih metrik so:

- število javljenih napak na izdano verzijo,
- število novih implementacij za izdano verzijo,
- količina časa, potrebna za nadgradnjo obstoječe implementacije na posamezno izdano verzijo,
- obseg števila spremenjenih ali zamenjanih funkcionalnosti same informacijske rešitve v okviru izvajanja implementacijskega projekta.

Mehke metrike se v praksi uporabljajo vsakodnevno med projektom implementacije za stranko in razvojnim projektom verzije informacijske rešitve, vendar so te premalo analizirane s strani tretje osebe, ki bi na njihovi podlagi zahtevala korektivne ukrepe.

Na Sliki 7 so predstavljene aktivnosti produktne analize, kot se izvajajo na primeru informacijske rešitve AdInsure. Za to funkcijo smo v predstavitvi modela po Kittlausu in Cloughu v poglavju 1.4.2 prikazali dve temeljni aktivnosti managerja produkta programske opreme. To sta **analiza učinkovitosti produkta** in **analiza zadovoljstva strank**. S Sliko 7 ugotavljamo, da ti dve temeljni aktivnosti v primeru informacijske rešitve AdInsure opravlja vodja implementacij za stranke.

Učinkovitost produkta in zadovoljstvo strank se meri predvsem s pomočjo mehkih metrik, ki dajejo sliko o ustreznosti in kakovosti funkcionalnosti informacijske rešitve. Večina informacij o tem pride prek dnevnega dela na projektih za implementacije, s katerimi se sooča vodja implementacij za stranke.

Aktivnosti iskanja priložnosti in analiza zmag in porazov ne spadata med temeljne aktivnosti managerja produkta programske opreme, pač pa jih ta le orkestrira. Aktivnost iskanja priložnosti za izboljšave v informacijski rešitvi ter njeni poziciji in sprejetosti med strankami izvaja direktor poslovne enote. Aktivnost analize zmag in porazov, s katero se ugotavlja vzroke uspešne ali neuspešne uporabe produkta ali njegovih komponent, se ne izvaja v okviru podjetja.

Slika 7: Razdelitev izvajanja aktivnosti v okviru funkcije produktne analize za informacijsko rešitev AdInsure



Za funkcijo produktne analize lahko trdimo, da se v določeni meri izvaja, vendar ne celovito, saj se pri odločitvi za korektivne ukrepe večkrat upoštevajo parcialne zahteve implementacijskih projektov. Z uvedbo funkcije managerja produkta programske opreme, ki bi v interesu vzdrževanja dolgoročne vrednosti produkta urejal razmerja med zahtevami implementacijskih projektov ter izvajanjem korektivnih ukrepov, bi podjetje pridobilo dodatno kontrolo in poenotenje.

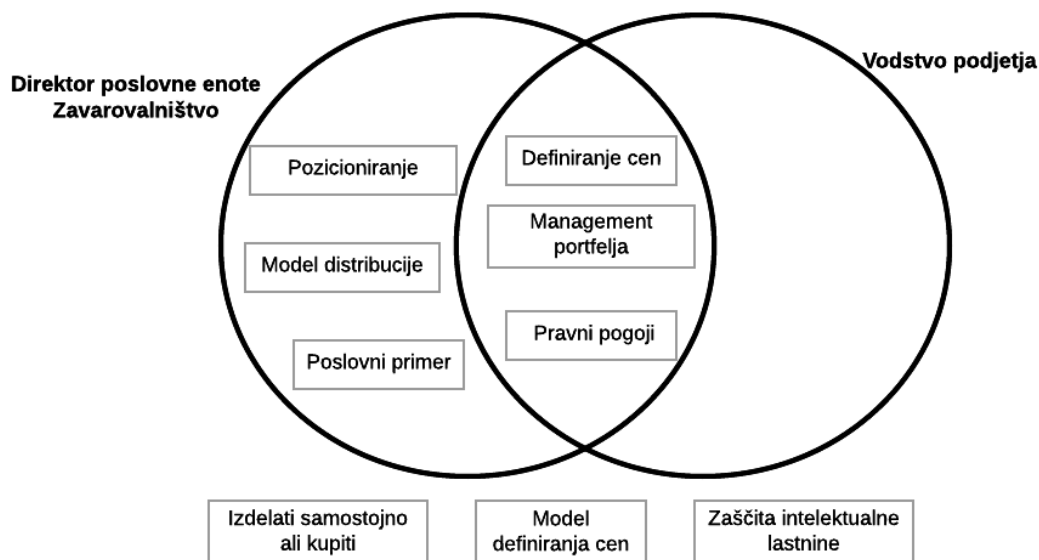
4.3 Izvajanje aktivnosti v povezavi s funkcijo strategije produkta

Strategija produkta naj bi torej odražala smer evolucije vprašanj, povezanih s produktom programske opreme, na različnih področjih, kar smo opisali v 1.4.2.3. Za informacijsko rešitev AdInsure smo v 3.4 ugotovili, da vizija produkta ni jasno zapisana, ampak je ohlapna, in si jo vsak posameznik predstavlja drugače. To predstavo dobi predvsem iz aktivnosti, ki se dogajajo v podjetju v kontekstu informacijske rešitve in implementacij na njeni osnovi. Enako smo ugotovili za strategijo produkta, ki bi morala opredeljevati način dosega ciljev v viziji.

Zaradi tega je smiselno, da se na nivoju poslovne enote definira strategija produkta za obdobje 2 do 5 let, ki bo opredeljevala:

- smernice funkcionalnega obsega razvoja informacijske rešitve,
- ciljne trge,
- jasno pozicioniranje informacijske rešitve na trgu programske opreme za zavarovalniško industrijo,
- pričakovan razvoj trga in dolgoročni načrt razvoja produkta ter virov za uresničevanje te strategije,
- funkcionalno obseg področij katere informacijska rešitev pokriva in česa ne pokriva.

Slika 8: Razdelitev izvajanja aktivnosti v okviru funkcije strategije produkta za informacijsko rešitev AdInsure



V poglavju 1.4.2.3 smo opisali aktivnosti, kise izvajajo v funkciji strategije produkta, katere Kittlaus in Clough (2009, str. 54–76) opredeljujeta kot temeljne aktivnosti managerja produkta programske opreme. Na Sliki 8 prikazujemo trenutno izvajanje teh aktivnosti za informacijsko rešitev AdInsure.

Po modelu Kittlausa in Clougha iz poglavja 1.4.2 je **management portfelja** aktivnost, katero primarno izvaja vodstvo podjetja. V primeru informacijske rešitve AdInsure se ta aktivnost deli med vodstvo in direktorja poslovne enote Zavarovalništvo. Poleg tega se tako delita še **aktivnosti definiranja cen** in določanja **pravnih pogojev**. Informacijska rešitev trenutno ni prodajana samostojno v obliki produkta programske opreme, ampak vključuje vedno tudi storitve razširitev in prilagoditev za vsako posamezno stranko. S tem je oblikovanje pravnih pogojev in definiranje cen definirano na nivoju pogodbe za implementacijo, ki vključuje tako pogoje uporabe informacijske rešitve kot za izvedbo storitev razširitev in prilagoditev za potrebe konkretne stranke.

S **pozicioniranjem produkta** na trgu informacijskih rešitev za zavarovalniško industrijo se primarno ukvarja direktor poslovne enote in je pri tem zelo uspešen, kar se kaže v dobri oceni informacijske rešitve v poročilih konzultantskega podjetja Celent, ki je specializirano v ocenjevanje stanja informacijske tehnologije za področje zavarovalništva.

Model distribucije je v primeru informacijske rešitve AdInsure trenutno jasen, vendar za povečanje skalabilnosti prodaje neustrezen, saj predvideva prodajo informacijske rešitve le neposredno s prodajo storitev za razširitve in prilagoditve.

Poslovne primere večjih sklopov funkcionalnosti najbolje pozna direktor poslovne enote, katerih vrednost je v večji meri definirana prek večje zanimivosti informacijske rešitve na trgu ter posledično potenciala za več prodaje. Dodatno vrednost določajo obstoječe stranke, ki čakajo na dodatne funkcionalnosti v obstoječih implementacijah.

Z aktivnostmi odločanja, ali **izdelati samostojno ali kupiti, definiranja modela cen ter zaščite intelektualne lastnine**, se v podjetju nihče ne ukvarja. Aktivnost odločanja glede samostojne izdelave ali nakupa obstoječe rešitve v tem primeru ni relevantna, saj je rešitev že razvita in jo je treba le ustrezno preoblikovati. Ostali dve aktivnosti pa bi z vpeljavo koncepta prišle do izraza in bi jih moral manager produkta aktivno prevzeti.

Ocena smiselnosti vpeljave koncepta v okviru funkcije strategije produkta

Iz perspektive funkcije strategije produkta bi bila uvedba koncepta managementa produkta programske opreme za informacijsko rešitev AdInsure potrebna. S tem bi dosegli v prvi vrsti jasno vizijo in strategijo produkta, ki bi se ustrezno bolj fokusirano izvajali. Poleg tega bi aktivnosti, ki se trenutno ne izvajajo, dobile svojega lastnika, na katerega bi se preneslo tudi del aktivnosti, ki jih trenutno izvaja direktor poslovne enote Zavarovalništvo.

Aktivnosti bi tako pridobile večji fokus ter pozornost. Predpogoj za to je, seveda, da dobi pozicija managerja produkta programske opreme zadostno podporo s strani vodstva podjetja. S tem bi bil dosežen tudi večji fokus pri vsakodnevnih vprašanjih in odločitvah, ki se pojavljajo tekom izvajanja projektov implementacij za stranke ter kombinacija teh vprašanj v povezavi s strategijo podjetja.

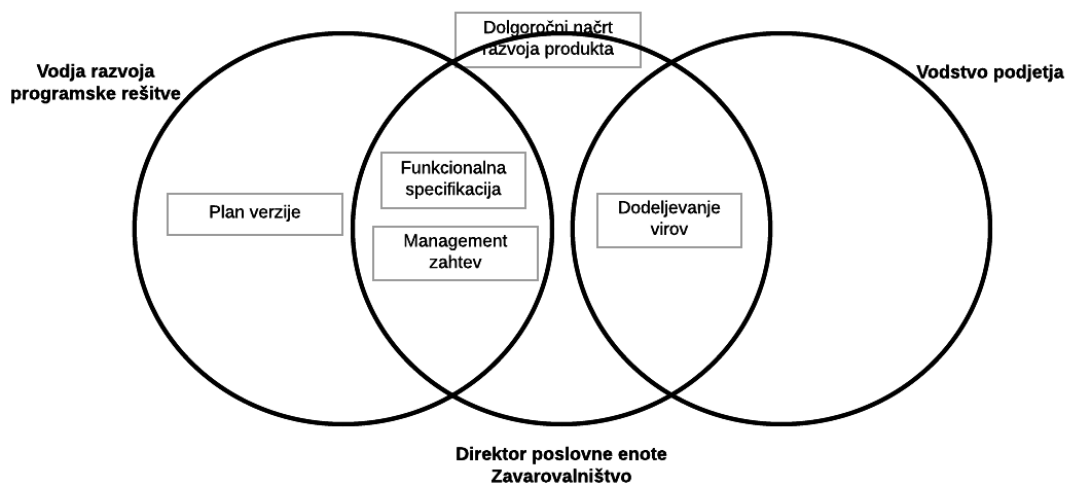
4.4 Izvajanje aktivnosti v funkciji planiranja produkta

V okviru informacijske rešitve AdInsure se planiranje verzij in funkcionalnosti, vključenih vanje, tesno povezuje s funkcionalnimi zahtevami obstoječih strank ter implementacijskimi projekti za stranke, za katere se prva verzija implementacije šele pripravlja. Obstoječe stranke večkrat pogojujejo nadgradnjo z vključenostjo funkcionalnosti v verzijo, ki zanje predstavljajo dodano vrednost. V primeru implementacijskih projektov za nadgradnje in prilagoditve te zahteve izhajajo predvsem iz prodajnih aktivnosti, katere, kot smo ugotovili v poglavjih 3.3.1.1 in 3.3.2, v glavnem izvaja direktor poslovne enote Zavarovalništvo. Tako prek njega prihaja velik del funkcionalnih zahtev.

Velikokrat obljube in dogovori z obstoječimi strankami in njihovimi implementacijskimi projektnimi plani določajo časovne okvire izdajanja verzij, kar je včasih, v celotnem obsegu in z ustrezno kvaliteto, nemogoče izpolniti.

Pomembno vlogo pri planiranju verzije imajo tudi potencialne nove stranke oziroma proces prodaje rešitve, katere se lažje pridobiva s prikazom čimprejšnjega delovanja funkcionalnosti, ki jih stranke pričakujejo in zahtevajo pred dokončno odločitvijo. Tako ima prodaja pomemben vpliv na odločitve pri planiranju verzij.

Slika 9: Razdelitev izvajanja aktivnosti v okviru funkcije planiranja produkta za informacijsko rešitev AdInsure



Vse aktivnosti v okviru funkcije planiranja produkta spadajo po Kittlausu in Cloughu (2009, str. 76–94) med temeljne aktivnosti managerja produkta programske opreme. Slika 9 prikazuje razdelitev teh aktivnosti med različne vloge v podjetju. Tako vidimo, da se temeljne aktivnosti delijo med več vlog v podjetju.

V okviru te funkcije naj bi se pripravljala in dopolnjeval **dolgoročni načrt razvoja produkta**, katerega značilnosti in namen sta opisana v 1.4.1.3. V okviru informacijske

rešitve AdInsure sicer obstaja okvirni načrt verzij za 1 leto vnaprej in zaradi tega je na Sliki 9 ta aktivnost prikazana kot delno izvajana s strani direktorja poslovne enote. Potrebno bi bilo bolj natančno spremljanje realizacije tega letnega načrta verzij in določiti korektivne ukrepe ob ugotovitvah, da se ta načrt ne realizira v pričakovanih okvirih. Trenutno se namreč postavke z načrta, ki bi že morale biti izvedene, še vedno razumejo in komunicirajo kot plan za prihodnost, s tem pa načrt nima potrebne teže oziroma s časom izgublja svoj pomen.

Plan verzije pripravlja vodja razvoja informacijske rešitve AdInsure, pri čemer upošteva prioriteten seznam zahtev. Usklajevanje pri pripravi plana verzije poteka ne relaciji med direktorjem poslovne enote Zavarovalništvo in vodjem implementacij za stranke.

Funkcionalne specifikacije informacijske rešitve in **management zahtev** pripravlja direktor poslovne enote. Zaradi velikega obsega njegovih zadolžitev so te ohlapne in nejasno definirane. Kittlaus in Clough (2009, str. 79) navajata druge vire, ki ugotavljajo korelacijo med uspešno realizacijo produkta in kvaliteto zahtev. Prav tako je management zahtev, kot ugotovljeno v 1.4.1.4, izredno pomembna aktivnost za uspeh produkta.

Ocena smiselnosti vpeljave koncepta v okviru funkcije planiranja produkta

Z uvedbo pozicije managerja bi se izvajanje teh aktivnosti, ki so trenutno razdeljene v glavnem med direktorja poslovne enote Zavarovalništvo in vodjo razvoja informacijske rešitve, izboljšal pregled nad zahtevami. Istočasno bi se razbremenilo ti dve vlogi, ki trenutno opravljata večino teh aktivnosti.

Prednost vpeljave ločene vloge managerja produkta programske opreme pa bi bila tudi v tem, da bi se planiranje verzij produkta in funkcionalnosti, zajetih v posamezno verzijo, izvajalo bolj neodvisno. Trenutno so vplivi in pritiski s strani prodajnih aktivnosti zelo močni, saj prek zavez in obljub, izdanih strankam, razvojni oddelek postavlja v podrejen položaj. Za dolgoročno ohranjanje vrednosti informacijske rešitve in njene kvalitete to ni ustrezno.

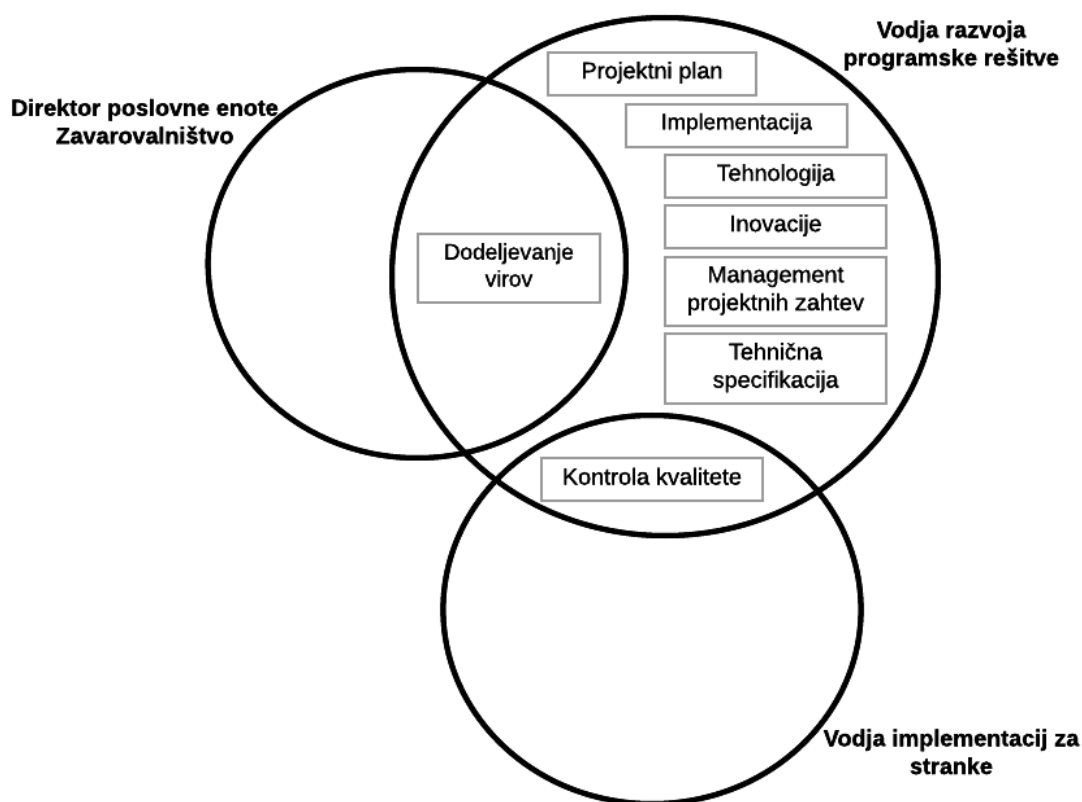
4.5 Izvajanje aktivnosti v funkciji razvoja

Funkcionalne in ostale zahteve za razvojni oddelek so za informacijsko rešitev AdInsure podane s strani direktorja poslovne enote Zavarovalništvo, delno pa prihajajo iz implementacijskih razvojnih projektov za posamezne stranke. Zahteve so podane v precej neformalni obliki, brez opisanih natančnih funkcionalnih specifikacij. Uvrščanje zahtev v plane verzij je v kombinaciji s funkcijo planiranja verzij izvedeno v mesečnih ciklih. Razvoj informacijske rešitve uporablja agilni pristop k razvoju programske opreme po SCRUM metodologiji. Ta metodologija predvideva razvojne cikle, sprinte v krajših časovnih obdobjih od 1 do 4 tednov, po koncu katerih se preveri izvedeno stanje in definira

cilje za naslednji cikel (Hauer, 2013, str. 14–16). Agilne metode v razvojnem procesu produkta predlagata tudi Kittlaus in Clough (2009, str. 95).

Oddelek, ki se ukvarja z razvojem informacijske rešitve, obsega več kot 40 razvijalcev, ki vsebinsko in tehnično pokrivajo različna vsebinska področja produkta in skrbijo za implementacijo in management projektnih zahtev po vsebinskih področjih. Na Sliki 10 je predstavljena razdelitev aktivnosti v funkciji razvoja produkta, katere Kittlaus in Clough (2009, str. 94–95) ne prištevata med temeljne funkcije managerja produkta programske opreme.

Slika 10: Razdelitev izvajanja aktivnosti v okviru funkcije razvoja produkta za informacijsko rešitev AdInsure



Delitev virov med projekti implementacij za stranke in razvojnim oddelkom za razvoj same informacijske rešitve se izvaja na nivoju poslovne enote. Ekipa v razvojnem oddelku so precej stabilne, saj je sam razvoj produkta dolgotrajen proces, ki zahteva kompetentne ljudi, zato se dogajajo prehodi iz tega oddelka v implementacijske projekte le izjemoma.

Projektne plan izvaja vodja razvoja informacijske rešitve v mesečnih ciklih po SCRUM metodologiji. Enako velja za **management projektnih zahtev**, pri katerih mu vsebinsko pomagajo izkušenejši inženirji po posameznih vsebinskih področjih.

Tehnične specifikacije pripravljajo in skrbijo za ustrezno izvajanje same **implementacije** zahtev. Izvajajo jih arhitekti programske opreme, ki so v oddelku razdeljeni glede na tehnično specifična področja.

Razvojni oddelek je glede tehničnih in vsebinskih kompetenc zelo dobro podprt in dobro pokriva aktivnosti, povezane s **tehnologijo** in **inovacijami**. V glavnem pa se s tem ukvarjajo arhitekti programske opreme, katerim so dodeljena različna tehnična področja.

Težave v razvoju informacijske rešitve predvsem izhajajo iz preteklega načina dela, ko so bili inženirji, ki sedaj spadajo v oddelek razvoja informacijske rešitve, dodeljeni na več implementacijskih projektov, kjer so pokrivali določeno vsebinsko področje. Z organizacijsko spremembo se še vedno dogaja, da se od teh inženirjev pričakuje večja angažiranost na implementacijskih projektih. Druga večja težava, ki se kaže v oddelku razvoja informacijske rešitve, pa je količina visokoprioritetnih zahtev s strani implementacijskih projektov. Te izhajajo iz zavez danih strankam, kjer se je, ali pričakovalo, da bo čas potreben za pripravo rešitve krajši, ali pa so ob določanju prioritet v preteklosti te zahteve izpadle iz planiranja verzije. Tako je konflikt med implementacijskimi projekti in oddelkom razvoja informacijske rešitve stalno prisoten.

Tako ugotovimo, da si vlogo internega naročnika razvojnega oddelka, kot zapisano v 1.4.2.5 po Kittlaus in Clough (2009, str. 94), delita

- direktor poslovne enote z zahtevami novih funkcionalnosti predvsem iz vira potencialnih novih strank,
- vodja implementacij za stranke z zahtevami iz izdanih zavez obstoječim strankam.

Slabost razvojnega oddelka je, da nima svoje testne ekipe, ki bi preverjala kandidate za izdane verzije pred samo izdajo in na vsakodnevem nivoju skrbela za preverjanje sprememb v aplikaciji. Tako si vire za **kontrolno kvaliteto** deli z implementacijskimi projekti. Posledično se tako dogaja, da rezultati dela testne ekipe niso dovolj vidni ali slišani in se izvedene napake na odpravljajo sproti.

Druga slabost razvojnega oddelka je, da je število razvijalcev precej veliko. Ti pa niso razdeljeni v formalne time, katerim se dodeli projektne vodje, odgovornosti in jasna pričakovanja z definiranimi cilji.

Ocena smiselnosti vpeljave koncepta v okviru funkcije razvoja

Glede na to, da več ali manj vse zahteve za razvojni oddelek prihajajo s strani direktorja poslovne enote oziroma implementacijskih projektov, ki so prav tako pod njegovo kontrolo, lahko rečemo, da v tem primeru sama formalna uvedba pozicije managerja produkt programske opreme dela v razvojnem oddelku ne bi bistveno spremenila. V

razvojem oddelku je treba nadaljevat z agilnimi metodami v razvojnem procesu, ki bodo ob veliki kompleksnosti rešitve prinesle izboljšave.

Istočasno je smiselno razmisliti o delitvi razvojnega oddelka v time glede na vsebino in tehnologijo. Manjše ekipe so po mnenju Abbota in Fischerja (2015, str. 44–47), zapisano v 2.4.2.1, bolj usmerjeno vodene.

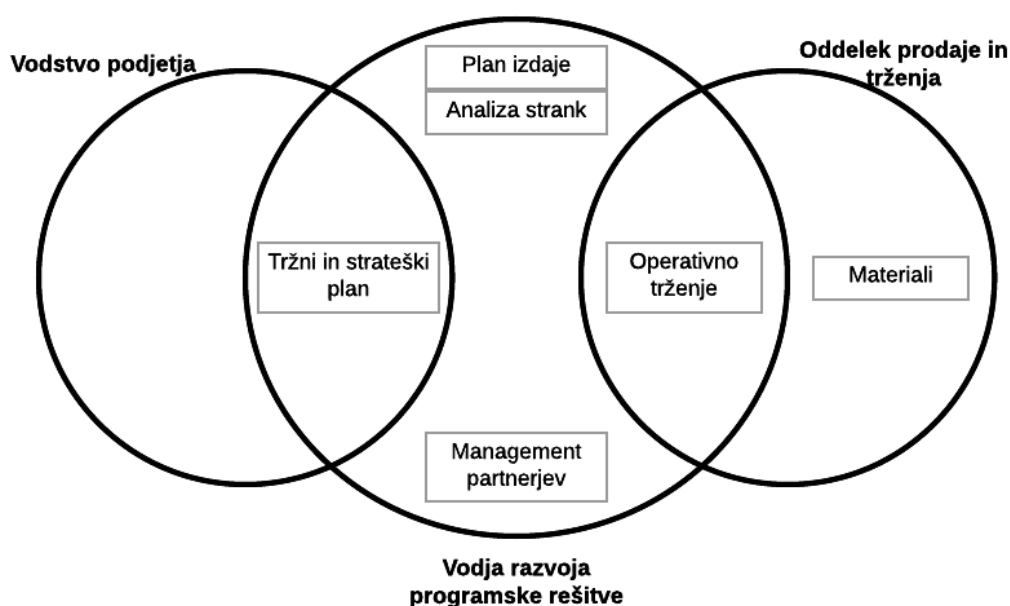
Z uvedbo testnega tima znotraj samega razvojnega oddelka, ki bi dnevno skrbel za preverjanje vseh sprememb, na kreirane napake opozarjal ter bil ustrezno slišan in upoštevan, bi se kvaliteta same informacijske rešitve tudi izboljšala.

4.6 Izvajanje funkcije trženja

Po mnenju Kittlausa in Clougha (2009, str. 97) se naj manager produkta programske opreme ne bi direktno ukvarjal s prodajnimi in trženjskimi akcijami, ampak bi naj bila to naloga oddelka za trženje in prodajo, vendar v manjših podjetjih, kjer ni ločenega oddelka trženja, se s trženjem produkta ukvarja tudi manager produkta programske opreme. Po njunem modelu pa sicer aktivnosti v tej funkciji ne spadajo med temeljne aktivnosti managementa produkta programske opreme.

V praksi informacijske rešitve AdInsure skrbi za ustrezno zastopanost rešitve v tržni strategiji podjetja direktor poslovne enote Zavarovalništvo in celotno vodstvo podjetja. Rešitev je ena glavnih strateških usmeritev podjetja, zato je tudi ustrezno podprta in se z notranjo promocijo produkta ni treba posebej ukvarjat.

Slika 11: Razdelitev izvajanja aktivnosti v okviru funkcije trženja za informacijsko rešitev AdInsure



Na trgu zavarovalniških informacijskih rešitev vse trženjske aktivnosti izvaja direktor poslovne enote. Na tržiščih posameznih držav mu podpora pri teh aktivnostih nudijo direktorji hčerinskih podjetij v teh državah. Predvsem so to države Balkana in Rusija. Na tržiščih izven teh držav se to izvaja iz Slovenije. Okvir trženjskih aktivnosti, ki se izvajajo za informacijsko rešitev AdInsure, zajema:

- analizo trga in potencialnih strank po državah ter analizo obstoječih strank in njihovih zahtev in pričakovanj;
- priprava predstavitev in demonstracij delovanja rešitve pri potencialnih strankah z definiranimi uporabniškimi primeri na zahtevo strank;
- analiza konkurenčnih rešitev in določanje prednosti in slabosti teh rešitev ter iskanje možnosti vpeljave prednosti konkurenčnih rešitev v informacijsko rešitev AdInsure.

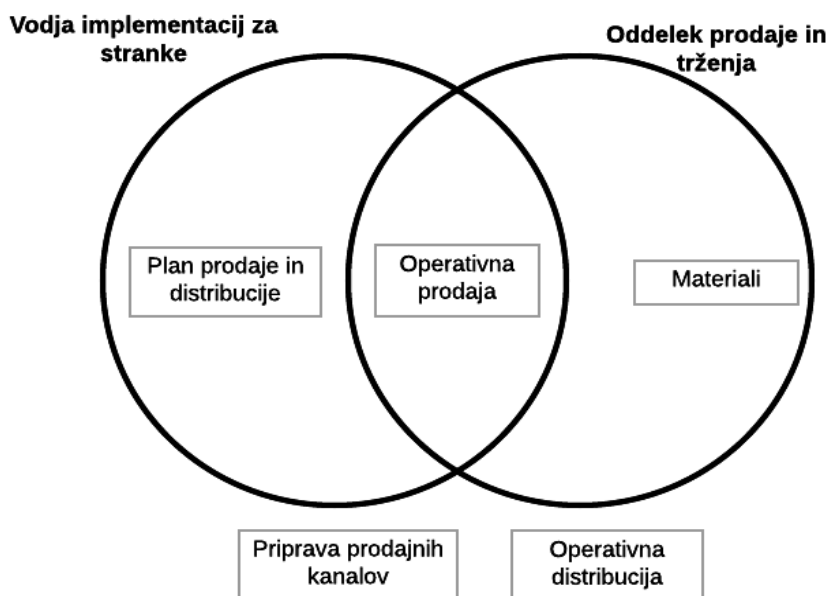
Na Sliki 11 je prikazana razdelitev aktivnosti v funkciji trženja. Iz nje ugotovimo, da se aktivnosti, ki jih izvaja direktor poslovne enote, v večini prekrivajo z aktivnostmi funkcije trženja. Z doslej obravnavanimi temeljnimi aktivnostmi managementa produkta programske opreme smo ugotovili, da se aktivnosti, ki jih izvaja direktor poslovne enote, dobro prekrivajo z aktivnostmi managerja produkta programske opreme. V tej funkciji pa se aktivnosti direktorja poslovne enote bolj kažejo v vlogi trženjskega oddelka.

4.7 Izvajanje funkcije prodaje in distribucije

Informacijska rešitev AdInsure ni oblika produkta za široko potrošnjo, katerega je mogoče prodati in namestiti pri stranki brez dodatnih storitev, potrebnih za prilagoditve. Prav tako trenutno sama rešitev v implementacijskem pogledu, ki vsebuje tudi ustrezno tehnično dokumentacijo, ni v primerni obliki, da bi bila strategija partnerske prodaje mogoča. Trenutno se uporablja strategija direktne prodaje. Slika 12 prikazuje razdelitev aktivnosti prodaje, ki kaže, da **priprava prodajnih kanalov** in **operativna distribucija** nista izvajani ravno zaradi načina direktne prodaje. V praksi to pomeni, da večino prodajnih aktivnosti informacijske rešitve izvaja direktor poslovne enote Zavarovalništvo ob podpori direktorjev hčerinskih podjetij v državah, na trgih, na katerih primerno deluje tudi potencialna stranka.

Rezultatsko je proces prodaje trenutno v podjetju zelo uspešen, kar gre pripisati predvsem zelo dobremu trženju in demonstraciji zahtevanih prilagajanj zahtevam posameznih potencialnih strank. To prilagajanje pa ima tudi negativne posledice v tem, da se lahko projekt implementacije za stranko hitro zaplete do te mere, da se pojavlja vprašanje profitabilnosti samega implementacijskega projekta, ki v fazi prodaje niso bile ustrezno analizirane. Vendar so to predvsem rezultati direktne prodaje, kjer se prodaja končna implementacija za stranko. Cena implementacije je sestavljena iz dela za uporabniške licence informacijske rešitve AdInsure in dela za stroške storitev prilagoditev in razširitev po zahtevah stranke.

Slika 12: Razdelitev izvajanja aktivnosti v okviru funkcije prodaje in distribucije za informacijsko rešitev AdInsure



Povečanje skalabilnosti prodaje na takšen način težje dosežemo, saj je za vsako implementacijo potrebna svoja projektna skupina, kar pa več ali manj linearno povečuje tudi stroške. V primeru, da pa se izvaja implementacija za stranko, ki deluje na trgu, katerega znanja o specifičnih zahtevah ni v podjetju, je strošek implementacije še dodatno višji.

Predlog spremembe strategije prodaje

Povečanje skalabilnosti prodaje je mogoče doseči predvsem z osvajanjem tujih trgov, ki imajo vsak svoje specifikke in zahteve. Lokalni trg in njihove zahteve najbolj poznajo podjetja, ki že delujejo na tem tržišču. Poznavanje lokalnega trga in razmer je torej prednost, katero podjetje z direktno prodajo težko doseže oziroma je za podjetje predraga. Pogoji za povečanje skalabilnosti prodaje informacijske rešitve je potrebna ustrezna vključitev prodaje preko lokalnih partnerjev v strategijo prodaje.

Partnerji na lokalnih trgih izvajajo implementacijske storitve po meri stranke in storitve podpre strankam po sami izvedbi implementacije. Na takšen način bi podjetje samo informacijsko rešitev AdInsure prodalo v večjih količinah brez povečanja stroškov, katere prinaša povečano število zaposlenih, raziskave specifičnih zahtev lokalnih trgov, lokalno trženje in prodajo implementacijskih storitev.

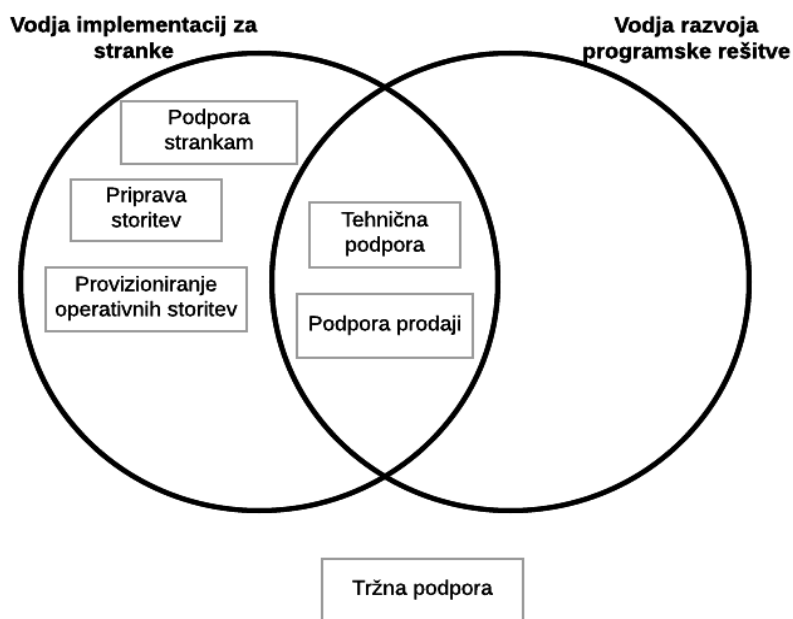
Naloga prodaje in vodstva podjetja je v tem, da vpliva na razvoj tako, da se bo razvijalo informacijsko rešitev s ciljem zaključene celote funkcionalnosti katera bo ustrezno tehnično dokumentirana. Takšno rešitev je potem mogoče brez poglobljenega tehničnega

znanja o samih tehničnih rešitvah prodajati prek partnerjev, ki implementirajo zahteve iz specifik posamezne stranke, znanje o sami informacijski rešitvi pa pridobivajo iz kvalitetne tehnične dokumentacije, ki se mora ujemati z resničnim delovanjem informacijske rešitve AdInsure.

4.8 Izvajanje funkcije podpore in storitev

V primeru informacijske rešitve AdInsure obravnavamo v okviru podpore in storitev vse aktivnosti, ki se izvajajo nad samo rešitvijo. V tem pogledu so tako vsi implementacijski projekti za prilagoditve in razširitve stranka za informacijsko rešitev AdInsure. Na podlagi tega bi lahko sklepali, da je naloga managerja produkta programske opreme, da vzpostavi in v celoti skrbi za podporo v okviru informacijske rešitve. Vendar zunanji pogled prek implementacijskih projektov za stranke prikaže potrebo po podpori in storitvah za rezultate implementacijskih projektov.

Slika 13: Razdelitev izvajanja aktivnosti v okviru funkcije podpore in storitev za informacijsko rešitev AdInsure



V primeru informacijske rešitve AdInsure in njenih implementacij se je ob reorganizaciji v aprilu 2015 uvedla tudi ekipa za podporo, katere glavni namen je izvajanje podpore strankam, ki že uporabljajo implementacijo na osnovi informacijske rešitve Adinsure v produkciji. Ekipa za podporo organizacijsko spada v oddelek implementacij za stranke. Skupina v praksi izvaja prvi in drugi nivo podpore strankam ter komunicira in predaja programske zahtevke vzpostavljenim implementacijskim timom za posamezno stranko. Hkrati je skupina zadolžena za preverjanje kvalitete implementiranih funkcionalnosti tako za posamezne implementacije za stranke kot za funkcionalnosti, ki so del osnovne

informacijske rešitve. Na Sliki 13 je prikazana razdelitev aktivnosti podpore in storitev, katero izvaja oddelek implementacij preko samih implementacijskih projektov ter s pomočjo ekipe za podporo, ki deluje znotraj oddelka. Ti tako izvajajo aktivnosti **podpore strankam, pripravo storitev** in prek implementacijskih projektov tudi **provizioniranje storitev**.

Po modelu Kittlausa in Clougha, opisanem v poglavju 1.4.2, se manager produkta programske opreme primarno ukvarja s samim produktom, kar v funkciji podpore in storitev za informacijsko rešitev AdInsure pomeni, da bi moral poskrbeti tudi za podporo in storitve za samo informacijsko rešitev. To v trenutnih v procesih dela manjka.

Uvedba samostojne ekipe podpore in storitev informacijsko rešitev AdInsure

Za povečanje skalabilnosti prodaje, katero bi dosegli z uvedbo prodaje prek partnerjev, je potrebna sprememba tudi na drugih področjih. Za podporo partnerjem je treba ustrezno vzpostaviti in izobraziti ekipo, ki bo izvajala prva dva nivoja podpore za samo informacijsko rešitev. Ekipa bi hkrati predstavljala podporo implementacijskim projektom, ki se izvajajo znotraj podjetja. Ni namreč smiselno, da se razvojna ekipa informacijske rešitve ukvarja z vsakim tehničnim in funkcionalnim vprašanjem, ki se pojavi tekom izvajanja projekta implementacije za stranke.

Trenutno stanje kaže, da se z izvajanjem podpore in storitev na nivoju informacijske rešitve nihče ne ukvarja oziroma je to prepuščeno oddelku za implementacije, ki nudi podporo za implementacije same, vprašanja in javljene težave pa prehitro pristanejo v razvojni ekipi. Smiselno je, da se z uvedbo managementa produkta programske opreme organizira ustrezno tehnično ekipo, ki bo zbiral in analiziral javljene napake, ki pridejo s strani implementacijskih projektov in jim določala prioriteto reševanja. Ekipa mora biti tehnično dovolj izurjena, da je sposobna pripravljati primere demonstracijskih rešitev za prodajo in trženje ter ostalo tehnično in vsebinsko podporo za ta področja. Sposobna mora biti tudi za reševanje enostavnejših težav in vprašanj, ki so jim posredovana. S tem bi funkcijo, ki jo trenutno izvaja ekipa za podporo v okviru oddelka implementacij, prenesli na to ekipo. Hkrati bi ekipa lahko bila zadolžena za izvajanje kontrole kvalitete oziroma testiranja informacijske rešitve, ki naj bi se po modelu sicer izvajala v okviru razvoja. Vendar, ker v primeru Adacte ne gre za izvajanje podpore več produktom na nivoju zavarovalništva, je smiselno, da ekipa za podporo deluje v obliki podaljška razvojne ekipe informacijske rešitve.

4.9 Temeljne aktivnosti po vlogah v poslovni enoti Zavarovalništvo

Pregled načina dela v poslovni enoti Zavarovalništvo, ki smo ga v tem poglavju povezali s funkcijami in aktivnostmi, definiranih v modelu po Kittlausu in Cloughu, je pokazal na delno ujemanje izvajanih aktivnosti z aktivnostmi, opredeljenimi v samem modelu. Prav

tako smo za vsako funkcijo posebej ugotovili, da se te aktivnosti izvajajo po različnih vlogah v podjetju.

V Tabeli 4 so prikazane temeljne aktivnosti, katere Kittlaus in Clough (2009, str. 56) prištevata med temeljne aktivnosti managerja produkta programske opreme, ter njihova trenutna razdelitev po vlogah v podjetju. Poleg tega so v tabeli prikazane tudi temeljne aktivnosti, ki se v podjetju ne izvajajo oziroma jih izvajajo zunanje organizacije za celotno tržišče in se v poslovni enoti Zavarovalništvo te rezultate interpretira.

Tabela 4: Izvajanje temeljnih aktivnosti po vlogah v podjetju za informacijsko rešitev AdInsure

Vloga v podjetju / Funkcija v modelu	Direktor poslovne enote Zavarovalništvo	Vodja razvoja informacijske rešitve	Vodja implementacij za stranke	Ni izvajano ali izvajano izven podjetja
Analiza trga	Analiza konkurenčnosti			Ocena tehnologije
Produktna analiza			Učinkovitost produkta Zadovoljstvo strank	
Strategija produkta	Pozicioniranje produkta Model distribucije Poslovni primer Pravni pogoji Management portfelja Pravni pogoji			Izdelati samostojno kupiti Model definiranja cen Zaščita intelektualne lastnine
Planiranje produkta	Dolgoročni načrt razvoja produkta Funkcionalne specifikacije Management zahtev	Funkcionalne specifikacije Plan verzije Management zahtev		

Iz zapisanega ugotovimo, da se temeljne aktivnosti po modelu Kittlausa in Clougha večinoma izvajajo. Izjema so aktivnosti, ki v primeru informacijske rešitve AdInsure trenutno niso toliko pomembne zaradi narave same rešitve in strategije prodaje. Informacijska rešitev se namreč samostojno brez izvajanja storitev dodelav in prilagoditev

ne prodaja, zato je del teh neizvajanih aktivnosti zajet ob pripravi pogodbe za prodajo informacijske rešitve skupaj s storitvami.

Velik del temeljnih aktivnosti izvaja direktor poslovne enote Zavarovalništvo, del katerih pa si deli z vodjo razvoja informacijske rešitve.

Direktor poslovne enote Zavarovalništvo je po notranjem dogovoru in poimenovanju v podjetju poslovni lastnik (angl. *product owner*) informacijske rešitve AdInsure, kar Fricker (2012) opisuje kot drug izraz za managerja produkta. Zaradi tega je razumljivo, da se njegove aktivnosti v večji meri ujemajo z aktivnostmi po modelu Kittlausa in Clougha.

Smiselno je razmisliti o vzpostavitvi ločene vloge managerja produkta programske opreme od vloge direktorja poslovne enote. Obseg zadolžitev direktorja poslovne enote je precej širok, hkrati pa se v določenih situacijah lahko znajde z nasprotujočimi si interesi.

4.10 Predlagane spremembe v organizacijski strukturi

Z vzpostavitvijo ločene vloge managerja produkta programske opreme bi dosegli bolj fokusirano izvajanje aktivnosti, povezanih z osnovno informacijsko rešitvijo, s katero bi ta strmel k cilju, da informacijska rešitev postane bolj stabilna, samostojna in bolj neodvisna od vpliva specifičnih zahtev implementacij za posamezne stranke.

Pozicija managerja produkta programske opreme bi morala biti ločena od hierarhije poslovne enote, da bi s tem zagotovili enakovredni nivo komunikacije med direktorjem poslovne enote, katerega cilj je tudi prodati čim več storitev in pridobiti čim več novih strank. Cilj čim večje prodaje za vsako ceno namreč vpliva na nivo kvalitete informacijske rešitve, kar za interes stabilnosti in vzdržnosti informacijske rešitve na dolgi rok ni sprejemljivo. V primeru podrejenosti managerja produkta pod direktorja poslovne enote komunikacija in konflikt, ki sta v takih primerih pričakovana, ne bi bila ustrezno rešena.

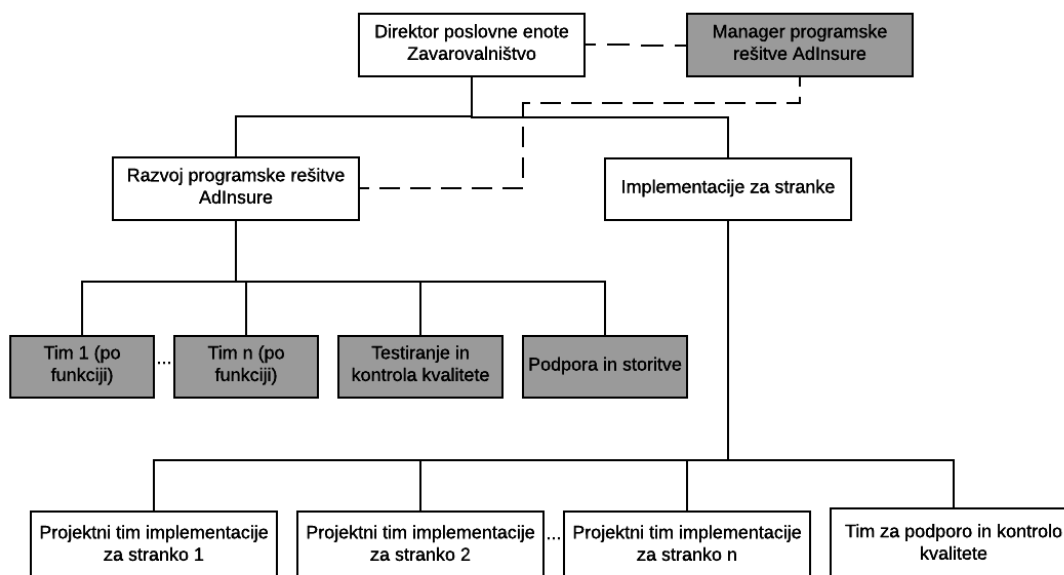
Predlagana druga sprememba v organizacijski strukturi je razdelitev ekipe za razvoj informacijske rešitve v več ločenih timov z določenimi projektnimi vodji. Timi naj bodo organizirani glede na vsebinska področja, katere informacijska rešitev podpira. Abbot in Fischer (2015, str. 44) namreč pravita, da je treba poiskati time ustrezne velikosti, kar sicer ustreznost definirata skozi različne faktorje, vendar predlagata, da tim naj ne bi bil večji od 15 oseb. S tem se bo izboljšala razdelitev in prevzemanje odgovornosti, za katere bodo timi zadolženi, izboljšala naj bi se tudi komunikacija med posamezniki in med timi.

Na Sliki 14 je predlagana spremenjena organizacijska struktura, na kateri so temno sivo obarvane dodane organizacijske strukture.

S predlaganimi spremembami bi bila usmerjenost k dolgoročni stabilnosti informacijske rešitve boljša. Z uspešnim delom managerja produkta programske opreme in ustrezno podporo vodstva podjetja bi informacijsko rešitev lahko pripeljali do te faze, da bi bilo

rešitev mogoče prodajati prek partnerjev, ki bi s svojim poznavanjem lokalnega okolja lažje izvajali storitve prilagoditev in razširitev po meri končne stranke. S tem pa bi dosegli povečanje skalabilnosti prodaje same informacijske rešitve.

Slika 14: Predlagana sprememba organizacijske strukture



SKLEP

Ugotovili smo, da štejemo tako poslovni model razvoja in prodaje produktov programske opreme kot model masovnih prilagoditev med skalabilne poslovne modele. Za to, da je rast prodaje resnično hitrejša kot naraščajo stroški, je treba poskrbeti z ustrezno kvalitetno informacijsko rešitvijo. Za informacijsko rešitev AdInsure, ki je osnova za izvedbo implementacij po meri strank, velja, da mora v čim večji meri izpolnjevati zahteve strank, hkrati pa mora biti dovolj generična, da se ista funkcionalnost uporabi v čim večjem obsegu v implementaciji za stranko. Poleg tega mora biti rešitev ustrezno dokumentirana in ne pretirano tehnično zahtevna za namestitve in uporabo, saj se s tem omogoči vključitev lokalnih implementacijskih partnerjev, ki bi bili sposobni analizirati zahteve strank na lokalnih trgih ter te implementirati brez prekomernega vključevanja razvojnega oddelka informacijske rešitve. Tako bi na način prodaje licenc informacijske rešitve dosegli večjo skalabilnost prodaje in s tem podpirali skalabilen poslovni model. Partnerji pa bi izvajali storitve prilagoditev in razširitev za stranke.

Vpeljava koncepta managementa produkta programske opreme pri tem vsekakor pomaga. Njegov glavni cilj je namreč obdržati vrednost produkta skozi čim daljše obdobje. Pozicija managerja produkta programske opreme, katerega delo ni lahko in enostavno, mora dobiti ustrezno težo od vodstva podjetja. Koncept sam pričakuje in opozarja na možnost pogostih konfliktnih situacij z različnimi oddelki, ki vplivajo na produkt v njegovem življenjskem

ciklu. S poudarkom na pomembnosti te pozicije je delo managerja produkta programske opreme lažje, s tem pa tudi njegov vpliv, s katerim je življenjski cikel produkta bolj kontroliran.

V primeru informacijske rešitve AdInsure smo ugotovili, da se aktivnosti, ki jih opravlja direktor poslovne enote Zavarovalništvo, najbolj ujemajo s temeljnimi aktivnostmi v funkcijah modela managementa produkta programske opreme po Kittlausu in Cloughu. Vendar istočasno izvaja aktivnosti, ki jih model predvideva kot aktivnosti, katere manager produkta programske opreme orkestrira, vendar jih sam aktivno ne izvaja. Ujemanja in odstopanja med izvajanimi aktivnostmi smo podrobneje predstavili in na podlagi teh ugotovitev prišli do sklepa organizacijskih sprememb.

Z uvedbo pozicije managerja produkta programske opreme, ki bi prevzela del aktivnosti in nalog z vloge direktorja poslovne enote, ki trenutno deluje kot poslovni lastnik produkta, bi dosegli večjo usmerjenost v izboljšanje same informacijske rešitve in njeno dolgoročno vzdrževanje vrednosti. Direktor poslovne enote namreč v svojih vlogah lahko deluje v konfliktu med interesi za:

- kvaliteto same informacijske rešitve in njeno usmerjenostjo k ciljem dolgoročnega načrta razvoja produkta;
- uspešnost in profitabilnost projektov implementacij za posamezne stranke;
- povečevanje prodaje.

Z uvedbo pozicije managementa produkta programske opreme bi dosegli večjo usmerjenost v samo informacijsko rešitev. Manager produkta programske opreme bi ob upoštevanju aktivnosti, ki jih ta koncept zanj predvideva, poskrbel za:

- dolgoročni načrt produkta razvoja produkta ter njegovo sledenje in evalvacijo, posledično pa boljšo usmerjenost;
- določitev vizije in strategije produkta, s pomočjo katere je določanje prioritet lažje, predvidljivost pričakovanih sprememb pa večja;
- validacijo in kontrolo kvalitete pred izdanimi verzijami znotraj razvojnega oddelka;
- bolj jasno razmejitev med informacijsko rešitvijo in implementacijami ter s tem možnost za vključitev zunanjih partnerjev v prodajni proces.

V magistrskem delu smo se omejili na dva modela predvidenih aktivnosti in področij, katera izvaja in pokriva manager produkta programske opreme. Ta dva dobro vzorčno predstavljata področje dela, situacij in problemov, ki se ob tem pojavljajo. Sam opis aktivnosti smo omejili na splošni opis. V analizi obstoječih procesov dela za informacijsko rešitev AdInsure pa smo se omejili na ujemanja in razlike v aktivnostih za model po Kittlausu in Cloughu. V prihodnje bi bilo treba na področju te informacijske rešitve raziskati procese dela v samem razvoju informacijske rešitve in poiskati možnosti za

spremembe, s čimer bi dosegli takšno obliko rešitve, ki bi omogočala spremembo v prodajni strategiji s prodajo prek partnerjev.

LITERATURA IN VIRI

1. Abbot, M.L. & Fischer, M.T. (2015) *The Art of scalability: Scalable Web Architecture, Processes and Organizations for the Modern Enterprise*. New Jersey: Pearson Education.
2. Adacta, d.o.o. (2009). *Pravilnik o sistemizaciji delovnih mest v podjetju Adacta, d.o.o.*(interno gradivo). Ljubljana: Adacta, d.o.o.
3. Adacta, Adacta's News Centre. Najdeno 2. julija. 2016 na spletnem naslovu <http://www.adacta-group.com/news-center/celent-adinsure-is-a-modern-and-attractive-solution-with-exceptional-customer-feedback>
4. Adacta, O Adacti. Najdeno 25. junija. 2016 na spletnem naslovu <http://www.adacta.si/about#about>
5. Baden-Fuller, C., & Haefliger, S. (2013). Business models and technological innovation. *Long range planning*, 46(6), 419–426.
6. de la Barra, C. L., Galdames, S., Crawford, B., Soto, R., & Crawford, K. (2015). Leadership in Agile Software Development Methods. In *HCI International 2015-Posters' Extended Abstracts* (pp. 154–158). Springer International Publishing.
7. Beck, K., Beedle, M., van Bennekum, A., Cockburn, A., Cunningham, W., Fowler, M., Grenning, J. Highsmith, J., Hunt, A., Jeffries, R., Kern, J., Marick, B., Martin, R.C., Mellor, S., Schwaber, K., Sutherland, J. in Thomas, D. (2001, februar). *Principles behind the Agile Manifesto*. Najdeno 20. maja. 2016 na spletnem naslovu <http://www.agilemanifesto.org/principles.html>
8. Bondi, A. B. (2000). Characteristics of scalability and their impact on performance. *WOSP '00 Proceedings of the 2nd international workshop on Software and performance* (str. 195–203). New York.
9. Cusumano, M. (2003). Finding your balance in the products and services debate. *Communications of the ACM*, 46(3), 15–17.
10. Cusumano, M. A. (2008). The Changing Software Business: Moving from Products to Services. *Computer*, 41(1), 20–27.
11. Davis, A.M. (2005). *Just Enough Requirements Management: Where Software Development Meets Marketing*. New York: Dorset House Publishing.
12. Ebert, C.(2006). The Impacts of Software Product Management. *The Journal of Systems and Software*, 80(6), 850–861.
13. Ebert, C. & Brinkkemper, S. (2014). Software product management – An industry evaluation. *The Journal of Systems and Software*, 95, 10–18.
14. Entrepreneur - Scalability. Najdeno 4. aprila 2016 na spletnem naslovu <https://www.entrepreneur.com/article/243237>
15. The Economic Times - Scalability. Najdeno 20. maja. 2016 na spletnem naslovu <http://economictimes.indiatimes.com/definition/scalability>

16. Fricker, S.A. (2012). Software Product Management. A. Maedche, A. Botzenhardt, L. Neer (ur.), *Software for People: Fundamentals, Trends and Best Practices* (str. 53–81). Heidelberg: Springer.
17. Gorchels, L. (2006). *The product manager's handbook*. New York: McGraw-Hi.
18. *Growth vs. Scaling* Najdeno 20. maja. 2016 na spletnem naslovu <https://www.fundable.com/blog/post/growth-vs-scaling>
19. Hauer, D. (2013). *Analiza primernosti razvojne metodologije informacijskih sistemov SCRUM za izbrano razvojno ekipo* (magistrsko delo). Ljubljana: Ekonomska fakulteta.
20. Joung, S. (23. 9. 2015). Would You Make It on Shark Tank? The Importance of Scalable Business Models. *The Business*. Najdeno 22. 5. 2016 na spletnem naslovu <http://www.business.com/business-opportunities/the-importance-of-scalable-business-models/>
21. Kilpi, T. (1997). Product management challenge to software change process: Preliminary results from three SMEs experiment. *Software Process Improvement and Practice*, 3(3), 165–175.
22. Kittlaus, H., & Clough, P. N. (2009). *Software Product Management and Pricing : Key Success Factors for Software Organizations*. Berlin: Springer.
23. Kotler, P., & Armstrong, G. (2012). *Principles of marketing*. Upper Saddle River, New Jersey: Pearson Prentice Hall.
24. Kostoff, R.N. & Schaller, R. R. (2001). Science and Technology Roadmaps. *IEEE Transactions on Engineering Management*, 48(2), 132–143.
25. Krajnc, A. (2011). *Model za vrednotenje pridobitev uporabe programskih tovarn pri razvoju informacijskih rešitev* (doktorska disertacija). Maribor: Fakulteta za elektrotehniko, računalništvo in informatiko.
26. Kuo, T. C. (2013). Mass customization and personalization software development: a case study eco-design product service system. *Journal of Intelligent Manufacturing*, 24(5), 1019–1031.
27. Maglyas, A., Nikula, U. & Smolander, K. (2011). What do we know about software product management? - a systematic mapping study. *2011 Fifth International Workshop on Software Product Management (IWSPM)* (str. 26–35). Trento.
28. McGrath, M. (2001). *Product strategy for high technology companies*. McGraw Hill Professional.
29. Porter, M. E., Goold, M., & Luchs, K. (1996). From competitive advantage to corporate strategy. *Managing the multibusiness company: Strategic issues for diversified groups* (str. 285–314).
30. Rajlich, V. T. & Bennett, K. H. (2000). A staged model for the software life cycle. *Computer*, 33(7), 66–71.
31. Regnell, B., Brinkkemper, S. (2005). Market-Driven Requirements Engineering for Software Products. V A.Aurum & C. Wohlin (ur.), *Engineering and Managing Software Requirements* (str. 287–308), Berlin: Springer Verlag.

32. Sneed, H.M., Hasitschka, M., Teichmann, M. T. (2004). *Software-Produktmanagement - Wartung und Weiterentwicklung bestehender Anwendungssysteme*. dpunkt.verlag.
33. Stark, J. (2011). *Product lifecycle management: 21st Century Paradigm for Product* (2nd Edition). London: Springer.
34. Patrashkova-Volzdoska, R. R., McComb, S. A., Green, S. G., & Compton, W. D. (2003). Examining a curvilinear relationship between communication frequency and team performance in cross-functional project teams. *IEEE Transactions on Engineering Management*, 50(3), 262–269.
35. Vilet, H. V. (2007). *Software Engineering: Principles and Practice*. Willey.
36. Van de Weerd, I, Brinkkemper, S., Nieuwenhuis, R., Versendaal, J., Bijlsma, L. (2006). A Reference Framework for Software Product Management. Utrecht University. Najdeno 16. maja 2016 na spletnem naslovu <http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.81.8849&rep=rep1&type=pdf>
37. Wikipedia-DevOps. Najdeno 30. junija 2016 na spletnem naslovu <https://en.wikipedia.org/wiki/DevOps>
38. Xu, L., Brinkkemper, S. (2007). Concepts of product software. *European Journal of Information Systems*, 16(5), 531–541.