

Integracija informacijskog sustava i interneta stvari

Sokač, Danijel

Master's thesis / Diplomski rad

2018

Degree Grantor / Ustanova koja je dodijelila akademski / stručni stupanj: **University of Zagreb, Faculty of Organization and Informatics / Sveučilište u Zagrebu, Fakultet organizacije i informatike**

Permanent link / Trajna poveznica: <https://um.nsk.hr/um:nbn:hr:211:954567>

Rights / Prava: [Attribution 3.0 Unported](#) / [Imenovanje 3.0](#)

Download date / Datum preuzimanja: **2024-05-24**



Repository / Repozitorij:

[Faculty of Organization and Informatics - Digital Repository](#)



SVEUČILIŠTE U ZAGREBU
FAKULTET ORGANIZACIJE I INFORMATIKE
V A R A Ž D I N

Danijel Sokač

Integracija informacijskog sustava i
interneta stvari
DIPLOMSKI RAD

Varaždin, 2018.

SVEUČILIŠTE U ZAGREBU
FAKULTET ORGANIZACIJE I INFORMATIKE
V A R A Ž D I N

Danijel Sokač

Matični broj: 44485/15-R

Studij: Informacijsko i programsko inženjerstvo

**Integracija informacijskog sustava i
interneta stvari**

DIPLOMSKI RAD

Mentor:

izv.prof.dr.sc. Ruben Picek

Varaždin, rujan 2018.

Sadržaj

1. Uvod	1
2. Poslovanje i digitalna transformacija	3
2.1. Pojam i definicija.....	5
2.2. Tehnologije digitalne transformacije.....	5
2.2.1. <i>Big Data</i> i analiza	6
2.2.2. Autonomni roboti	7
2.2.3. Simulacija.....	8
2.2.4. Horizontalna i vertikalna integracija sustava	8
2.2.5. Industrijski Internet stvari	8
2.2.6. Kiber sigurnost	9
2.2.7. Računalstvo u oblaku	10
2.2.7.1. Modeli računalstva u oblaku	10
2.2.7.2. Servisi računalstva u oblaku.....	11
2.2.8. Aditivna proizvodnja.....	12
2.2.9. Proširena stvarnost	12
2.3. Potreba digitalne transformacije u poslovanju	13
3. Internet stvari.....	16
3.1. Pojam i definicija.....	16
3.2. Postojeća znanja i trenutna istraživanja.....	16
3.3. Arhitektura.....	17
3.3.1. MLN model	17
3.3.2. SOA model	18
3.3.2.1. Sloj osjeta	19
3.3.2.2. Sloj mreže.....	19
3.3.2.3. Sloj servisa	20
3.3.2.4. Sloj sučelja	20
3.4. Internet stvari u poslovanju	21
3.5. Područja primjene.....	22
3.5.1. Pametna kuća.....	23
3.5.1.1. Definicija izraza „pametna kuća“.....	24
3.5.2. Pametni grad.....	25
3.5.2.1. Karakteristike pametnih gradova	26
3.5.2.2. Veza između Pametne mreže i Pametnog grada	28
3.5.3. Pametna poljoprivreda.....	30

4. Poslovni informacijski sustav i tehnološki aspekti.....	33
4.1. Pojam i definicija.....	33
4.2. Prednosti i mane	34
4.3. Povijest ERP-a.....	36
4.4. Arhitektura i moduli	40
4.4.1. Implementacija drugog stupnja	41
4.4.2. Klijent/server implementacija trećeg stupnja	41
4.4.3. Moduli ERP-a.....	42
4.5. Faze implementacije ERP-a	44
4.5.1. Faza odluke - Prelazak na ERP	48
4.5.1.1. Prepoznavanje inovacije.....	50
4.5.1.2. Pokretanje implementacije	52
4.5.2. Predimplementacijska faza.....	53
4.5.2.1. Odabir ERP-a	53
4.5.2.2. Projektno planiranje	57
4.5.2.3. Dizajn i prilagodba	58
4.5.3. Implementacijska faza	58
4.5.3.1. Instalacija ERP-a	58
4.5.3.2. Konfiguracija ERP-a	59
4.5.3.3. Čišćenje i priprema podataka	59
4.5.3.4. Testiranje ERP-a	59
4.5.3.5. Službena objava.....	59
4.5.4. Postimplementacijska faza (životni ciklus ERP-a).....	59
4.5.4.1. Upoznavanje ERP-a	59
4.5.4.2. Korištenje ERP-a.....	59
4.5.4.3. Poboljšavanje ERP-a.....	59
4.5.4.4. Otkazivanje ERP-a	60
4.6. Strategije implementacije ERP sustava	60
4.6.1. Veliki prasak.....	61
4.6.2. Fazna implementacija.....	62
4.6.3. Organizacijske karakteristike i tradicionalne strategije implementacije	63
4.6.4. ASAP (eng. <i>SAP's Accelerated SAP</i>)	67
4.6.5. OUM (eng. <i>Oracle Unified Method</i>).....	68
4.6.6. Microsoft Dynamics' Sure Step	69
4.7. Trendovi u poslovanju	70
4.7.1. Cloud ERP	71
4.7.2. Primjeri korištenja IoT-a u ERP-u.....	72

4.8. Važnost ERP-a u poslovanju	74
5. Analiza procesa integracije Internet stvari-a u ERP	75
5.1. Važnost integracije Internet stvari-a u ERP u poslovanju.....	75
5.2. Analiza znanstvenih izvora.....	76
5.3. Prijedlog procesa integracije	78
6. Praktični primjer – Integracija IoT modula na Azure i ERP (Microsoft Dynamics NAV). 81	
6.1. Opis i konceptualizacija	81
6.2. Korišteni alati	81
6.3. Mogućnosti realizacije.....	82
6.4. Scheme i opisi izvedbe IoT modula	84
6.5. Spremanje podataka.....	84
6.6. Priprema Azure-a.....	86
6.7. Programiranje IoT modula i slanje podataka u <i>Cloud</i>	86
6.8. Podešavanje povezivanja <i>Azure</i> -a sa <i>Power BI</i> -om	97
6.9. Podešavanje povezivanja <i>Azure</i> -a, <i>Power BI</i> -a i <i>Microsoft Dynamics NAV</i> -a.....	97
6.10. Integracija modula u IS	99
6.11. Zaključna analiza	110
7. Zaključak	112
8. Literatura	113

1. Uvod

Internet stvari (IoT – eng. *Internet of Things*) je dostupan i okružuje nas svakodnevno. Regulator temperature u kući, u novije vrijeme: hladnjaci, perilice za odjeću, tosteri i slično. Sve to sadržava koncept interneta stvari. Također bi se moglo reći da su avioni puno ranije koristili slični koncept tokom praćenja letova sa radarom i upravljanja zračnim prometom. Također postoji i tzv. „*Smart city*“ koncept koji se odnosi na očitavanje senzora u svrhu obavijesti o broju slobodnih parkirnih mjesta na parkiralištima. Senzori pokreta isto mogu biti dio koncepta IoT-a, primjerice: rotirajuća vrata, klizna vrata, razni likovi poput Djeda Božićnjaka i slično... Jedna od najčešćih primjena IoT je postavljanje protupožarnih alarma koji su zapravo senzori za očitavanje dima. IoT je prisutan i u poslovnom svijetu.

Tako IoT može biti i postati jako dobro sredstvo u postizanju boljeg poslovnog cilja. Može se koristiti kao sredstvo provođenja koncepta poslovne inteligencije (BI - eng *Business Intelligence*) i poslovne analize (BA – eng. *Business Analytics*). Ta dva koncepta su ključna za maksimiziranje uspješnosti poslovanja i često IoT može tome najvećim dijelom pridonijeti. Koncepti se mogu također koristiti i u razvoju tzv. „pametne poljoprivrede“ te pridonijeti povećanju efikasnosti proizvodnje. Bitno je napomenuti da IoT nije jedini koji može poboljšati sam poslovni rezultat. Uz sve to, aktualna transformacija, kroz koju prolaze organizacije trenutno, naziva se digitalna transformacija koja je još poznata pod nazivom Industrija 4.0 (eng *Industry 4.0*). Digitalna transformacija je veliki korak kroz koji organizacije moraju proći da bi ostale konkurenti u ovom razdoblju uz rapidni razvoj digitalnih tehnologija koje mogu upravo tome pridonijeti. Tema digitalne transformacije je obrađena u detaljnijem tonu jer je aktualna i važna za organizacije. Glavni cilj ovog rada je integrirati IoT u informacijski sustav te pokazati kako se može doprinijeti poslovanju.

Kao što je najavljeno, objasniti će se digitalna transformacija u detaljnijem tonu. Unutar te teme dotaknuti će se DESI indeks koji služi za mjerenje stupnja digitalizacije, koncepti digitalnih poduzeća i pametnih gradova, strategija digitalne transformacije, okvir digitalne transformacije. Sljedeća tema će biti koncept Interneta stvari odnosno kraće: IoT. Nakon toga će se izraditi teoretska slika o važnim aspektima IoT-a. Predstaviti će se, ukratko, koncepti IoT-a. U nastavku, objasniti će se ERP kao poslovni IS i njegovi tehnološki aspekti. Kratko će se osvrnuti na razvoj ERP-a od početaka do sadašnjosti i detaljnije će se obraditi Cloud ERP te primjena IoT tehnologija u ERP-u. Sljedeće će se dotaknuti tema važnosti integracije IoT sustava u informacijski sustav i kako on utječe na samo poslovanje tvrtke. Naposljetku ima praktični primjer koji će prikazati prototip rada te integrirati ga u informacijski sustav. Praktični

primjer će obuhvatiti izradu IoT modula te sakupljanje podataka u online sustavu te korištenje tih podataka u informacijskom sustavu. U ovom dijelu će se prikazati cijeli praktični dio diplomskog rada.

2. Poslovanje i digitalna transformacija

Preteča industriji je ceh. Ceh je jednak današnjem obrtu. To je skupina ljudi koja radi isti posao. S vremenom su se cehovi počeli udruživati i tako na pozornicu nastupaju prve tvornice. Prve tvornice bile su temeljene na ručnom radu. Te prve tvornice, rekli bismo, su počeci industrije. Uz samu podjelu rada dolazi do efikasnog korištenja ljudskih resursa i proizvodnje proizvoda. U ovom trenu već se može pričati o industriji. U drugoj polovici 18. stoljeća javlja se Prva industrijska revolucija. Prva industrijska revolucija se temelji na pojavi parnog pogona. Druga polovica 19. stoljeća značajna je za pojavu Druge industrijske revolucije. Glavni izumi u to vrijeme bili su nafta (1859.), električna žarulja (Thomas Alva Edison, 1879.) i motor sa izmjeničnom strujom (Nikola Tesla, 1887.). Glavni rezultati ove industrijske revolucije je pojava filma, razvoj zrakoplovnog prometa, upotreba nafte kao gorivo u automobilima. Zanimljivo je shvatiti da se nafta javila kao prekretnica u prošloj industrijskoj revoluciji, ali kako nije bilo alata koji bi omogućili njezinu efikasnu primjenu, ona nije doživjela značajno korištenje. Međutim s pojavom motora na izmjeničnu struju, pronašao se način iskorištavanja nafte koji se koristi do danas. 1990-tih godina se pojavila Treća industrijska revolucija odnosno Digitalna revolucija. Ova revolucija je obilježena razvojem digitalne tehnologije. Treća industrijska revolucija traje i dan danas, glavna značajka ove industrijske revolucije je pojava računala izgrađenih od električnih komponenti i prekid korištenja analogno mehaničkih i elektronskih tehnologija.

Treća industrijska revolucija je dala dobar zamah industriji koja svakim danom zahtjeva sve kvalitetnije proizvode kako su spomenuli Brettel i sur (2014). Sam zahtjev za kvalitetnim proizvodom povlači i veći trošak proizvodnje. Česta prepreka u postizanju tog cilja je i vlada države u kojoj se nalazi pogon. Digitalna transformacija je jedna od većih povlastica zbog čega su vlasnici tvornica ciljali preseliti pogon u drugu zemlju kojoj je u interesu provesti tu transformaciju i na taj način su željeli osigurati svoju kompetentnost i zahtijevanu kvalitetu. Ubrzo su tvornice shvatile da kupcima nije u interesu plaćati visoku cijenu za inkrementalno poboljšanje i zbog toga su mnoge njemačke tvornice počele proizvoditi personalizirane proizvode. Sa tim potezom su granice tvornice srušene te su i kupci uključeni u lanac vrijednosti. Glavna pomoć u povećavanju kvalitete je kooperacija strojeva i ljudi te mogućnost planiranja proizvoda uz pomoć simulacije.

Kako je već napisano, svaka industrijska revolucija uzrokovana je pojavom neke tehničke inovacije koja je bila prekretnica u to vrijeme. Industrijske revolucije su počele sa vodom i vodenom parom da bi zadnja industrijska revolucija bila označena pojavom računala

odnosno pojavom programibilnih logičkih kontrolera (*eng. Programmable logic controllers - PLC*) čija je svrha bila automatizirati neke procese u tvornicama u 1970. godini. Predviđa se da će slijedeća industrijska revolucija biti pokrenuta zbog Interneta koji omogućuje komunikaciju između ljudi, a isto tako i komunikaciju između ljudi i strojeva. To isto navode i Brettel i sur (2014). Proizvodna industrija je važan dio ekonomije zemlje te je izuzetno važna ako onda većinskim udjelom sudjeluje u BDP-u zemlje i osigurava radna mjesta velikom broju radnika. U težnji da se sama proizvodnja nastavi i da bude kompetentna u svjetskoj trgovini, često se odlučuje seliti proizvodne pogone u zemlje sa nižim standardom. Taj potez uvelike utječe na sam proizvod koji se tamo nastaje, ali tvornice nastoje održati kvalitetu proizvoda na istoj razini u svim podružnicama proizvodnih pogona. Brettel i sur (2014) također ističu kako se zbog toga zemlje moraju usredotočiti na ublažavanje napetosti između ekonomije razmjera i opsega isto kao i planiranje i orijentacije vrijednosti. U ovo doba, masovna proizvodnja je glavna djelatnost većinskom broju tvornica. Kako bi se masovna proizvodnja uspješno izvela potrebno je također i uvesti standarde. Kako samo tržište raste, raste i potražnja za proizvodima. Mnoge tvornice, zbog toga, nisu vidjele budućnost u posvećivanju pažnje varijacijama proizvoda koje zanimaju korisnike. One su izostavljale takve proizvode u svrhu efikasnog povećanja produktivnosti. Međutim, kada više tvornica razmišlja na taj način, dolazi do zasićenja tržišta. Više sličnih ili istih proizvoda na tržištu je natjeralo tvornice da posegnu za strategijom personaliziranih proizvoda. Jasno je da ta strategija zahtjeva pomno planiranje prije nego tvornica uskoči u izvršavanje iste. U toj strategiji je izuzetno važno proizvoditi točno onoliko koliko treba, da se ne gomilaju zalihe u skladištu. S vremenom tržište postaje zahtjevnije pa se javlja i potreba da se sustavi restrukturiraju, a sama pomoć u tome je razvijen Internet i senzori koji mogu pomoći u tome i u praćenju proizvoda i proizvodnog procesa kroz njihov cijeli životni ciklus.

Jasno se nazire da prilazi novoj industrijskoj revoluciji. Ona je poznata pod nazivima četvrta industrijska revolucija ili industrija 4.0. Poznato je da organizacije moraju konstantno učiti na tržištu i osluškivati aktualne trendove te biti spremne pravovremeno odgovoriti na promjene koje im omogućuju da ostaju glavni akteri u svojem području. Želja za održavanjem kompetentnosti često rezultira sa povisivanjem standarda ili prihvaćanjem moderne tehnologije koja olakšava i poboljšava poslovanje. Industrijska revolucija uvijek sa sobom donosi neki novitet. Novitet u četvrtoj industrijskoj revoluciji je objašnjen u slijedećim poglavljima. Vidjet će se koje su to tehnologije koje su važne u novoj industrijskoj revoluciji.

2.1. Pojam i definicija

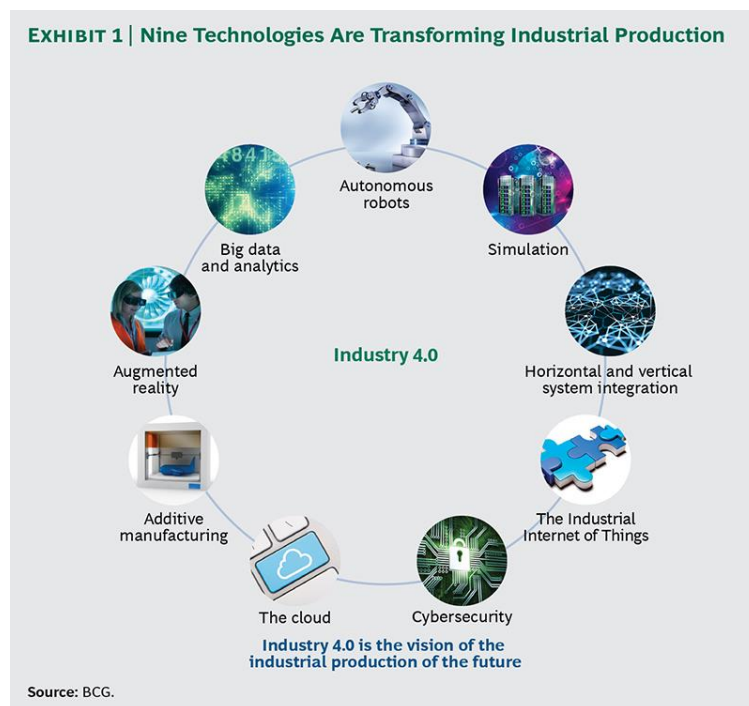
U tijeku je četvrta industrijska revolucija koja se temelji na 9 tehnoloških napredaka. U ovoj revoluciji će IT sustavi, senzori, strojevi i proizvodi biti povezani i izvan organizacije. Primjena Industrije 4.0 se često svodi na smanjenje troškova i efikasna proizvodnja. Uvođenjem ove revolucije potiče se da se tvornice šire i otvaraju nove pogone koji potpomognuti robotima rade svoj posao.

Uzevši u obzir prethodni odlomak, lako se zaključi da je Industrija 4.0 je finalno stanje digitalne transformacije, a digitalna transformacija je alat za postizanje tog stanja. Koristeći napisano, može se formirati definicija: Industrija 4.0 je konačno stanje procesa digitalne transformacije, a digitalna transformacija je proces učenja organizacije kako otkriti i koristiti nove tehnologije koje omogućuju komunikaciju među entitetima kroz internet mrežu u svrhu analize i poboljšanja sadašnjih procesa i budućeg poslovnog rezultata. Entiteti u ovoj definiciji mogu biti ljudi, strojevi, sustavi, proizvodi i slično. Globalno gledano, Industrija 4.0 je sljedeći korak u životnom vijeku organizacija. Koje su to nove tehnologije koje su utjecale na pojavu četvrte industrijske revolucije?

2.2. Tehnologije digitalne transformacije

Tehnologija koja omogućuje digitalnu transformaciju je već zastupljena i korištena u industriji. Pojava digitalne transformacije ona će prebaciti proizvodnju recimo. Digitalna transformacija će preobraziti proizvodnju te ju potpuno integrirati u automatiziran i optimiziran tijek proizvodnje koji vodi to novih veza između dobavljača, proizvođača i korisnika te između čovjeka i stroja.

Te tehnologije su bile prisutne u industriji i prije nove revolucije ali nisu sve organizacije na to reagirale. Prepoznao se potencijal i mogući smjer razvoja tehnologije u smjeru digitalne transformacije. Svrha tih tehnologija je komunikacija između ljudi i strojeva te ubrzavanje procesa proizvodnje. Kratko će se predstaviti sve tehnologije u slijedećim poglavljima.



Slika 2.2.1. Devet tehnologija digitalne transformacije.
(Izvor: Rüßmann i sur, 2015)

2.2.1. *Big Data* i analiza

Big data je jedna od uobičajenih tehnologija u Industriji 4.0. Navedena tehnologija se koristi za poboljšanje cijele organizacije, ali i kao pomoć u odlučivanju. Ova tehnologija je alat unapređivanja poslovanja organizacije. Svrha tehnologije je analiza velikog broja podataka i isporuka rezultata koji doprinosi organizaciji u većini slučajeva.

Poslovna inteligencija (*eng. Business intelligence*) i poslovna analiza (*eng Business analytics*) usko su vezane uz pojam Big Data i analizu istog. Trenutno se Big Data i analiza Big Data-e se koriste za opisivanje skupova podataka i tehnike analize u velikim (reda veličine od terabajta do eksabajta) i kompleksnim aplikacijama (od senzora do podataka sa društvenih medija) tako da zahtijevaju napredno i jedinstveno spremište podataka, menadžment, analizu i tehnologije vizualizacije. Poslovna analiza i poslovna inteligencija su spojene u pojam „*big data*“ analiza (*eng. big data analytics*).

Chen i sur (2012) objašnjavaju da je srž same poslovne inteligencije i analize sakupljanje podataka koje se temelji na tehnikama u području upravljanja bazom podataka. To predstavlja prvu fazu. Razvojem web 2.0 alata, započinje nova faza poslovne inteligencije i analize koja se temelji na analizi ponašanja i obrazaca korisnika na web aplikacijama. Podaci prikupljeni na ovaj način mogu dati pregled ponašanja korisnika na webu te se prepoznao

potencijal i Google je izradio svoj *Google analytics*. Treća faza se javlja pojavom mobitela i sve veći prijelaz na mobilne uređaje. U ovoj fazi se isto sakupljaju podaci kao i iz prethodne analize zajedno sa dodatnim podacima iz senzora itd.

Predstavivši ovu tehnologiju jasno je da koncepti poslovne analize i poslovne inteligencije su usko vezani uz *Big Data* i analizu tih podataka. U prethodnom poglavlju je opisan tijek razvoja poslovne inteligencije i analize do 2012. godine. Danas se već vidi utjecaj web 3.0 koncepta, a podaci sa mobilnih uređaja, kao što su lokacija, aplikacija, povijest pretraživanja i slično, se već poprilično dugo sakupljaju i procesiraju u svrhu ciljanja reklamama i personaliziranih ponuda za svakog pojedinačnog korisnika. Nije isključeno da se ti podaci koriste u svrhu zarade na korisnicima i prodaje istih drugim organizacijama što često rezultira mnoštvom neželjenih e-mailova u pretincu e-mail pošte. Opisan koncept korišten je u praktičnom primjeru ovog rada. Svakako, za više informacija lako se pronađu dodatni izvori na temu.

2.2.2. Autonomni roboti

Tvornice u industrijama koriste robote za zadatke koje čovjek ne može precizno i točno odraditi. Robotizacija je jedan od motiva automatizacije. Roboti se koriste kako bi povećali efikasnost procesa u kojem sudjeluju te kako bi surađivali sa ljudima. Oni postepeno evoluiraju pa i sami mogu surađivati međusobno. Rüßmann i sur (2015) navode robota XuMi koji je proizveden od ABB-a. YuMi je specijalno dizajniran robot koji sa dvije ruke i računalnom vizijom može sigurno surađivati s ljudima i prepoznavati okolinu.

U svijetu robota može se čuti i pojam inteligencija roja (eng. Swarm Intelligence). Navedeni pojam se uči na kolegiju Višeagentni sustavi. Inteligencija roja je distribuirano ponašanje pojedinaca koje uvjetuje funkcioniranje i ponašanje cijele grupe. Naime pojedina jedinka bez znanja o svim članovima grupe, osim onima u neposrednoj blizini, može bez problema funkcionirati i obavljati zadatke. Ovo je dobar primjer distribuiranog ponašanje i sa sobom nosi neke pogodnosti. U svijetu automatizacije, inteligencija roja omogućuje dodavanje nove jedinke bez da se mijenja cijeli sustav, sposobnost samoorganizacije jedinke i prilagodba u skladu sa trenutnom lokacijom i minimalistički dizajn. U kontekstu automatizacije, navode Ijspeert i sur, minimalistički dizajn bi značio minimalan napor u zadržavanju resursa za izračune, a cilja se na što je pametnije grupno ponašanje.

Glavna stvar oko inteligencije roja je da pojedine jedinke „roja“ ne vide ukupnu sliku onog što „grade“. Svaka jedinka komunicira sa svojim susjedom i koordinira svoje akcije prema

tome. To je jedan od zanimljivijih koncepata koji se lako može primijeniti u robotici. Baš kako je napisano u prethodnom poglavlju, svi autonomni roboti komuniciraju međusobno te su fleksibilni i promjenjivi. U ovom kontekstu je jasno da se ne govori o tome kako bi roboti zamijenili ljudsku radnu snagu. Upravo suprotno, oni bi pomagali i povećali efikasnost izvršavanja dodijeljenih zadataka te radili u interakciji sa ljudima.

2.2.3. Simulacija

Uz razvoj 3D modeliranja, razvija se i primjena simulacija. Simulacije postaju vrlo važan čimbenik prilikom pripreme proizvoda za produkciju kada se recimo radi o automobilima, strojevima i slično. Simulacija može predložiti korisniku određeni set ishoda i slučajeva. Simulacije se mogu koristiti i za postavljanje robota te drugih strojeva.

Simulacija je tehnologija pogodna za poboljšavanje rezultata procesa pa i procesa samih. Simulacijom se izrađuje virtualni model sa određenim rezultatima, koji se onda preslikava u stvarni sustav očekujući iste ili slične rezultate. Ovo je jedna od korisnijih tehnologija 4. industrijske revolucije.

2.2.4. Horizontalna i vertikalna integracija sustava

Rüßmann i sur (2015) navode problem nedovršene integracije informacijskog sustava u cijelost. Često se događa da su organizacija, dobavljači i korisnici loše povezani. Loša povezanost je prisutna i na drugim razinama organizacije. Industrija 4.0 omogućiti će čvršću povezanost među komponentama organizacije i tijekom sakupljanja informacija.

Sama organizacija funkcionira kao cjelina. Tako da bi organizacija trebala biti i realizirana kao jedna usko povezana cjelina u kojoj svi dijelovi komuniciraju međusobno u svrhu rješavanja aktualnih zadataka. Vrlo je ključno pratiti informacije tijekom cijelog životnog vijeka proizvoda. Od samog početka do izlaska iz organizacije. Također komunikacija između dobavljača i organizacije te organizacije i kupca je vrlo važna jer može smanjiti nepotrebne troškove i povećati sam poslovni uspjeh.

2.2.5. Industrijski Internet stvari

Industrijski Internet stvari (*eng. The Industrial Internet of Things*) pronalazi svoju primjenu u industriji. Koristi se kao poveznica među strojeva i tvorničkih senzora koji zajedno tvore jedan upravljački sustav tvorničkog procesa. Rüßmann i sur (2015) definiraju ovu komponentu digitalne transformacije. Svrha industrijskog Interneta stvari je uvesti

automatizaciju i decentralizaciju odlučivanja. Tako da se strojevi fokusiraju samo na okolinu koju vide i koja im treba, a ne na cijeli sustav u kojem postoje.

Internet stvari je novija tehnologija oko koje se vrte brojna istraživanja. Upotreba te tehnologije je u porastu. Korištenje Internet stvari u industriji je velika pomoć organizaciji zbog toga jer IoT omogućuje umrežavanje gotovo svih aktera u procesu proizvodnje i na taj način smanjuje pojavu pogreške i zastoja. Međutim, odlomak iznad ne opisuje što je Internet stvari, stoga će se to objasniti u slijedećem, istoimenom poglavlju.

2.2.6. Kiber sigurnost

Industrijska transformacija predstavlja IoT kao jedan dio četvrte industrijske revolucije. Problem koji se javlja je: IoT sam po sebi nije izuzetno zaštićeni i imuni na hakerske napade. Upravo suprotno, IoT predstavlja male rupice kroz koje haker može upasti u sustav i uzrokovati štetu ili olakšati slijedeći neovlašteni ulazak. Kako se najavljuje „Internet stvari“ što u doslovnom prijevodu znači da su stvari povezane na Internet. To znači da svaka ta stvar mora biti adekvatno zaštićena inače se javljaju problemi. Zbog toga cijela ova priča o kiber-sigurnosti. Postoje primjeri gdje su prikazani uspješna hakiranja automobila te kontroliranje istog sa udobnosti svoje sobe i kućne Internet veze. I bilo bi problematično da pećnica povezana na Internet odjednom umjesto na 200 stupnjeva počinje raditi na 400. Mnogi su uvidjeli problem u kiber-sigurnosti te su sklopili partnerstva sa firmama koje se bave kiber-sigurnošću.

Često sama sigurnost postaje mnogo više nego što se može zamisliti. U prodaji sa robotima mnogo je važno regulirati prava pristupa strojevima.. Sigurnost dobiva veći obujam te je korisno da tvrtke surađuju sa drugim tvrtkama koje se bave sigurnošću u svrhu učvršćivanja sigurnosti sustava tvrtke. Vrlo je korisno surađivati sa takvim tvrtkama, jer one mogu savjetovati i poboljšati uslugu. U nastavku će se kratko predstaviti kiber sigurnost.

Guiora (2017) navodi da je kiber sigurnost (*eng. cybersecurity*) ustrajnost u zaštiti informacija, komunikacija i tehnologija od štete uzrokovane slučajno ili namjerno. Ovaj pojam ne znači ništa drugo nego zaštita od hakerskog napada koji se obično naziva kiber napad. Glavna meta ovakvog načina napada je napad na povjerljivost, dostupnost i integritet, a obično putem ranjivosti dostupnih na hardveru ili softveru.

Prvo, razmotri se slijedeći primjer koji navodi Guiora (2017): Osoba pristupa bankovnom računu i utvrđuje da nedostaje sav novac. Također vidi pisma od agencija koje tvrde da posjeduju velike iznose novca. Da se zaključiti da je osoba žrtva krađe identiteta.

Pitanje je: Odgovara li to definiciji kiber sigurnosti? Jeste, jer zloupotreba nečijeg broja socijalnog osiguranja se suočava sa nedostatkom zaštite informacija od štete.

Kiber sigurnost je vrlo važna u svijetu IoT-a te se treba ozbiljno pristupiti toj tehnologiji. Primarni cilj je zaštititi korisnike i društvo od štete. Često se događa da su predstavnici štete hakeri koji pokušavaju prodrijeti u sustav tražeći rupe i sigurnosne propuste. Aplikacija mora biti detaljno razrađena i sigurnost implementirana u sve dijelove aplikacije kako bi se minimizirala mogućnost upada u sustav.

2.2.7. Računalstvo u oblaku

Dolaskom digitalne transformacije na prag, uočava se da je računalstvo u oblaku dobra prilika. Računalstvo bi tvrtkama omogućilo bolja reakcijska vremena i dostupnost informacija izvan granica organizacije.

Računalstvo u oblaku (*eng. cloud computing*) je sastavni dio 4. industrijske revolucije. Prednosti, koje ono donosi, očitavaju se i na strani programera i na strani korisnika. Uvođenje računalstva u oblaku miče se većina troškova koje bi programer imao da mora sam brinuti kako će postaviti servis i održavati ga aktivnog. Računalstvo u oblaku omogućuje prostor na kojem programeri zakupe prostor i objave servis, a trošak je puno manji nego kada bi sam brinuo o svemu. Korisnici, s druge strane, mogu nesmetano koristiti servise koji se nalaze u oblaku (*eng. cloud*). Bez toga, bi bilo puno teže otkriti i koristiti uslugu ili servis.

Do pojave ove tehnologije nije bilo moguće dijeliti programe i usluge preko interneta. Računalstvo u oblaku omogućuje da jedna usluga ili program bude dostupan globalno. Na taj način se smanjuju troškovi izrade aplikacije koja koristi tu uslugu i isto tako više razvojnih timova diljem svijeta može pristupiti toj usluzi. Također, računalstvo u oblaku je otvorilo mogućnosti za iznajmljivanje aplikacija odnosno usluga tako da autori mogu zaraditi objavljujući uslugu online.

2.2.7.1. Modeli računalstva u oblaku

Postoje nekoliko modela računalstva u oblaku koje spominju i objašnjavaju Raihana G. F. H. i College J. M. (2012), a biti će navedeni u nastavku:

- javno računalstvo u oblaku (*eng. Public Cloud Computing*) – Infrastruktura je dostupna za široku javnost ili velikoj industrijskoj grupi te ju posjeduje organizacija koja prodaje usluge računalstva u oblaku.

- privatno računalstvo u oblaku (eng. *Private Cloud Computing*) – Samo organizacija upravlja infrastrukturom. Može biti upravljana od strane organizacije ili neke treće strane te može postojati u zgradi organizacije ili izvan nje.
- hibridno računalstvo u oblaku (eng. *Hybrid Cloud Computing*) – Infrastruktura se sastoji od dva ili više „oblaka“ koji su i dalje jedinstveni entiteti samo su povezani standardizacijom ili vlasničkom tehnologijom koja omogućuje portabilnost podataka i aplikacije.

2.2.7.2. Servisi računalstva u oblaku

Raihana G. F. H. i College J. M. (2012) spominju i objašnjavaju moguće servise koji će se navesti u nastavku.

SaaS (eng. *Software as a Service*)

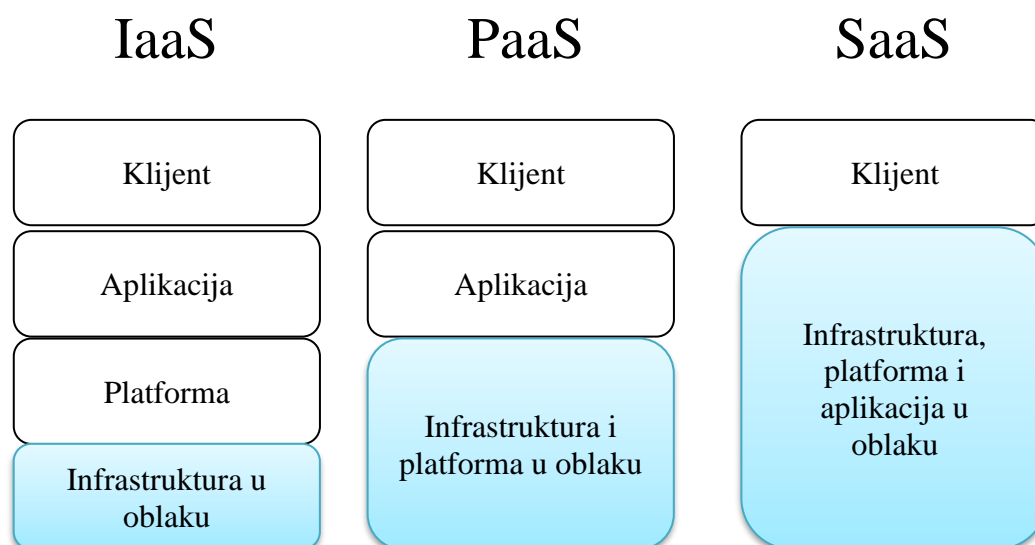
Servisi aplikacija u oblaku (eng. *Cloud Application Services*) ili SaaS dostavlja softver kao servis preko interneta. Na taj način se uklanja potreba za instaliranjem i pokretanjem softvera na računalu te se pojednostavljuje održavanje.

PaaS (eng. *Platform as a Service*)

Platforma kao servis (PaaS) sakriva sloj softvera i prezentira ga kao servis te omogućuje da se koristi za izradu servisa više razine. U prijevodu, mogućnost koju ima korisnik je da sam objavi aplikaciju kreiranu u programskom jeziku koje je podržan od strane pružatelja PaaS-a. U ovom slučaju korisnik ima pristup samo kontroli aplikacije koju je objavio i eventualno konfiguracije okoline gdje se ta aplikacija nalazi. Korisnik ne upravlja ni održava pozadinsku infrastrukturu *cloud*-a uključujući mrežu, servere, operacijske sustave ili spremišta.

IaaS (eng. *Infrastructure as a Service*)

Servisi infrastrukture u oblaku ili IaaS dostavlja računalnu infrastrukturu, obično okolinu virtualizacije za platformu, kao servis. Umjesto da se kupuju serveri, softver, prostor podatkovnog centra ili mrežna oprema i računala klijenti, mogu se kupiti ti resursi kao u potpunosti unajmljeni servis. Servis se obično naplaćuje koliko se resursa troši ili na temelju računalne korisnosti (izračuni, prostor, aplikacije...).



Slika 2.2.7.2.1 Prikazuje strukturu servisa računalstva u oblaku (Izvor: Al-Ghofaili A. A. i Al-Mashari M. A., 2014, str. 44)

Na slici 2.2.7.2.1 je vizualno prikazana struktura spomenutih servisa računalstva radi lakše predodžbe.

2.2.8. Aditivna proizvodnja

Aditivna proizvodnja (*eng. Additive Manufacturing*) postaje popularna dolaskom četvrte industrijske revolucije. Aditivnom proizvodnjom se smatra 3D ispisivanje. Prednost ovog koncepta je da će omogućiti proizvodnju personaliziranih proizvoda te isto tako uvođenjem decentralizirane aditivne proizvodnje će smanjiti udaljenosti transporta i zaliha.

3D ispisivanje je jedna od tehnologija koja omogućuje digitalnu transformaciju. Više informacija je moguće pronaći na internetu te na linku u literaturi koja se koristi. Glavna značajka ove tehnologije je slojevito ispisivanje 3D modela. Postoji više tehnologija 3D ispisivanja koje su spomenute u prethodnom poglavlju. Ova tehnologija se polako koristi u medicini za izradu dijelova za pomoć ljudima u oporavku. Poprilično je korisna te se pomoću nje se može ispisivati sve što se dizajniralo u 3D alatu. Ne može se koristiti u apsolutno svim područjima, ali primjena tehnologije se uvijek može naći.

2.2.9. Proširena stvarnost

Proširena stvarnost (*eng. Augmented Reality*) je tehnologija koja je još uvijek relativno nova, a glavna srž ove tehnologije u kontekstu industrije je da bi radnici nosili uređaje (pametne naočale) i na te uređaje dobivali upute kako ispravno izvršiti dodijeljeni zadatak, također

dodano bi pružale informacije u realnom vremenu koje mogu pomoći u odlučivanju. Sa ovom tehnologijom se susrećem na trenutnom radnom mjestu na kojem radim aplikaciju za pametne naočale koja se koristi u industrijskoj okolini. Navedeni opis zapravo je srž aplikacije na kojoj radim.

Chi i sur (2013) definiraju proširenu stvarnost kao stvarnost koja kreira okolinu gdje su računalno generirane informacije preklapljene sa korisničkim pogledom stvarnog svijeta. Iako je proširena stvarnost relativno nova, javlja se sve više različitih aplikacija koje ju koriste u raznim kontekstima.

2.3. Potreba digitalne transformacije u poslovanju

Koja je svrha digitalne transformacije u poslovanju, na koji način ona poboljšava i olakšava poslovanje. Osvrt na razlike između „staromodno“ poslovanje i poslovanje koje uključuje „digitalna transformacija“ (potkrepljeno primjerima)

Koja je Hrvatska trenutna pozicija u poretku listi koja ovisi o DESI koeficijentu? Što nekoj zemlji znači taj DESI koeficijent? Na koji način digitalna transformacija pomaže poslovanju te koje su razlike između „staromodnog“ poslovanja i kada je prisutna digitalna transformacija? To su neka od pitanja na koja će se odgovoriti u trenutnom poglavlju. Žitnik B. (2016) navodi da je Hrvatska 2016. godine držala 24. mjesto u tablici rezultata. Također navodi sve dostupne servise koji se ubrajaju pod digitalizaciju i oštro kritizira rad i zalaganje aktualne vlade po pitanju ulaganja i napretka digitalizacije u Hrvatskoj. Stier D. I. (2016) piše da je istraživanje povezano sa DESI koeficijentom pokazalo da je u 2016. godini hrvatska zadržala poziciju. Stier D. I. (2016) navodi da digitalizacija u Hrvatskoj polako napreduje i iziskuje prilagodbu i strategije kako bi se stupanj digitalizacije u Hrvatskoj povećao.

Digitalna transformacija je ključan faktor u zadržavanju kompetentnosti na svjetskoj sceni. Hrvatska trenutno polako puži za većim europskim liderima te nastoji držati korak i ubrzati malo u svrhu prilaska prema vrhu ljestvice. Preduvjeti za digitalnu transformaciju su Internet, računalo i volja. Volja Vlade je ključna i najvažnija u ovom procesu digitalne transformacije. Vlada je ta koja je dužna potaknuti i predstaviti ključne mjere i strategije kako bi digitalna transformacija bivala sve više zastupljena u hrvatskoj stvarnosti. Rasprava oko toga koliko Vlada stvarno radi za taj cilj nije tema ovog rada. Digitalna transformacija je korisna za poduzeća jer uklanja postotak ljudske greške. Uvođenjem digitalne transformacije se omogućuje precizniji rad i istovremeno se uklanjaju nepotrebni troškovi koji mogu nastati kao odgovor na ljudsku grešku. Napredne metode planiranja mogu pomoći u tome da se uklone

čekanja i prazni hodovi u procesima proizvodnih industrija. Korištenjem tehnologija digitalne transformacije povećava se kvaliteta proizvoda.

U nastavku će se navesti nekoliko primjera te kako bi se ti primjeri „odigrali“ u situaciji sa i bez digitalne transformacije.

(1) Poduzeće se bavi preradom materijala ili sirovina u svoj proizvod i ključno je da ima dovoljne količine materijala na zalihama. Spletom okolnosti, dobavljač ne stiže dostaviti planirane količine materijala.

a. **Bez digitalne transformacije:** Poduzeće na dan pristizanja resursa od dobavljača shvati da dostava kasni te da neće moći riješiti narudžbu na vrijeme. Obavještavaju naručitelja te se može dogoditi da dio pogona nema što raditi zbog neplaniranog kašnjenja. Preraspodjela radnika se radi na licu mjesta bez prethodnog planiranja. Najgori ishod je da cijela smjena nema posla jer je, primjerice, ova zaliha resursa trebala za danas.

b. **Sa digitalnom transformacijom:** Poduzeće ima uvid u rute i vrijeme dostave materijala i u trenutnoj situaciji može pravovremeno reagirati i obavijestiti naručitelja o mogućem kašnjenju narudžbe ili procijeniti ima li potrebe za obavještavanjem ili je moguće pravovremeno završiti narudžbu unatoč nastaloj situaciji. Moguće rješenje je da se obavijesti vozač o mogućim zastoјima i alternativnom putu kako bi se izbjeglo dodatno kašnjenje. Poduzeće može obavijestiti radnike da odrade neke druge narudžbe nakon što djelomično završe trenutnu narudžbu.

(2) McDonald's je poznati lanac brze hrane. U svojoj opremi imaju strojeve koji koriste tablete kako bi se radili gazirani sokovi. Znači radnik je dužan zamijeniti tabletu kako bi dobili gazirane sokove kao što su Coca Cola, Fanta, Sprite. Ovo je stvarna doživljena situacija. Ti strojevi nemaju indikator za te tablete, iako imaju indikator dok ponestane vode u stroju.

a. **Bez digitalne transformacije:** Tijekom dana, bude navala i gužva te se redaju gazirani sokovi, a radnik zaboravi na tabletu koju treba zamijeniti nakon nekog vremena. Ukoliko se gazirani sokovi i dalje rade sa već istrošenom tabletom, dobiju dojam da je sok razrijeđen. Što znači da se više aroma ne osjeća kako bi trebala. Mogući ishod su žalbe korisnika na kvalitetu soka. Najgori slučaj je da neki korisnici prestanu naručivati

sokove ili prestanu dolaziti. Znači gubitak zarade ili gubitak korisnika su mogući najgori ishodi.

- b. Sa digitalnom transformacijom:** Stroj sadrži u sebi senzor koji broji sokove koji se prodaju. Definira se neki broj sokova nakon kojega je potrebno tabletu promijeniti. Stroj obavještava zvučnim signalom kada broj sokova prođe definiranu granicu što podsjeća radnika da se treba zamijeniti tableta. Na taj način se održava prikladna kvaliteta usluge i zadovoljstvo korisnika.

3. Internet stvari

Mlada ali korisna tehnologija, Internet stvari (*eng. Internet of Things*), nudi rješenja za automatizaciju i poboljšati performanse sustava u kojem se koriste. Razni sigurnosni propusti i neistražene mogućnosti su neke značajke ove tehnologije. Međutim, ista se može koristiti u bilo kojoj industriji. U slijedećim poglavljima proći će se definicija, arhitektura i primjene tehnologije.

3.1. Pojam i definicija

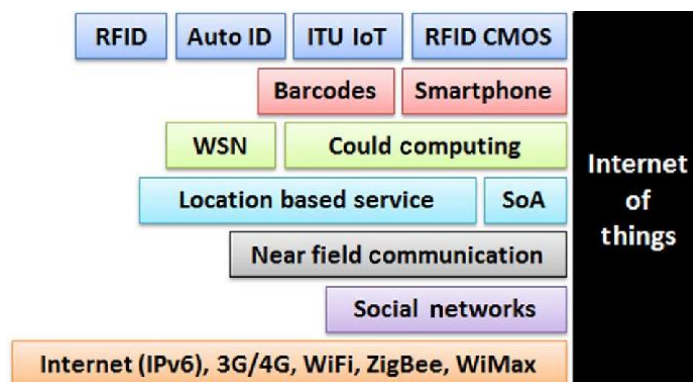
Na spomen pojma Internet stvari, glavna asocijacija su senzori i povezivanje na Internet. U suštini, to i jest Internet stvari. Međutim, Internet stvari se odnosi na uređaje koji su opremljeni radio-frekvencijskom identifikacijskom (RFID) tehnologijom. Internet stvari se ne odnosi samo na senzore, u Internet stvari se uključuju i GPS uređaji te mobilni uređaji. Xu, He i Li (2014) definiraju Internet stvari na sljedeći način: „Internet stvari je dinamička globalna mrežna infrastruktura sa samo-konfigurirajućim sposobnostima temeljenima na standardnim i interoperabilnim komunikacijskim protokolima gdje fizičke i virtualne 'stvari' imaju svoj identitet, fizičke atribute, i virtualne osobnosti te koriste inteligentna sučelja, i neprimjetno su integrirani u mrežu informacija.“ U nastavku će se koristiti kratica IoT za Internet stvar prema direktnom prijevodu na engleski.

Prisutan je veliki interes za industrijskim IoT-om. Neke od područja primjene IoT projekta su: poljoprivreda, industrija obrade hrane, praćenje okoliša itd. Ovo područje je još otvoreno za mnoga istraživanja i nove članke koji mogu dopunjavati neke od poznatih akademskih baza podataka (IEEE, Xplore, Web of Knowledge, ACM digital library, INSPEC i ScienceDirect).

3.2. Postojeća znanja i trenutna istraživanja

Osnovna tehnologija Interneta stvari je RFID tehnologija koja omogućuje prijenos informacije pomoću čitača, a koristeći bežičnu komunikaciju. Prije se ova tehnologija počela koristiti u logistici, maloprodaji i upravljanjem lancem zaliha. Koristila se na način da radnik skenira bar kod proizvoda te na taj način ažurira trenutno stanje tog proizvoda. Recimo u skladištu, neki proizvod može dobiti informaciju da je u dostavnom vozilu iako je prije bio u skladištu i sl. Druga tehnologija, koja se veže za Internet stvari, je mreže bežičnih senzora (*eng. wireless sensor networks - WSN*). Xu, He i Li (2014) spominju da te mreže koriste međusobno spojene inteligentne senzore u svrhu osjećanja i praćenja. Kako već ide sa tehnologijama, one

se razvijaju i unaprjeđuju. Tako su napredovale RFID i WSN tehnologije, omogućujući više primjena IoT-a. Druge tehnologije također doprinose IoT-u, a to su pametni telefoni, računalstvo u oblaku i sl. Sve tehnologije su opisane na slici 3.2.1.



Slika 3.2.1. Tehnologije povezane sa IoT (Izvor: Xu, He i Li, 2014:3)

IoT je tehnologija koja mora biti ponovno iskoristiva u realnom svijetu. U svrhu održavanja ponovne iskoristivosti potrebno je definirati standarde koji bi bili korišteni od strane programera i od strane proizvođača komponenata IoT-a. Postoje mnoštvo organizacija koja se bavi standardizacijom ove tehnologije te se moraju dogovoriti koji će standardi biti primijenjeni na regionalnoj razini a koji na međunarodnoj razini.

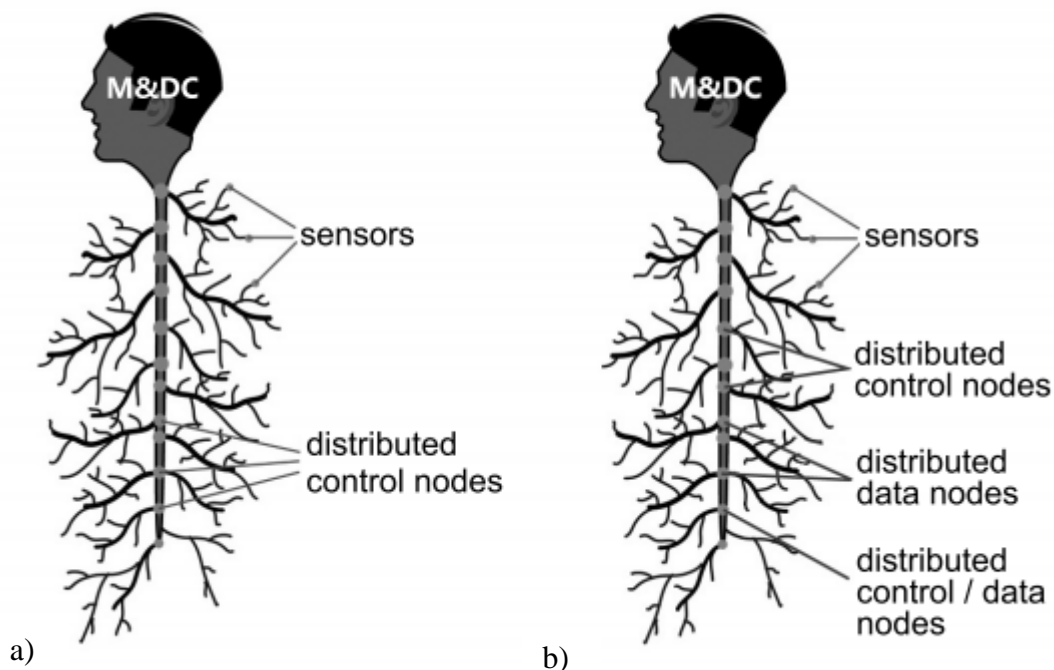
3.3. Arhitektura

Na internetu postoji nekoliko mogućih arhitektura IoT-a. Ukratko će se proći svaka te se objasniti koncepte i značenje svake. Odabrane arhitekture spomenuli su Ning i Wang (2011) te ih je moguće pronaći u literaturi na kraju rada.

3.3.1. MLN model

Javljala se ideja da Internet budućnosti djeluje kao ljudski živčani sustav koji ima osjetila, stoga se ova arhitektura skraćeno naziva MLN (*eng. man-like nervous model*). Ta arhitektura bi se sastojala od tri dijela: mozak (upravljanje i centralno podatkovno središte: M&DC – *eng. management & centralized data center*), leđna moždina (distribuirani kontrolni čvorovi) i mreža živaca (IoT mreža i senzori). Glavni tijek rada ovog modela je da podaci iz mreže živaca odlaze putem leđne moždine prema mozgu, a onda mozak vraća neku informaciju nazad do senzora. Ovaj način ilustrira ideju kako bi se trebala odvijati komunikacija u IoT svijetu. Osnovni model arhitekture je prikazan na slici 3.3.1.1.a). Razlika između ljudskog živčanog sustava i ove arhitekture je ta da distribuirani kontrolni čvorovi u nekim situacijama

bi trebali biti sposobni odgovoriti vanjskim i unutrašnjim podražajima. Drugi, modificirani, model je prikazani na slici 3.3.1.1 b). U ovom model centralno središte se nalazi i u nekim distribuiranim kontrolnim čvorovima leđne moždine te na taj način se smanjuje odgovornost glavnog centralnog središta na više kontrolnih čvorova po leđnoj moždini.



Slika 3.3.1.1. MLN model arhitekture: a) osnovni, b) modificirani (Izvor: Ning i Wang, 2011:2)

3.3.2. SOA model

Internet stvari se često može sastojati od više različitih uređaja odnosno sustava koji naizgled nisu kompatibilni. Drugim riječima može se reći da je Internet stvari mreža heterogenih sustava ili uređaja. U svrhu integriranja takvih sustava i uređaja koristi se SOA (servisno orijentirana arhitektura, *eng. Service-oriented architecture – SOA*). Postoji nekoliko arhitektura SOA-e koje se predlažu, a ona koju su odabrali kao preporučenu i opisali Xu, He i Li (2014) se sastoji od 4 sloja: sloj osjeta (*eng. sensing layer*), sloj mreže (*eng. networking layer*), sloj servisa (*eng. service layer*), sloj sučelja (*eng. interface layer*). Xu, He i Li (2014) se osvrću na bitne osobine kod razmatranja Interneta stvari sa dizajnerske i sa tehničkog aspekta. Sa **dizajnerskog aspekta** bitno je razmotriti slijedeće osobine: **energija** (koliko dugo uređaj može raditi bez da je priključen na izvor električne energije), **latentnost** (koliko dugo traje

putovanje i obrada poruke), **propusnost** (koliko maksimalno prometa se može slati kroz mrežu odjednom), **skalabilnost** (koliko uređaja može jedna mreža podržavati), **topologija** (kako su strukturirani komunikacijski tokovi među uređajima), **sigurnost** (koliko je sigurno upotrebljavati uređaje, tj. koliko ima sigurnosnih propusta u uređaju). Drugi aspekt kojeg autori razmatraju je **tehnoški aspekt**. Uzimajući u obzir ovaj aspekt, potrebno je u arhitekturi Interneta stvari razmatrati slijedeće: **ekstenzibilnost, skalabilnost, modularnost i interoperabilnost** heterogenih uređaja. Internet stvari je u suštini decentralizirana i heterogena mreža, pa po tom pitanju, SOA arhitektura omogućuje orkestraciju, komunikaciju i suradnju različitih uređaja unutar IoT mreže.

Iako prethodni modeli koji su slični ljudskom živčanom sustavu omogućavaju dobru decentralizaciju odgovornosti što je ključno, međutim SOA omogućuje interoperabilnost među uređajima povezanih u mrežu. Samim time se detaljnije posvećuje pozornost interoperabilnosti, odnosno SOA-i. U slijedećim poglavljima će se objasniti četiri sloja arhitekture Interneta stvari.

3.3.2.1.Sloj osjeta

Sloj osjeta (*eng. sensing layer*) je zaslužan za komunikaciju različitih uređaja opremljenih RFID tehnologijom ili senzorima. Internet stvari je mreža mnoštva uređaja opremljenih tom tehnologijom. U ovom sloju, uređaji mogu osjetiti ostale uređaje u svojoj okolini. Da bi se komunikacije i orkestracija mogla odvijati potrebno je jednoznačno označiti svaki uređaj, a tome služi općeniti jedinstveni identifikator (*eng. universal unique identifier - UUID*). Svaki uređaj ima dodijeljen taj identifikator. Taj identifikator omogućuje da se dohvati ciljani uređaj ili servis jer se isti upotrebljava kod identifikacije servisa.

3.3.2.2.Sloj mreže

Glavni zadatak sloja mreže (*eng. networking layer*) je međusobno povezivanje i mapiranje „stvari“ koje se nalaze na mreži. Taj sloj može i agregirati poznate i pristigle informacije. Ključno je da mreža bude sposobna automatski prepoznati „stvari“ na mreži te da se može definirati uloga „stvari“ u pripadajućem kontekstu i u svrhu razvoja, upravljanja i planiranja ponašanja „stvari“ jer ne ponašaju se sve „stvari“ jednako. Neki uređaji mogu imati ulogu centralnih čvorova u nekoj lokalnoj mreži i pritom raditi operacije agregiranja i slanja podataka na Internet ili neko drugo odredište, dok u nekom drugom scenariju uređaj može biti samo list u toj mreži i mjeriti ili sakupljati podatke i slati ih na centralni čvor koji dalje prosljeđuje podatke na odredište.

Dizajnerski aspekt ovog sloja pokreće razna pitanja, kao što su pitanje energetske učinkovitosti u mreži, pitanje sigurnosti i privatnosti, pitanje tehnologije upravljanja mrežom, kako obrađivati signale i podatke te kako kontrolirati otkrivanje i dohvat servisa.

3.3.2.3.Sloj servisa

Sloj servisa (*eng. service layer*) je zaslužan sa dodjeljivanje različitih servisa uređajima na mreži. Razne organizacije omogućuju API-ove i protokole za najčešće korištene servise kako bi se isti mogli ponovno iskoristiti kada za to bude potrebe. U ovom sloju događa se „slijepi spoj“ servisa i povezane „stvari“ jer se ne mora nužno dogoditi da je taj isti servis već „upoznao“ tu „stvar“. U ovom sloju postoje neki problemi koji su aktualni te se razmatraju. Neki od tih problema su problemi povezani sa servisima uključujući razmjenu i spremanje informacije, upravljanje podacima, pretraživanje i komunikacija. Komponente, koje opisuju Xu, He i Li (2014), su:

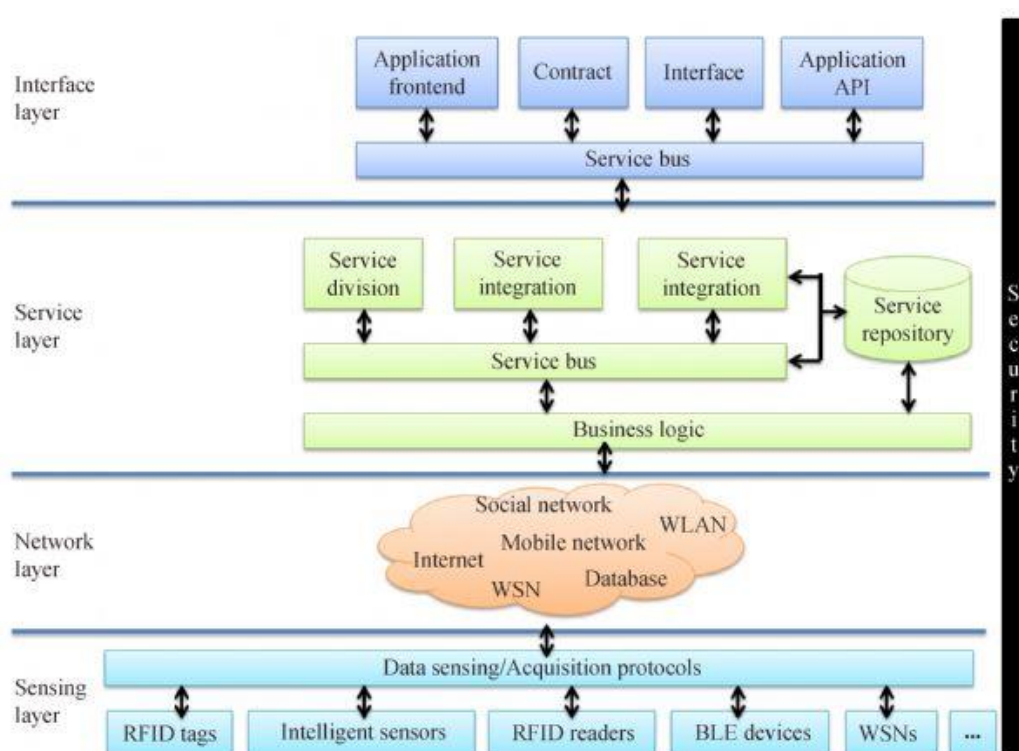
- 1) **Otkrivanje servisa** se odnosi na traženje objekata koji su sposobni ponuditi željeni servis i informaciju.
- 2) **Kompozicija servisa** omogućuje interakciju i komunikaciju između povezanih stvari. Kompozicija servisa je zaslužna za planiranje i rekreiranje najpovoljnijih te najpouzdanijih servisa u svrhu zadovoljenja zahtjeva.
- 3) **Upravljanje povjerljivošću** cilja na sigurni aspekt komunikacije među servisima. Cilja se na razvoj mehanizma koji osigurava pouzdanost informacija koje se dijele među servisima i uređajima te na autentikaciju servisa da se ne dogodi maliciozna akcija koja može naštetiti cijelom sustavu.
- 4) **API servisa** je sučelje koje je korišteno od strane uređaja u svrhu dobivanja informacija i funkcionalnosti dostupnog servisa.

3.3.2.4.Sloj sučelja

Idealni slučaj bi bio kada bi svi proizvođači pratili jedan standard i jedan protokol i postavili ga kao apsolutni standard za IoT uređaje. Stvarnost je drugačija: različiti proizvođači ne prate uvijek iste standarde i protokole. Heterogenost standarda i protokola uzrokuje probleme u razmjeni informacija među uređajima te njihovoj međusobnoj komunikaciji. Kontinuiran rast uređaja prijavljenih na mreži otežava dinamičko spajanje, komunikaciju, isključenje i rad. Zbog toga se javlja sloj sučelja (*eng. interface layer*) kako bi se pojednostavnilo upravljanje i unutarnja povezanost među uređajima. Sloj servisa nudi općeniti

API koji omogućuje lakšu interakciju aplikacija i servisa. Efektivan način korištenja servisa je rezerviranje servisa. Xu, He i Li (2014) opisuju postupak:

Proces rezerviranja servisa prvo izvršava upit tipova (eng. types query) koji šalje zahtjev za servisom sa generičkim WSDL formatom i tada koristi mehanizam pretraživanja kandidata (eng. candidate search) kako bi pronašao potencijalne servise. Temeljeno na kontekstu aplikacije i informaciji kvalitete usluge, sve servisne instance se poredaju i mehanizam rezerviranja servisa po zahtjevu (eng. On-Demand service provisioning) se koristi kako bi se identificirale instance servisa koje zadovoljavaju zahtjeve aplikacije. I na kraju, proces evaluacije se koristi kako bi se ocijenio sam proces.



Slika 3.3.2.1. SOA za IoT (Izvor: Xu, He i Li, 2014, str. 5)

Slika 3.3.2.1. Prikazuje opisanih 4 sloja SOA-e za Internet stvari. Sve opisano u prethodnim poglavljima nalazi se na slici.

3.4. Internet stvari u poslovanju

Prije četvrte industrijske revolucije, poslovanja su se svodila na determinističkim procesima te se lagano moglo preslikati stvarno stanje u dijagram u *Business Process Modeleru* (BPM-u). Do sada se samo poslovanje temeljilo na poznatim podacima i podacima koji su nastali u samom poslovanju. Dolazak IoT-a se počinje činiti kao odlična prilika za poboljšavanje kompetentnosti na tržištu. Sve više se traži načini kako bi se IoT integrirao u

poslovanje. Haller S. i Magerkuth C. (2011) su dotaknuli temu problematike integracije IoT-a u poslovanje. Oni navode da sadašnji sustavi, koje je moguće konfigurirati, integrirati i pratiti, moraju biti u mogućnosti samostalno se konfigurirati, pratiti i oporavljati. U prijevodu sustav treba postati inteligentan kako bi se mogao nositi sa mnoštvom IoT uređaja. Također navode da dosadašnji SOA (*eng. service-oriented architecture*) pristup kod web servisa neće biti dovoljno dobar za IoT uređaje zbog njihove heterogenosti te predlažu da se umjesto SOA koristi REST (*eng. Representational State Transfer*) servisi. Haller S. i Magerkuth C. (2011) navode da se dosadašnji BPM ne može nositi sa procesima koji se aktiviraju na određenu poruku odnosno događaj poslan od strane IoT uređaja. Takav događaj može rezultirati sa više različitih slučajeva i kompleksnosti ovisno o trenutku kada taj događaj pristigne. Novi BPM bi trebao moći modelirati takve događaje bez dodavanja kompleksnosti na postojeće modele. Još jedan problem, koji ističu, je relevantnost podataka. Vrlo je bitno pratiti odakle dolaze podaci te imaju li oni vrijednosti koja je relevantna za poduzeće da se izbjegne preplavljanje poduzeća da nepotrebnim podacima.

Internet stvari u poslovanju može rezultirati sa smanjenjem troškova i poboljšanjem produktivnosti. Istovremeno može donijeti sa sobom probleme poduzeću kao što su: nemogućnost uvrštavanja IoT-a u modelirane procese, stohastičko ponašanje, troškove integracije IoT-a u poslovanje. Ukoliko se ovi problemi uspješno riješe, IoT će biti pun pogodak za poduzeće koje ga koristi. Jedan od problema koje nosi IoT je sigurnost. Sigurnosni aspekt IoT uređaja nije dovoljno učvršćen te se tu mora pristupati s povećim oprezom te se konzultirati na koji način osigurati svoj IoT uređaj od hakera i onemogućiti zlonamjerne akcije.

Postoje neki poznati primjeri primjene IoT-a u poduzećima kao što su bar kodovi i kontrola stanja na zalihama pogotovo ako poduzeće posluje i *online* i *offline*, GPS praćenje vozila ukoliko se poduzeće bavi dostavom, kontrola strojeva u proizvodnom procesu te osiguravanje optimalnih uvjeta rada i slično.

3.5. Područja primjene

U nastavku će se predstaviti osnovna područja primjene Interneta stvari. Osnovni koncepti koji su danas poznati. Svako potpoglavlje će ilustrirati koncept tako da se lakše dobije ideja za razvoj proizvoda.

3.5.1. Pametna kuća

„Pametna kuća (*eng. Smart Home*) se može definirati kao prebivalište opremljeno računalnom i informacijskom tehnologijom koja predviđa i odgovara na potrebe stanovnika poboljšavajući njihovu udobnost, ugodnost, sigurnost i zabavu kroz upravljanjem tehnologijom unutar doma i povezanost prema svijetu izvan njega.“ (Harper, 2003, str. 17) Pametne kuće su trenutno više ravnije znanstvenoj fantastici nego suvremenoj realnosti, međutim nije na odmet tvrditi da će u budućnosti trenutna kuće biti smatrane staromodnima, a pametne kuće normalnom pojavom.

Glavni razlog toga da su pametne kuće jednake znanstvenoj fantastici je taj da su to skupe investicije u vremenu pisanja rada. Naime postojano je nekoliko problema koji se javljaju u tržištu sa pametnim kućama. Pametne kuće su skupa investicija pa one ni ne pokrivaju veliki postotak ljudi, već samo ljude u srednjoj i visokoj klasi. Drugi problem je da kuće već postoje i nije jednostavno zapravo samo tu postojeću kuću pretvoriti u pametnu kuću. S druge strane, ljudima se ne isplati financirati ponovnu izgradnju kuće koja će biti uvelike skuplja nego postojeća. Pametne kuće su tehnologija koja bi trebala biti unificirana, odnosno trebao bi postojati standard koji omogućuje da bilo koji uređaj za namjenu u pametnoj kući bude stvarno kompatibilan s istom neovisno o proizvođaču. To je još jedan problem. Glavni cilj ove tehnologije je riješiti se rutinskog dijela života koji lako može biti upravljan računalnom tehnologijom. Što i proizvođači pokušavaju postići, ali nikada nisu mjerili korisnost svojih proizvoda što predstavlja problem. Međutim, s obzirom na više konteksta u kojima se nalazi proizvod, vrlo je teško odrediti ocjenu korisnosti.

Trenutni tijek kupnje pametne kuće je da kupac plati izgradnju i tek onda kreće izgradnja pametne kuće. Teoretski, kupac kupuje nešto što nije mogao isprobati niti zna što će mu ta investicija donijeti. To je jedan od težih izazova kojima se susreće industrija pametne kuće. Industrija pametne kuće bi trebala motivirati kupca da kupi kuću, međutim da bi to bilo moguće mora zadovoljiti i većinu kriterija koje oni traže prije nego li se odluče za kupnju. Ideja je također bila da se definiraju neki moduli iliti blokovi koji bi omogućili krojenje sustava pametne kuće po željama korisnika na brz način i koristeći koncept ponovne iskoristivosti. Također su željeli omogućiti laku nadogradivost, prilagodljivost i laku instalaciju sustava.

Pametna kuća služi primarno da se zadovolje potrebe korisnika odnosno stanovnika. Vrlo je teško povjerovati da će nešto viđeno na malom ekranu postati stvarnost. Glavni problem svega ovoga je pametna izgradnja sustava pametne kuće. Idealno bi bilo stvoriti modularnost i omogućiti „gradnju“ sustava pametne kuće prilagođavajući se potrebama korisnika. Vrlo je

važno potpunu pažnju posvetiti samom korisniku, jer ipak je korisnik taj koji će koristiti kuću. Korak prema uspješnim sustavima pametnih kuća je neki standardni protokol koji će omogućiti sučelje i integraciju više različitih sustava, zbog prisustva standarda kojih su se proizvođači tih sustava držali. Vrlo je teško prodati pametnu kuću nekome, jer je većina korisnika skeptična i potreban je jedan testni period da se korisnik odluči hoće li kupiti kuću ili ne. Također je jedan od problema, koji je naveden u odlomku prije, da se mora naći način kako opremiti već izgrađene i opremljene kuće tako da one postanu pametne kuće. To je poprilično težak zadatak jer uglavnom svaka integracija zahtjeva neke veće promjene. Imajući to na umu, idealno je da se grade pametne kuće iz temelja zbog lakšeg integriranje svih potrebnih uređaja i senzora sa samom kućom još dok se ista gradi. Lakše je dok se zna gdje će ići što nego dok se radi naknadno. Postoji još jedan problem, naveden u ranije u tekstu, a to je sama cijena kuće. Po cijeni, kuća obuhvaća ljude koji pripadaju srednjem odnosnom visokom sloju. Pod terminom „sloj“, misli se na financijski stabilne osobe.

3.5.1.1. Definicija izraza „pametna kuća“

Trebalo bi biti jasno da jedan zatvoreni strujni krug ne znači da je uređaj pametan. Znači termostatski koji očitava vanjsku temperaturu nije pametan, međutim termostatski koji regulira temperaturu u odnosu na vanjsku temperaturu jest pametniji. Harper (2003) klasificira pametne kuće ovisno o razini „pameti“. Umjesto prepisivanja njegove klasifikacije, samo će se sumirati točno na što se misli izrazom: razina „pameti“. Postoji nekoliko razina, prva i osnovna razina je pametna kuća opremljena inteligentnim objektima. Sljedeća razina je pametna kuća sa inteligentnim objektima koji komuniciraju kako bi povećali svoju funkcionalnost. Povezane kuće su naprednije po tome što omogućuju pristup servisima unutar i izvan kuće. Učeće kuće uče o potrebama svoji stanovnika te na taj način orkestriraju svoje akcije. Uslužne kuće prate navike i običaje svoji stanovnika i od toga uče te i predlažu neke aktivnosti svojim stanovnicima kako bi im udovoljile. Akcije koje se izvode u pametnim kućama mogu biti jednostavne (stanja uključeno/isključeno) ili izrazito kompleksne akcije koje se ne mogu opisati jednostavnim stanjima. Još jedan problem koji se javlja je sigurnost. Glavna briga u današnje vrijeme je sigurnost. Za pametne kuće to je jedan od korisničkih zahtjeva koji bi trebao biti besprijekorno zadovoljen osobito ako se radi o uslužnoj kući. Ne bi bilo poželjno da se događaju maliciozne akcije u vlastitom domu.

Pametna kuća je koncept koji može promijeniti cijeli način i filozofiju življenja u istoj. Također postoje različite razine uključenosti kuće u život stanara te koliko ona aktivno sudjeluje u osiguravanju njihove udobnosti. Naravno, što je pametna kuća više pametna to je bolje

iskustvo življenja u istoj. Jedan od glavnih problema pametne kuće je da se ista mora graditi od početka te je vrlo teško normalne kuće pretvoriti u pametne dok su već izgrađene. Ovo je bio samo dio predstavljanja koncepta pametne kuće, ukoliko se želi saznati više može se pronaći literatura na internetu i poslužiti se njome u dopunjavanju znanja.

3.5.2. Pametni grad

Gradovi nisu samo pojavili, oni su nastajali godinama te su bili predmet proučavanja i planiranja te izgradnje. Gradovi se spominju od antičkih vremena pa sve do suvremene povijesti i još uvijek su tema istraživanja i akademskih radova. Kroz povijest su nastajale neke ideje kako bi grad trebao izgledati te koje su mu najbitnije osobine, postoje i razne knjige koje se bave tom problematikom.

Pametni grad je malo naprednija vrsta grada. Trenutno isto tako nije standard u svijetu nego je još uvijek bliže znanstvenoj fantastici nego realnosti. Postoje razne definicije pametnih gradova koje predlažu specijalisti. Te definicije su naveli Eremia, Toma i Sanduleac (2017):

„Pametni grad“ koristi informacijsku i telekomunikacijsku tehnologiju u svrhu poboljšanja sposobnost življenja, rada i održivosti.

Grad koji prati i integrira uvjete svih njegovih kritičnih infrastruktura – uključujući ceste, mostove, tunele, željezničke pruge, podzemne željeznice, aerodrome, pomorske luke, komunikacije, vodu, električnu energiju, također i glavnih zgrada (eng major buildings) – može bolje optimizirati svoje resurse, planirati svoje preventivne aktivnosti održavanja i pratiti sigurnosni aspekt dok maksimizira usluge za stanovnike.

Pametni grad može biti viđen kao određen geografski prostor sposoban upravljanja resursima (prirodnim, ljudskim, alatima, zgradama i infrastrukturuom) i otpadom nastalim od životnog stila. On bi trebao biti održiv i ne bi smio biti štetan za okolinu.

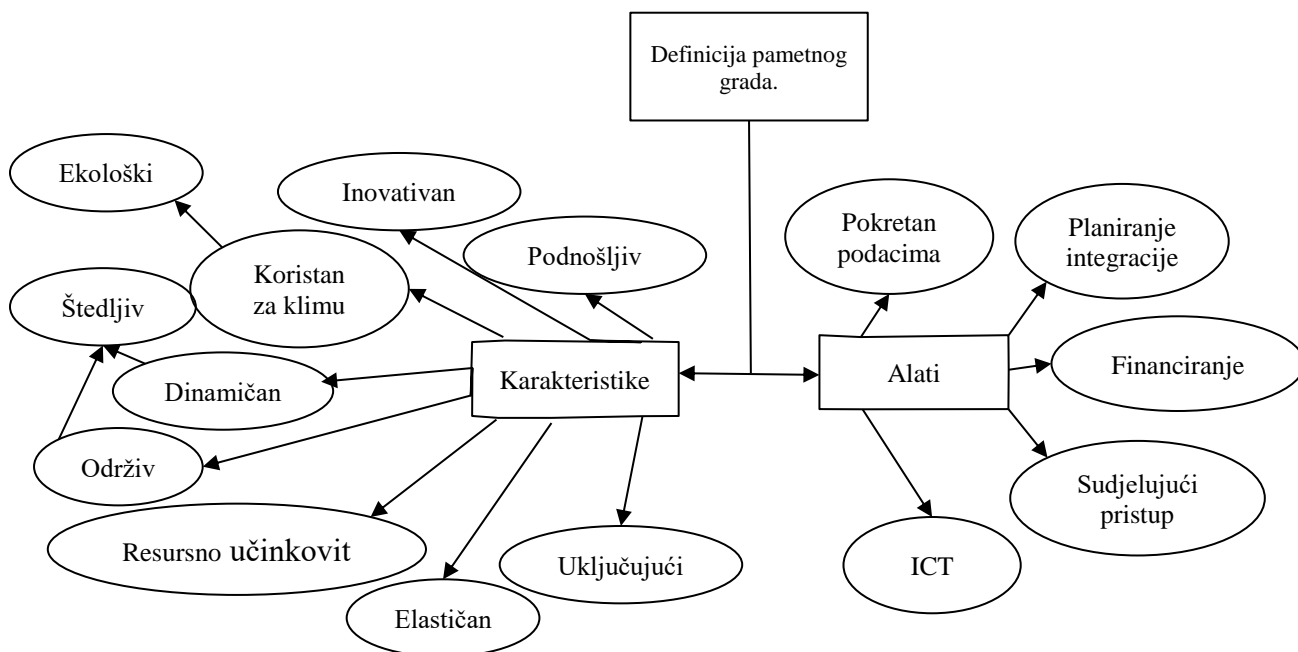
Iz definicija je jasno vidljivo nekoliko stereotipa pametnog grada: povezanost za ICT-om, upravljanje resursima i ekološka korisnost. Jasno je vidljivo da ljudi uče iz grešaka koje i dan danas pokreću trenutne gradove. Pametni grad će surađivati sa prirodom i za prirodu i nikako protiv prirode.

Pametni grad je noviji koncept koji se uvelike razlikuje od „staromodnih“ gradova. Pogotovo zato što pametni grad ima stalni uvid u potrošnju resursa. Ideja oko tog koncepta je stvarno korisna za sve gradove i trebali bi se svi okrenuti u tom smjeru. Prednost ovog koncepta je ta što se lako može mjeriti potrošnja resursa te se mogu prikupljati druge informacije. Samo mašta je granica u tom smislu. Nedostatak tog koncepta je cijena infrastrukture i cijena

uvođenja iste u već postojeću. Što može biti jedna od glavnih prepreka u razvoju i širenju pametnih gradova. U nastavku će se pokazati neke karakteristike pametnih gradova.

3.5.2.1. Karakteristike pametnih gradova

Postoje glavne karakteristike grada i njegovih alata koji mogu transformirati grad u pametni grad, a te karakteristike su prikazane na slici 3.5.2.1.1.



Slika 3.5.2.1.1. Karakteristike i alati korišteni za definiranje pametnog grada

(Izvor: Eremia, Toma i Sanduleac, 2017, str. 16)

U ISO standardu 37120/2014, kojeg spominju Eremia, Toma i Sanduleac (2017), postoji 17 ključnih indikatora za ocjenjivanje gradova sa stajališta osiguravanja urbanih usluga i kvalitete života: ekonomija, edukacije, energija, okoliš, financije, odgovor na požar i hitne slučajeve (*eng. fire and emergency response*), vlada, zdravstvo, razonoda (rekreacija, sigurnost, sklonište, kruti otpad, telekomunikacije i inovacije, transport, urbano planiranje, otpadne vode, vode i provođenje zdravstvenih mjera (*eng. sanitation*)).

Tablica 3.5.2.1.1 prikazuje područja primjena koncepta pametnog grada.

ICT je ključan za omogućivanje razvoja grada sa perspektive ekonomije te društvene i kulturne perspektive. Također je važno da se u razvoj grada uključe i građani te da se uči na greškama i omogućuju se uvjeti koji su ključni za povećavanje životnog standarda ljudi koji žive u pametnom gradu. ICT tehnologija je ključna kako bi dala aspekt „inteligentnosti“ gradu. U tome uvelike pridonosi i Internet stvari koji sa senzorima i računalstvom u oblaku olakšava

put prema pametnom gradu. Upravo Internet stvari omogućuje lakši pristup servisima i informacijama te oni mogu pomoći u transformaciji odnosno izgradnji pametnog grada. Senzori također omogućuju realizaciju nekih od karakteristika pametnog grada. Internet stvari se koristi kako bi povezao većinu aspekata grada u jednu inteligentnu cjelinu.

Područje primjene	Opis	Primjeri
Pametne zgrade (<i>eng. Smart Buildings</i>)	Pametne zgrade koje pripajaju prednosti komunikacijskih i kontrolnih sustava	Optimiziranje sustava grijanja, ventilacije i klimatizacije.
Edukacija, medicinska i socijalna pomoć	Primjene koje omogućuju poboljšanje aktivnosti u tim domenama i osiguravaju pristup svim građanima do usluga visoke kvalitete.	Promatranje sustava starih ljudi, promatranje putem telemedicine (<i>eng. telemedicine</i>).
Pametna energija	Pametni sustav električne energije koji povezuje sve komunalne usluge i krajnje korisnike kroz pametnu infrastrukturu.	Primjene „Smartgrid“-a iliti pametne mreže, optimizacija mrežnih operacija, održavanje standarda okoline, pametno osvjetljenje (<i>eng. smart lighting</i>).
Pametna mreža (<i>eng. smart grid</i>) (pametno mjerenje prirodnog plina, vode i električne energije)	Mjerenje potrošnje energije, vode i prirodnog plina u realnom vremenu.	Online informacija o potrošnji; bežična pametna brojila.
Pametne komunalne usluge (<i>eng. smart utilities</i>) (pametna distribucija vode i pametno upravljanje otpadom)	Inteligentno upravljanje sustava distribucije vode i otpadnih voda.	Pametni sustavi otpadnih voda; praćenje krutog otpada u realnom vremenu.
Pametno parkiranje	Upravljanje parkirnih mjesta koristeći senzore, CCTV.	Praćenje sustava vozila.
Integrirani sustavi opskrbe	Usklađivanje ponude sa potražnjom; mjerenje, promatranje i organiziranje transporta oko nabavnih lanaca gradova.	
Pametni i integrirani transport	Promatranje prometa i optimizacija u realnom vremenu koristeći i kombinirajući sva transportna sredstva.	CCTV za promet; Pametne mreže parkirnih mjesta; Minimiziranje utjecaja na okolinu.

Tablica 3.5.2.1.1. Smjerovi razvoja pametnih gradova (Izvor: Eremia, Toma i Sanduleac, 2017, str. 17)

U ovom poglavlju je navedeno što pametni grad čini pametnim. Ciljevi pametnih gradova iz tablice 3.5.2.1.1. se čine vrlo interesantnim i bilo bi super kada bi se svi ti ciljevi implementirali. Kako je spomenuto, dobra pomoć pri realizaciji tih ciljeva je IoT. Ukoliko bi došlo do realizacije navedenih ciljeva, uvelike bi se poboljšao životni standard i upravljanje troškovima i resursima grada. Pametni grad je praktički jedna povezana mreža informacija koje se koriste u svrhu poboljšanja standarda. Super bi bilo da se može na internetu provjeriti

dostupnost parkirnih mjesta bilo kojeg parkirališta u gradu. Također, mjeriti struju i vodu te vidjeti direktan izračun troška za trenutno stanje. Sve je to popraćeno i drugom, lošom, stranom. Sve povezano na Internet nosi za sobom i hakerske napade. Stoga je vrlo bitno brinuti o sigurnosti što će malo više biti spomenuto u sljedećem poglavlju.

3.5.2.2. Veza između Pametne mreže i Pametnog grada

Eremia, Toma i Sanduleac (2017) spominju da pametni gradovi integriraju različite infrastrukture i sustave vertikalno i horizontalno. Informacijska telekomunikacijska i komunikacijska infrastruktura predstavljaju podršku za upravljanje i kontrolu aplikacijama, a pametna mreža (*eng. smart grid*) je sustav koji omogućuje da ostali sustavi funkcioniraju. Bit ovih sustava je da surađuju zajedno i da se prilagođavaju kada rade pod ekstremnim uvjetima.

Električna energija je poprilično važan resurs za funkcioniranje grada. Kada se radi o pametnom gradu ona je kritičan resurs da sve funkcionira ispravno. U literaturi se spominje termin „pametna energija (*eng. smart energy*)“. Taj izraz označava veliku važnost električne energije u aktivnosti gradova. Izraz je temeljen na filozofiji prilagođavanja za dugoročne najefikasnije strategije sa ekonomske perspektive dok se zadovoljavaju ekološki zahtjevi. Ukoliko se dogodi da nestane električne energije, neki servisi i mreža komunalnih usluga postanu nedostupni što može naštetiti gradu.

Eremia, Toma i Sanduleac (2017) nabrajaju pet osnovnih smjerova razvoja pametnih mreža u gradovima:

- A. Promoviranje izvora čiste energije.** Ovaj smjer podržava ekološke zahtjeve. Gradovi ciljaju na korištenje pametnih mreža koje omogućuju razvoj obnovljivih izvora energije i elektrana visoko-efikasne kogeneracije (*eng. cogeneration*). Također je ključno integriranje distribuiranih izvora na siguran i pouzdan način što omogućuje fleksibilno opskrbljivanje električnom energijom.
- B. Pametno mjerenje.** Ovaj smjer će omogućiti da ljudi mogu pratiti vlastitu potrošnju te istovremeno će omogućiti ponuđačima da mogu udaljeno očitavati brojila, odnosno oni ne moraju biti fizički prisutni da očitaju brojilo već se isto može očitati u aplikaciji koja bi sadržavala stanja svih kućanstava u gradu. To će također omogućiti da ponuđači predvide kada će biti najveća opterećenja te u skladu sa time planirati sustav distribucije i buduće akcije radi održavanja dostupnosti mjerenog resursa.

- C. Efikasna javna rasvjeta.** Općine pokazuju povećan interes za prilagodbu održivih rješenja kao mjeru poboljšanja efikasnosti energije. Neki primjeri su korištenje lampi sa niskom potrošnjom i visokom efikasnosti za javnu rasvjetu (primjerice, LED lampe) ili korištenje senzora za automatsko paljenje ili gašenje svjetla kada je to potrebno, uključujući svjetla u administrativnim zgradama.
- D. Integracija električnog vozila.** Onečišćenje zraka u gradovima je jedan od najvažnijih problema javnog zdravstva. Integracija električnih vozila velikih razmjera zahtjeva inteligentna rješenja koja će se prilagoditi u električnim mrežama. Upravljanje punjenjem baterije je srž za izbjegavanje preopterećenja mreže i za podržavanje optimalnog korištenja čiste energije. Električno vozilo je sredstvo za rukovanje viškom energije iz obnovljivih izvora, većinom tijekom noćnih razdoblja.
- E. Aktivno uključivanje potrošača.** Primjene nove pametne mreže su usmjerene prema potrošaču; pametne mreže su korisnički definirane da informiraju, educiraju i pripomažu potrošaču u donošenju najboljih odluka. Sa integracijom električnog vozila i podržavanjem malih izvora energije, potrošači mogu postati klijenti, koji imaju mogućnost dodavanja energije u električnu mrežu tijekom razdoblja vršnih opterećenja. Potrošač može, zbog tog, biti aktivno uključen u efikasno korištenje energije.

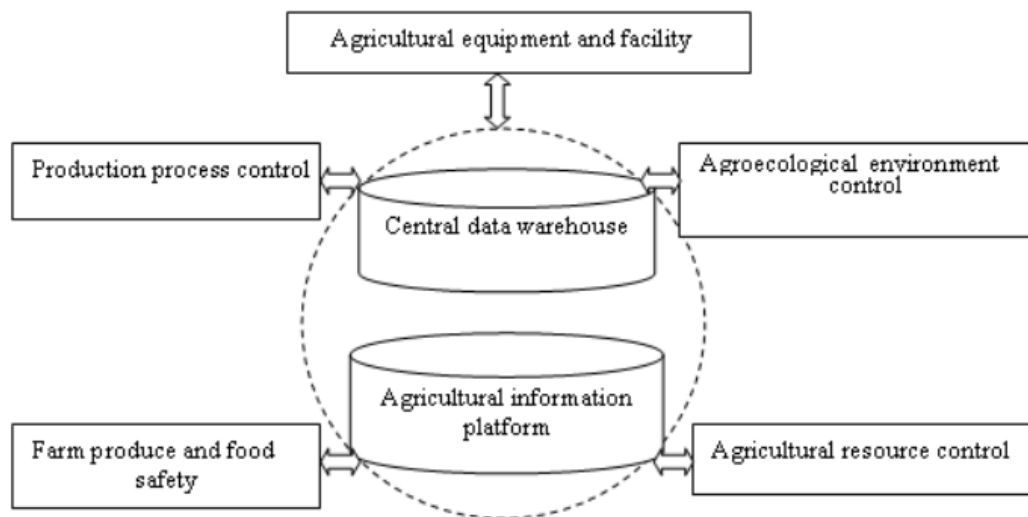
Pametna mreža je sredstvo, a pametni grad se koristi tim sredstvom da može normalno funkcionirati. Uz sve to, istaknulo se kako je električna energija bitan izvor energije u moderno doba. Duži nedostatak iste može uzrokovati ekonomsku štetu. Spomenuti su bili smjerovi razvoja pametnog grada. Implementacija i pokretanje tih smjerova nije jeftina i zahtjeva prilagodbu postojeće ili izgradnju nove infrastrukture. Možda najteži smjer za implementaciju su električni auti. Trenutno u svijetu gotovo svaka osoba ima bar jedan automobil, te tvornice prije nisu ni razmišljale o tome da bi mogli početi proizvoditi električne automobile. Danas su tvornice automobila krenule sa proizvodnjom električnih vozila te šire svoju ponudu na hibridna i električna vozila. To je pametan korak kako te tvrtke ne bi doživjele ekonomski šok i propast zbog naglog uvođenja električnih automobila. Ti automobili su još uvijek u klasi vrlo skupih automobila u Hrvatskoj unatoč najavama da će se cijene električnih automobila spustiti na razinu automobila na gorivo. Također nije uredu naglo uvesti električne automobile jer to može biti šok i za naftne tvrtke. Infrastruktura se mora postepeno prilagođavati što uključuje i uvođenje punionica za električne automobile da bi se investicija u njih isplatila. Trenutno je

razdoblje kada zemlje pokušavaju povećati udio električnih automobila i pridonijeti prirodi smanjujući emisiju štetnih plinova. Električni automobili su vrlo dobra investicija dugoročno gledano, ali početni trošak je često preveliki udarac za novčanik. Jednom je bila predstavljena cesta koja bi punila električne automobile kako bi se vozili po njoj te da bi ista imala svjetlosne znakove i oznake na cesti umjesto nacrtanih. Nije bilo naknadnih vijesti o njezinoj realizaciji, ali ona bi bila savršen primjer za aktivno uključivanje potrošača. Trenutno, su baterije problem koji može smanjiti trajanje električnog vozila i povećati frekvenciju punjenja po dnevnoj kilometraži. Smatra se da baterije možda nisu trenutno najbolje rješenje jer brzo opada njezin kapacitet sa količinom punjenja. Električna infrastruktura je srž jednog pametnog grada odnosno pametne mreže. Sam razvoj i napredak tehnologije, tjera i električnu infrastrukturu na modernizaciju i praćenje trendova. S obzirom da je pametna mreža gotovo uvijek na internetu, vrlo je bitno osigurati adekvatnu sigurnost i zaštitu kritičnih komponenti sustava. Koliko se god trudili zaštititi se, prijetnja sa interneta uvijek bude prisutna i tražila načine da uđe u sustave u koje nema dozvoljen pristup. Poznata je vijest kako su uspjeli hakirati električne aute i preuzeti kontrolu nad njima. Zamislite što bi se dogodilo da jedna hakerska skupina može isključiti struju cijelog grada na duži vremenski period iz udobnosti vlastitog doma i sofe. Koliko god to bilo opasno i riskantno, mora se razviti adekvatna zaštita komponentata sustava da bi sustav ispravno i sigurno radio. Uz zaštitu, vrlo je bitno i definirati standard za pametnu mrežu i pametni grad. Standardizacija za komponente je prijeko potrebna kako bi se spriječili dodatni i nepotrebni troškovi te problem nekompatibilnosti. To je vrlo bitno tako da se komponente različitih proizvođača mogu koristiti u istom sustavu tako da se korisnik ne mora držati samo određenog proizvođača, nego gleda na specifikacije proizvoda koji planira kupiti.

3.5.3. Pametna poljoprivreda

Pametna poljoprivreda (*eng. smart agriculture*) je koncept koji integrira poljoprivredu te ICT i IoT. Informatizacija poljoprivrede donosi veću preciznost u održavanju uvjeta povoljnih za određenu poljoprivrednu kulturu te samim time i efikasno veći urod. Postoji mnoštvo projekata koji su se zalagali za poticanje pametne poljoprivrede. Ti projekti se mogu naći u literaturi autora TongKe-a (2013). Postoji informacija da poljoprivrednici ne koriste potencijale pametne poljoprivrede kompletno već djelomično. Smatra se da bi trebao postojati repozitorij svih informacija vezanih za poljoprivredu u oblaku. Te informacije bi bile integrirane sa IoT i RFID tehnologijom i na taj način realizirati pametnu poljoprivredu u potpunosti. Taj repozitorij bi mogao služiti kao baza za odlučivanje u toj grani.

TongKe (2013) opisuje kontrolnu arhitekturu pametne poljoprivrede. Temelj arhitekture je računalstvo u oblaku i vizualizacija iste je prikazana na slici 3.5.3.1. Sastoji se od kontrolne platforme i baze podataka, a kontrolna platforma se sastoji od podsustava. TongKe (2013) navodi te podsustave koji se mogu pronaći nakon slike.



Slika 3.5.3.1 Kontrolna arhitektura pametne poljoprivrede temeljna na računalstvu u oblaku i Internetu stvari (Izvor: TongKe, 2013, str. 6)

Podsustav kontrole agro-ekološke okoline uključuje:

- (1) Praćenje kvalitete vode, automatsko poboljšanje kvalitete vode
- (2) Precizno gnojenje šteti gnojivo
- (3) Praćenje sastava zemlje, vlažnosti zemlje, svjetla, vjetra, zraka, itd.

Podsustav kontrole poljoprivrednih resursa uključuje:

- (1) Inteligentni staklenik koji omogućuje automatsko podešavanje temperature
- (2) Vodeno navodnjavanje koji može automatski kontrolirati protok i štedjeti vodu
- (3) Znanstvene bolesti i praćenje štetočina (*eng. Scientific disease and pest monitoring*)

Podsustav kontrole procesa prodaje uključuje:

- (1) Identifikacija individualne životinje omogućuje zdrav uzgoj
- (2) Praćenje rasta životinje i biljke
- (3) Sortiranje proizvoda garantira kvalitetu

Podsustav poljoprivrednih proizvoda i sigurnosti hrane uključuje:

- (1) Informiranje o cijelom logističkom procesu

- (2) Racionalan raspored prostora u skladištu
- (3) Sustav praćenja opskrbnog lanca poljoprivrednih proizvoda

Sustav poljoprivredne opreme i postrojenja uključuje:

- (1) Dijagnoza kvarova poljoprivrednih strojeva
- (2) Udaljena kontrola poljoprivrednih strojeva
- (3) Praćenje rada poljoprivrednih strojeva

U poglavlju je ukratko spomenuta ideja pametne poljoprivrede kako bi taj koncept izgledao u stvarnosti. Smatra se da je korisna stvar pratiti stanja u poljoprivredi, kao što su stanje spremnika za navodnjavanje, stanje vlažnosti zemlje i tako dalje. Bit pametne poljoprivrede je koristiti tehnologiju u ostvarivanju najveće moguće efikasnosti u proizvodnji. Koristeći tehnologije, prikupljaju se podaci koji onda služe za donošenje odluka. Neke od osnovnih stvari u ovom konceptu su praćenje sastava tla, praćenje vlažnosti, temperature, i slično. Ovaj koncept može uvelike povećati proizvodnju i uspješnost iste u poljoprivrednom sektoru. To je, naravno, prednost ovog koncepta. Nedostatak je naravno trošak implementacije koncepta. Za primjenu istoga potrebno je nabaviti senzore i uređaje. Također je potrebno dovesti Internet vezu ukoliko se želi upravljati sustavom preko web aplikacije. Sve to se isplati zbog povećanja efikasnosti procesa proizvodnje koja nastaje uvođenjem pametne poljoprivrede. Na kraju poglavlja su navedeni podsustavi koje se čine korisnima za spomenuti kako bi se dobila okvirna slika kako složiti arhitekturu cijele pametne poljoprivrede. Pametna poljoprivreda je trenutno prepoznata kao odlična investicija i sada se počelo aktivno ulagati u to područje. Također, moji interesi su integrirati jedan koncept pametne poljoprivrede u obiteljski posao uzgoja lješnjaka.

4. Poslovni informacijski sustav i tehnološki aspekti

Svijet se razvijao u svim sektorima, a posebno u tehnološkom i softverskom aspektu. Rezultat ovog razvoja bio je poslovni informacijski sustav poznat kao ERP sustav (*eng. Enterprise Resources Planning - ERP*). Abdullah i Ambedkar (2018) spominju da se ERP koristio u industriji kako bi odradio sve njezine funkcije kao planiranje, upravljanje proizvodnjom, upravljanje prodajom, upravljanje financijama, ljudskim resursima itd. Kako je ERP postao koristan i uspješan u industrijskoj primjeni, polako su nicali i proizvođači ERP sustava. Također, kako je ERP pokazivao da je dobar za organizaciju, firme su počele inicirati implementaciju ERP sustava koji će im osigurati konkurentnost na tržištu. ERP sustav olakšava upravljanje nad organizacijom sažimajući sve bitne informacije i vitalne dijelove u jedan sustav koji onda omogućuje i lakše odlučivanje.

4.1. Pojam i definicija

ERP sustav se naveliko spominje u prethodnom poglavlju, no što to stvarno jest ERP. Originalno, spominju Abdullah i Ambedkar (2018), *Enterprise Resource Planning* je nastao 1990. od Garner grupe u svrhu opisa buduće generacije softvera za planiranje materijalnih zahtjeva (*eng. Material Requirements Planning -MRP*). Sam početak ERP sustava se temeljio na upravljanju materijala i zaliha dok se današnji ERP temelji na planiranju resursa koji nisu samo zalihe već ima puno šire značenje.

Postoji mnoštvo definicija ERP sustava koje se zadržavaju u domeni poslovnog i strateškog upravljanja, optimiziranje poslovnih procesa i transakcije te integriranju svih aspekata poslovanja. Također, ERP pomaže u odlučivanju koje proizlazi iz svih informacija koje on sakuplja i sprema. ERP sustav postaje opće prihvaćeno rješenje za integrirani poslovni informacijski sustav. No to ne znači da će sve firme morati implementirati ERP sustav.

ERP sustav se sastoji od raznih modula izrađeni od strane proizvođača ERP-a. Modularnost ERP sustava omogućuje da se klijentima (organizacijama) ponudi široka lepeza mogućnosti. Na taj način se organizacije mogu odlučiti koji moduli im trebaju. To dovodi do toga da se ERP dobro integrira u organizaciju. Abdullah i Ambedkar (2018) spominju neke definicije ERP sustava. Najbliža definicija, uzimajući u obzir definicije autora, je: „ERP je sustav efektivnog planiranja i kontroliranja svih potrebnih resursa za uzimanje, proizvodnju, otpremu i izračun korisnikovih narudžbi u proizvodnom, distribucijskom ili uslužnom poduzeću.“ Što znači da je ERP sustav prisutan u svakom kutku organizacije te svojim

postojanjem pomaže, optimizira i usmjerava organizaciju prema konkurentnoj poziciji na tržištu.

4.2. Prednosti i mane

ERP ima pozitivne i negativne strane koje će se obraditi o ovom odlomku. O'Leary (2000) spominje u knjizi, a ja ću ih samo sumirati i objasniti.

Prednosti:

- Integrira aktivnosti poduzeća
- Koristi najbolje prakse i dodaje nove najbolje prakse
- Omogućuje organizacijsku standardizaciju
- Eliminira asimetriju informacija, jedinstvena baza podataka
- Pruža *online* i *real-time* informacije
- Omogućuje simultan pristup istim podacima za planiranje i upravljanje
- Olakšava intraorganizacijsku komunikaciju i suradnju
- Olakšava interorganizacijsku komunikaciju i suradnju

Integrira aktivnost poduzeća. ERP je skup integriranih aktivnosti koje omogućuju konzistentno rukovanje podacima. U većini slučajeva, postojali su različiti sustavi u različitim odjelima u poduzeću.

Implementira korištenje najboljih praksi. Najbolje prakse omogućuju poboljšanje poslovanja poduzeća. ERP integrira mnoštvo najboljih praksi te implementacija ERP-a zahtijeva i implementaciju najbolje prakse u poslovanju.

Omogućuje organizacijsku standardizaciju. ERP dozvoljava standardizaciju u razini organizacije i na taj način se osigurava da organizacija kreira svoj identitet u svrhu prepoznatljivosti na globalnoj razini.

Eliminira asimetriju informacija. ERP sprema podatke u jedinstvenu bazu podataka i ti podaci su dostupni globalno u organizaciji. Ti podaci se istovremeno koriste i za rad sustava, planiranje i upravljanje.

Omogućuje „online“ i „real-time“ informacije. U staromodnim sustavima se informacija zapisivala na papir i nosila na drugo mjesto te se onda ili prenosila dalje ili unosila u neki digitalni oblik. ERP omogućuje sakupljanje informacija kod izvora te se iste spremaju direktno na sustavu. Takve informacije su dostupne svima u realnom vremenu.

Omogućuje simultan pristup istim podacima za planiranje i upravljanje. ERP koristi jedinstvenu bazu koja omogućuje da su svi podaci dostupni svima koji ih trebaju i u realnom vremenu. Pa tako i onda ERP koristi iste podatke u svrhu upravljanja i planiranja. Ova prednost omogućuje pouzdanije planiranje u odnosu na staromodne sustave.

Dozvoljava intraorganizacijsku komunikaciju i suradnju. Implementacijom ERP-a ima manje konflikata među procesima. Događa se da neki koraci u procesu zahtijevaju neku potvrdu od druge funkcije. Pa u periodu nakon izdavanja zahtjeva za potvrdom i primitka potvrde, proces stoji. I u toj situaciji dvije različite funkcije komuniciraju međusobno.

Dozvoljava interorganizacijsku komunikaciju i suradnju. ERP nudi informacijski oslonac za komunikaciju i suradnju među organizacijama. Organizacije se sve više otvaraju prema partnerima u svrhu komunikacije i suradnje. ERP može dozvoliti takve suradnje.

Mane :

- Skupa je investicija, visok trošak
- Uvodi promjenu radnih navika, može uzrokovati neprihvatanje u organizaciji
- Dugi proces implementacije
- Prilagodba rješenja limitirana
- Često je potrebno provesti reinženjering poslovnih procesa
- Ovisnost o proizvođaču rješenja
- Složenost sustava

Skupa je investicija, visok trošak. Cijeli životni ciklus ERP-a je jedan proces koji traje i nosi ogroman početni trošak. Cijene ERP-a se kreću u od 3 do 10 miliona dolara. Zbog toga je to pravi šok za kapital poduzeća, ali se kasnije taj isti ERP sustav uvelike isplati. Visoki je trošak zbog toga jer je taj cijeli proces dugotrajan.

Uvodi promjenu radnih navika, može uzrokovati neprihvatanje u organizaciji. ERP implementira razne najbolje prakse te ukoliko poduzeće nema implementirane te prakse, onda će morati implementirati zajedno sa ERP-om. Ponekad to znači i promjenu rutine na koju je radnik navikao. Nemali broj puta se može dogoditi da radnik odbija raditi na novi način i na taj način zapravo sprječava da ERP bude uspješno implementirani.

Dugi proces implementacije. Jasno je da ERP sustav ne možete naručiti danas i koristiti ga sutra. To je proces koji traje više mjeseci ili godina. Taj proces započinje s odlukom da će se ERP implementirati, odabirom ERP-a pa sve do implementacije i praćenja rada na kraju.

Kroz cijeli taj proces, netko iz tima proizvođača ERP-a stalno bude u poduzeću da omogući uspješni prijelaz na ERP sustav.

4.3. Povijest ERP-a

Pronaći zadovoljavajuću literaturu je bio gotovo nemogući posao, ali sam našao članak o povijesti ERP koji ću koristiti za ovo cijelo poglavlje koristeći njihove činjenice. Jacobs i Weston (2006) su u članku detaljno prošli tijek od vremena prije MRP-a te do ERP sustava. Neke stvari iz njihovog članka su preskočene ili sumirane.

U 1960.-tima, fokus je na proizvodnji pa se uglavnom radi na optimiziranju proizvodnje. U 1960.-tima se rodio MRP. MRP je u tom razdoblju je bio softver za planiranje i raspoređivanje materijala za kompleksne proizvodne proizvode. Ranije verzije sustava tvorničkog planiranja i kontrole su za vođenje evidencije skladišta koristili magnetne trake. Magnetna traka je sadržavala datoteke proizvoda na skladištu. Traka transakcije se gradila kroz tjedan te se svaki tjedan kreirala glavna traka sa popisom narudžbi temeljenih na izračunatoj količini narudžbe, sigurnih zaliha i ručno izračunatih stanja. U to vrijeme su se ekonomske jednadžbe računale ručno jer tadašnja generacija računala nije mogla izračunati korijen nekog broja. Računanje koristeći magnetnu traku nije bilo optimalno za MRP. Magnetna traka je jednodimenzionalni medij dok je proizvodnja minimalno dvodimenzionalni problem. Postaje puno lakše kada se počinje koristiti RAM memorija koja je bolje rješenje za MRP od magnetne trake. U početku su rješenja za MRP bila ogromna i nespretna. Potrebno je bilo osoblje koje bi održavalo *mainframe* računala (IBM7094, IBM360s, IBM370s). Razvoj računalne tehnologije odvija se paralelno. Pojava RAM memorije je bio veliki korak za računala, a isto tako za poboljšanje MRP sustava. Potrebno je napomenuti da se u to vrijeme nije znalo za baze podataka.

U kasnim 1970-ima važni postaje marketing. To je rezultiralo u primjeni strategija ciljanog oglašavanja sa naglaskom na bolju proizvodnu integraciju i planiranje. MRP se lijepo uspio uklopiti u to razdoblje jer je omogućivao predviđanje, raspoređivanje, nabava i kontrolu trgovine. U to vrijeme MRP postaje standard za upravljanje i kontrolu proizvodnje. 1972. je godina kada se javljaju težnje da se standardiziraju integrirani sustavi za kontrolu proizvodnje i ta godina se smatra godinom kada MRP postaje polako MRP II.

Sredina 1970-tih je razdoblje kada nastaju poduzeća koja će kasnije biti ERP proizvođači. Važnije poduzeće iz tog razdoblja je: SAP (Mannheim, Njemačka; *Systemanalyse und Programmentwicklung*). U tom razdoblju Oracle je predstavio SQL koji će biti koristan za

MRP te isto tako se javlja poduzeće koje će omogućivati usluge administrativnog savjetovanja. U ovom razdoblju su nastali dokumenti koji su objašnjavali ili opisivali MRP kako bi šira publika shvatila o čemu se radi. Sada je već jasno kako MRP funkcionira zahvaljujući dokumentaciji koja je nastala.

1975 godina je godina kada je IBM ponudio sustav za upravljanje industrijom i računovodstvo (eng. *Manufacturing Management and Account System - MMAS*). Ovaj sustav se smatra prethodnikom ERP sustava te je imao integrirana knjiženja i predviđanja ažuriranja na temelju stanja iz transakcija od popisa i produkcije. Ovaj sustav je mogao kreirati proizvodne narudžbe iz narudžbi kupaca. 1978 godine IBM je predstavio IBM system 34 računalu koje je bilo manje i jeftinije od prošlih *mainframe* računala. To računalu je u sebi sadržavalo skup aplikacija poznatih pod nazivom *Manufacturing, accounting and production information and control system (MAPICS)*. Ovaj skup aplikacija je poboljšao MMAS te zapravo pokazao da je sustav korak bliže prema MRP II sustavu. Razvoj hardvera i softvera je toliko napredovao da stariji sustavi više nisu mogli raditi na novijim računalima, ali je omogućen razvoj novih sustava koji će puno bolje funkcionirati i bolje se integrirati. 1978 godine SAP je objavila novu verziju njihovog softvera zvanog SAP R/2 sustav koji je u potpunosti iskoristio prednost tadašnjih *mainframe* računala. Novosti koje se tu javljaju su suradnja među modulima softvera i praćenje narudžbe.

1980-ih, ovaj sustav je postao mnogo isplativiji zahvaljujući tehnološkom napretku, sada su dostupni fleksibilni diskovi sa kapacitetima pogodnima za mala i srednja poduzeća. Sada MRP više ne znači planiranje zahtjeva materijala (eng. *Material requirements planning*), nego počinje značiti planiranje industrijskih resursa (eng. *Manufacturing resource planning*). Eventualno MRP postaje MRP II i na taj način se sada imenuju najnovije mogućnosti sustava. Novosti koje MRP II ima u odnosu na MRP su: upravljanje zalihama, poboljšano izvještavanje trgovine, značajke detaljnog izvještaja o troškovima itd. Srž MRP II je logika MRP-a koja je sada napisana novim kodom.

Izbjegavalo se integriranje sustava od samog početka poslovanja poduzeća upravo zbog toga jer je poduzeće poslovalo sa relativno malim firmama. MRP se tako dizajnirao da je težio ka postajanju standardom te kako bi poduzeća zamijenila postojeće sustave MRP sustavom. Nakon nastanka IBM System 38 i kasnijeg računala IBM AS400 javljaju se skripte te se noviji sustavi temelje upravo na izvršavanju *batch* skripte. U ovom razdoblju je i predstavljeno transakcijsko orijentirano programiranje koje je otvorilo mogućnosti za poboljšanja. Neka od poboljšanja su bila transakcije i praćenje u realnom remenu, bolja podrška odlučivanju i izrada

izvještaja na zahtjev. U ranim 1980-tima, programira se u C-u i koristi se SQL te aplikacije tog doba su se mogle pokretati na računalima od različitih proizvođača. U kasnim 1980 IBM-ov MAPICS je na vrhuncu te se u to vrijeme IBM omogućuje računala Sveučilištima u svrhu edukacije. I u to doba su postojali ponuđači budućeg ERP sustava sa najvećim utjecajem na tržište, kao što su IBM, SAP, J.D. Edwards, Baan, PeopleSoft i Oracle.

Krajem 1980-ih, IBM je ažurirao COPICS softver i predstavio novi akronim CIM koji je označavao računalno integrirana industrija (eng. *Computer integrated manufacturing*). Ovaj novi koncept je ponudio konzistentniju integraciju informacija u poduzeće. CIM okvir je imao 3 razine podrške: najviša razina je podržavala funkcionalna područja i uključivala marketing, inženjering i istraživanje, planiranje proizvodnje, upravljanje postrojenjem, fizička distribucija i poslovno upravljanje, a na razini ispod ove se nalazio sloj podrške koji je uključivao administrativnu podršku, podršku razvoja aplikacije i podršku u odlučivanju, a najdonji sloj je bio serija srži aplikacija uključujući baze podataka i alata za komunikaciju i prezentaciju. Glavna značajka ovog softvera je da je prisutan u svim dijelovima poduzeća odnosno dobro je integriran. I sada je započeo put transformiranja iz MRP, pa MRP II i preko CIM do ERP sustava. 1989. godine je objavljena knjiga koja je opisivala što nudi MRP II te opisuje MRP II kao standard, ali unatoč tome autor knjige napominje da je standard jer sadrži kompletan set funkcija, ali to ne znači da je to idealan sustav sa svim mogućim funkcijama. MRP II je nazvan standardom jer opisuje najjednostavniji skup alata koji omogućuju MRP II nesmetan rad. Ovime se napominje želja za dodatnim funkcijama koje će tvoriti budući ERP sustav. Najznačajniji komentar na ovu knjigu je da funkcije planiranja i raspoređivanja čine srž oko koje se gradi MRP II sustav i da bez toga MRP II ne može postojati. Na taj način se nekako odredilo što je važno za MRP sustav i što će biti važno za ERP.

U ranim 1990-ima se javlja termin planiranje resursa poduzeća (eng. *Enterprise resource planning - ERP*) ili skraćeno ERP. Gartner grupacija je definirala ERP uključujući zahtjev za evaluacijom opsega da je softver integrirani i kroz i unutar različitih funkcionalnih silosa. Kod IBM-ovog MAPICS-a problem je bio da se knjiženja nisu vidjela u realnom vremenu nego su bila sumirana na kraju mjeseca što je bio nedostatak ovog kriterija. Znači da je nedostajalo suradnje među modulima da se izvrše knjiženja u realnom vremenu.

1992. godina je bila godina kada je SAP objavio proizvod R/3. Glavna razlika od prethodnih ERP sustava je bilo korištenje klijent-server arhitektura. Ta arhitektura je omogućila da se sustav može pokretati na računalima sa UNIX ili Windows NT operacijskim sustavom. Druga prednost ovog sustava je bila mogućnost da se kreiraju programi koji se mogu integrirati

sa SAP R/3 sustavom. Ova arhitektura je bila ozbiljan odmak od računala koja su se koristila do ovog vremena. Do kraja 1990-ih je bilo vidljivo da su ponuđači MRP-a i sada ERP-ova ugrabili svoj dio kolača i do tog razdoblja najveći su bili (od ponuđača sa najvećim udjelom prema manje): Baan, SAP, PeopleSoft, Oracle i J.D. Edwards.

Statistika iz 1999:

- *J.D. Edwards – više od 4700 klijenata sa mjestima u preko 100 država*
- *Oracle ima 41 000 klijenata na svjetskoj razini, i 16 000 u Americi*
- *PeopleSoft-ov alat je korišten od više od 50% tržišta ljudskim resursima*
- *SAP postaje svjetska multinacionalna kompanija i četvrta najveći dobavljač softvera u svijetu. Zapošljava više od 20 500 ljudi u više od 20 država.*
- *Više od 2800 poslovnih sustava od Baan poduzeća su implementirana na otprilike 4800 mjesta u svijetu.*

Kako je prilazilo drugo tisućljeće, održavale su se konferencije i prezentacije bitne za budućnost ERP sustava. Razdoblje između 2000. i 2002. godine je bilo teže razdoblje za ERP i ponuđače ERP-a zbog pojave poznate kao „dot-com balon“. Ta pojava je zahvatila poduzeća koja su se oslanjala na Internet te se ,posljedično, neka zatvorila, a neka izgubila veći dio dionica. Posljedica tog je bila da su od 4 glavnih igrača iz 1999. godine, ostala samo 3. Baan poduzeće je otpalo. Čak se razmišljalo o tome da se spoje da se olakša nastavak poslovanja. Konkretno, o spajanju su razmišljali PeopleSoft i J.D. Edwards. Tijek je bio poljuljan time da je Oracle objavio da preuzima kombinirana poduzeća. Na kraju su PeopleSoft, J.D. Edwards i Oracle postali jedno poduzeće. Tako da je 2005. ERP industrija ostala na 2 glavna ponuđača: Oracle i SAP.

Nakon 2000. godine, nadograđuje se ERP sustavi novim funkcionalnostima te nastaju podvrste ERP-a koje će se ukratko opisati u slijedećim poglavljima. Isprva ERP se fokusirao na poduzeće i procese unutar granica poduzeća. Javlja se potreba za rješavanjem nekih integracijskih problema pa se namjerava proširiti ERP sa novim funkcionalnostima. Uvodeći ideju o proširenju, potrebno je na neki način naznačiti verzije ERP-a jednoznačno. Pa se onda osnovni ERP još naziva i ERP 1.

Vasilev (2013) spominje novosti koje donosi ERP 2 u odnosu na ERP 1, a to su: upravljanje odnosima s dobavljačima (eng. *Supplier Relationship Management - SRM*), upravljanje životnim ciklusom proizvoda (eng. *Product Lifecycle Management - PLM*), napredno planiranje i optimizacija (eng. *Advanced Planning and Optimization - APO*). ERP

sustav dobavljače smatra kao dobro integrirane u lanac dobavljača, a ne kao natjecatelje, Proširivanje ERP sustava se uobičajeno naziva prošireni ERP (eng. *Extended Enterprise Resource Planning - EERP*). Ciljalo se u ERP 1 implementirati i koncept ESOA (eng *Enterprise Service Oriented Architecture*).

ERP 3 se ne fokusira samo prema stvarnim klijentima već i prema potencijalnim klijentima. Dalo bi se razlučiti da su integrirane informacije iz tržišta. Neka od glavnih funkcionalnosti, kaže Vasilev (2013), je proširena analiza tržišta. Pod tu analizu može ići i sakupljanje informacija o IP adresama te izvještaj na temelju geografske lokacije IP adrese, analiza koji preglednik i koji uređaj se koristio za pregledavanje ERP-ove web stranice. Prijelaz iz ERP 2 sustava u ERP 3 sustav zahtjeva prethodni rad analitičara sustava i dizajnera sustava kako bi se definiralo što je organizaciji potrebno i kako najbolje iskoristiti mogućnosti koje nudi ERP 3.

Uz istraživanje tržišta, predlaže se integracija društvenih grupa i društvenih web stranica koje primarno služe u svrhe promoviranja proizvoda ili usluge. Međutim kako je taj segment previše nestrukturiran i previše raznolik za integriranje u ERP sustav, nije toliko ni popularan. Također je korisna stvar integrirati poslovnu inteligenciju i upravljanje znanjem. Ti moduli mogu uvelike dodati vrijednost na ukupnu sliku ERP sustava.

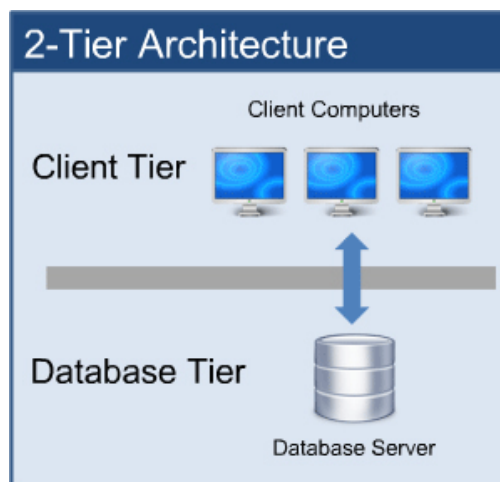
Zaključuje se da je ERP 1 fokusiran na samo poduzeće i isključivo unutar granica istog, dok se prošireni ERP koristi i informacijama izvan granica poduzeća te se fokusira na navedenu domenu. Analizom ERP 2 i ERP 3 zaključuje se da ERP 2 integrira klijente i dobavljače u lanac zaliha, a ERP 3 koristi nove tehnologije kako bi povećao dodanu vrijednost koju nudi.

4.4. Arhitektura i moduli

Arhitektura ERP-a govori o tome kako je riješen razmještaj sustava u organizaciji. Kako su razmješteni klijenti i serveri. Uglavnom se radi o distribuiranom sustavu u kojem su serveri centralizirani, a klijenti na više različitih lokacija kroz poduzeće. Klijenti nemaju informaciju o arhitekturi ERP-a sa strane korištenja sustava, jer se on prezentira obično kroz neku aplikaciju i korisničko sučelje. Kod ERP-a je prisutna klijent/server arhitektura koja prezentira povezivanje između baze podataka, procesiranja i prezentiranja. Abdullah i Ambedkar (2018) spominju dva tipa klijent/server arhitekture koje se obično implementiraju te će biti objašnjene u nastavku.

4.4.1. Implementacija drugog stupnja

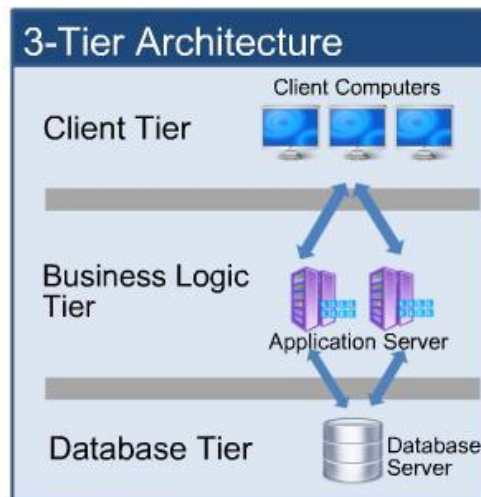
Slika 4.4.1.1. prikazuje ERP arhitekturu drugog stupnja. U arhitekturi drugog stupnja postoje dvije strane: klijenti i server koji obično sadrži aplikaciju i bazu podataka. Server u ovom slučaju i obrađuje stvari vezane za aplikaciju i bazu podataka istovremeno, dok se klijent i dalje fokusira samo na prezentiranje podataka i prosljeđivanje korisničkog unosa do servera. Moguće je izvesti taj arhitekturni tip sa više servera i klijenata distribuiranih kroz više lokalnih veza i veza šireg područja, a pritom odgovornosti svake strane ostaju iste.



Slika 4.4.1.1. Arhitektura drugog stupnja (Izvor: Abdullah i Ambedkar. 2018, str. 4)

4.4.2. Klijent/server implementacija trećeg stupnja

Za razliku od arhitekture drugog stupnja, u arhitekturi trećeg stupnja su funkcije baze podataka i aplikacije razdvojene. Ovaj tip arhitekture se koristi kod većih ERP sustava. Ovdje imamo 3 strane: klijenta, sloj poslovne logike i sloj baze podataka. U sloju poslovne logike se nalaze aplikacijske funkcije koje vode brigu o izvršavanju logike i algoritama koji se tiču aplikacije, dok u sloju baze podataka se rade operacije sa bazom podataka i vraćaju se gotovi podaci koji sloj poslovne logike uzme i prikaže klijentu. Specifično za ovu arhitekturu je da klijent nikada ne komunicira sa bazom podataka direktno, već se sve radi preko sloja poslovne logike. To je način kako se rastereti server podjelom odgovornosti. Arhitektura je prikazana na slici 4.4.2.1.



Slika 4.4.2.1. Arhitektura trećeg stupnja ERP sustav (Izvor: Tallam A., 2013)

4.4.3. Moduli ERP-a

ERP sustav je po prirodi modularan. Moduli se mogu kombinirati kako bi sustav odgovarao najbolje klijentu. Odabir modula uvijek je takav da čim bolje odgovara poduzeća koje ga implementira. Abdullah i Ambedkar (2018) spominju osnovne module u ERP sustavu. Osnovni moduli ERP sustava su:

- **Financije i računovodstvo**
 - Modul upravlja kompletnim priljevom i odljevom novca ili kapitala. Ovaj modul prati sve transakcije povezane s računovodstvom kao što su troškovi, bilanca, knjigu računa, proračuni, bankovne izjave, računi, upravljanje porezom itd. Financijsko izvještavanje je lagan zadatak za ovaj modul ERP sustava. Bilo koji financijski podaci, koji su potrebni za vođenje poslovanja, su dostupni u modulu financija.
- **Skladište**
 - Modul uključuje funkcionalnosti povezane sa skladištem kao što su kontrola skladišta, izvještavanje iskorištavanja zaliha itd. Proizvodi se modu identificirati po jedinstvenom serijskom broju. Koristeći te jedinstvene brojeve, sustav skladišta može pratiti proizvod i slijediti njegovu trenutnu lokaciju u poduzeću. Što znači ako je kupljeno 100 tvrdih diskova onda koristeći sustav skladišta se može pratiti koliko tvrdih diskova je instalirano, gdje su instalirani i koliko tvrdih diskova je preostalo.
- **Proizvodnja**

- Ovaj modul se sastoji od funkcionalnosti kao što su planiranje proizvodnje, praćenje dnevne proizvodnje, predviđanje napretka proizvodnje i izvještavanje stvarne proizvodnje. On isto pruža veliku pomoć za proizvodnu industriju, za dostavu proizvoda itd.
- **Prodaja**
 - Sve transakcije prodaje su upravljanje od strane modula prodaje u ERP-sustavu pa, proces prodaje uključuje procese kao prodajni upiti i analiza upita i rukovanje, izrada nacрта, prihvatanje prodajnih narudžba, izrada računa za prodaju sa pravilnim oporezivanjem, slanje materijala ili usluge, praćenje redoslijeda svih prodaja u tijeku.
- **Ljudski resursi**
 - Jedna od važnijih funkcionalnosti u ovom modulu je sustav plaća koji pomaže u upravljanju plaćama, izvještajima o plaćama također pomaže timu ljudskih resursa u efikasnom upravljanju ljudskim resursima. Također, pomaže u upravljanju informacijama zaposlenika, praćenje zapisa zaposlenika kao što su osvrta učinka, oznake, opisi poslova, matrica vještina, praćenje vremena i prisutnosti. Također može uključivati troškove putovanja. Praćenje edukacije zaposlenika se također može upravljati ERP sustavom.
- **Upravljanje lancem opskrbe**
 - Modul upravljanja lancem opskrbe upravlja tijekom proizvodnih artikala od proizvođača do kupca i od kupca do proizvođača. Uobičajene uključene uloge su proizvođači, distributeri, trgovci itd. Upravljanje lancem opskrbe uključuje upravljanje ponudom i potražnjom, reklamacije i proces zamjene, praćenje dostave i transporta itd.
- **Upravljanje odnosima s klijentima**
 - Upravljanje odnosima s klijentima pomaže poboljšati performanse prodaje kroz bolju službu za korisnike i uspostavljanje zdravih veza sa klijentima. Modul za upravljanje odnosima s klijentima se može integrirati s modulom prodaje kako bi se povećale mogućnosti prodaje. Svi spremljeni detalji o klijentu su dostupni u modulu upravljanja odnosima s klijentima. Modul upravljanja odnosima s klijentima pomaže u upravljanju i praćenju detaljnih informacija o klijentu poput povijesti

komunikacije, poziva, sastanaka, detalja o kupnjama, trajanje ugovora itd.

4.5. Faze implementacije ERP-a

ERP kao sustav ima svoj životni ciklus. Shatat A. (2015), Ijaz A. i sur (2014), Jagoda K. i Samaranayake P. (2017) te Erazo J., Arboleda H. i Pino F. J. (2017) su neki od autora znanstvenih radova koji identificiraju 3 faze životnog ciklusa ERP-a:

1. Predimplementacijska faza
2. Implementacijska faza
3. Postimplementacijska faza

Do današnjeg dana se smatralo da su ove tri faze zapravo faze životnog ciklusa ERP sustava. Međutim nakon čitanja članka autora Huang T. i Yasuda K. (2016), shvatio sam da zapravo njihovo istraživanje ima puno većeg smisla. Oni u svome radu opisuju kako životni ciklus ERP-a nisu te faze, koje sam spomenuo, već životni ciklus ERP-a počinje u postimplementacijskoj fazi. Postimplementacijska faza se smatra fazom kada je ERP sustav službeno pokrenuti ili „rođen“, a ostale faze su Huang T i Yasuda K (2016) kategorizirali pod nazivom: *Realizacija ERP-a*. Oni su te faze nazvali fazama implementacije ERP-a

U tablici 4.5.1 je strukturirani prikaz faza, podfaza i procesa u jednom životnom ciklusu ERP-a temeljeni na pročitanim znanstvenim člancima koji će se spominjati u nastavku prilikom obrade svih faza implementacije ERP-a. Dodana je nova faza: *Faza odluke*. Smatram da je ta faza potrebna jer je to početna točka svake organizacije koja razmišlja o uvođenju ERP-a. Potrebno je prikazati tu aktivnost kao jednu od faza implementacije ERP-a. U tablici 4.5.1 je također naznačeno koje faze pripadaju *Realizaciji ERP-a*, a koje *Životnom ciklusu ERP-a*.

Faze		Podfaze	Opis
Realizacija ERP-a	Faza odluke (Prelazak na ERP)	Prepoznavanje inovacije	<ul style="list-style-type: none"> • prepoznavanje ERP-a na tržištu i problemi koje ERP rješava • analiza značaja ERP-a za poduzeće • odabir načina implementacije • analiza troškova implementacije i ROI • odluka o implementaciji
		Pokretanje implementacije	<ul style="list-style-type: none"> • podizanje svijesti o međusobnoj komunikaciji i suradnji • osnivanje projektnog tima • educiranje projektnog tima o procesu implementacije ERP-a • definiranje budžeta implementacije i potrebnih dokumenata
	Predimplementacijska faza	Odabir ERP-a	<ul style="list-style-type: none"> • definiranje povlastica i utjecaj uvođenja ERP sustava u organizaciju • pokretanje analize troškova i povlastica uvođenja ERP sustava • analiza potreba organizacije • definiranje korisničkih zahtjeva • sakupljanje informacija o proizvođačima ERP sustava • odabir prikladnog ERP sustava (lista potreba poduzeća) njihove konzultantske usluge
		Projektno planiranje	<ul style="list-style-type: none"> • definiranje ciljeva, vizija i jasnih ishoda projekta • odabir projektnog menadžera

			<ul style="list-style-type: none"> • uspostavljanje upravnog odbora • definiranje obujma projekta • planiranje rasporeda projekta • definiranje strategije uvođenja ERP sustava • definiranje resursa, timova i sredstva za projekt • odabir i strukturiranje projektnog tima uzimajući u obzir tehničko i poslovno iskustvo • obuka projektnih timova • priprema plana implementacije
		Dizajn ERP-a	<ul style="list-style-type: none"> • analiza trenutnih poslovnih procesa kako bi se utvrdila potrebna razina reinženjeringa poslovnih procesa • analiza razlika između definiranih procesa u ERP sustavu i njihovih definicija u organizaciji • izvođenje reinženjeringa poslovnih procesa u svrhu dobivanja povlastica ERP tehnologije • izrada prototipa u svrhu testiranja dizajna sustava i integraciju novih procesa i sustava • prilagodba ERP sustava
	Implementacijska faza	Instalacija ERP-a	<ul style="list-style-type: none"> • instalacija ERP sustava
		Konfiguracija ERP-a	<ul style="list-style-type: none"> • konfiguracija ERP sustava
		Obrada i čišćenje podataka ERP-a	<ul style="list-style-type: none"> • izvođenje čišćenje podataka i pretvorbu podataka za implementirani ERP
		Testiranje ERP-a	<ul style="list-style-type: none"> • ispunjavanje testnih slučajeva sa stvarnim podacima

Životni ciklus ERP-a			<ul style="list-style-type: none"> • izvođenje sustavskog testa, korisničkog testa, testa prihvatanja i testa integracije
		Službeni početak korištenja ERP-a	<ul style="list-style-type: none"> • korištenje ERP-a u poslovanju
	Postimplementacijska faza	Upoznavanje	<ul style="list-style-type: none"> • edukacija korisnika
		Korištenje	<ul style="list-style-type: none"> • korištenje ERP-a • održavanje ERP-a
		Poboljšavanje	<ul style="list-style-type: none"> • napredno korištenje ERP sustava
		Otkazivanje	<ul style="list-style-type: none"> • ne odgovara više na izazove poduzeća • zastarjela tehnologija i sustav

Tablica 4.5.1 Strukturirani prikaz faza, podfaza i procesa implementacije ERP-a (autorski rad temeljen na pročitanim člancima)

4.5.1. Faza odluke - Prelazak na ERP

Prelazak na ERP je trenutak kada poduzeće počinje razmišljati o ERP sustavu te postaje svjesno istoga, ali se svi koraci ove faze baziraju na analizi poduzeća i spremnosti poduzeća da kreće u realizaciju uvođenja ERP-a. Pronalaženja članaka za ovu fazu nije bilo uspješno, pa sam temeljem znanja iz kolegija ERP sustavi i pročitanih članaka došao do nekog prijedloga podfaza i procesa koji bi se trebali odraditi u toj fazi uvođenja ERP sustava. Ova faza je označena sivom bojom u tablici 4.5.1 upravo zbog toga jer nije u potpunosti uzeta iz znanstvenog članka već se zaključivalo i procesi su samo prijedlog kako bi se trebala provoditi ova faza.

Poduzeće mora pomno razmotriti sve opcije i odlučiti hoće li implementirati ERP ili ne, jer jednom kada se uđe u taj proces, poduzeće mora moći snositi troškove te uložiti dodatne resurse kako bi proces implementacije uspješno završio.

O'Leary D. E.(2000, str 89 – 100.), Chofreh A. G., Goni F. A. i Jofreh M. G.(2011) te Al-Mashari M. (2003) neki su od autora koji naglašavaju kako se poduzeće odlučuje za uvođenje ERP-a. Neki direktno neki indirektno u nekim drugim fazama u tom procesu. Ističu autori kako poduzeće mora znati u što se upušta analizama. Najviše se ističe važnost komunikacije tijekom procesa.

U nastavku slijede motivacije koje potiču poduzeća na razmišljanje o odluci hoće li uvesti ERP sustav ili ne.

Tehnološke motivacije

O'Leary D. E.(2000, str 89 – 100.) spominje tehnološke motivacije koje vrlo vjerojatno nakon 18 godina od pisanja knjige više ne vrijede. Takve ću samo preskočiti jer nema smisla spominjati takve motivacije. Neki od motivacija, koje spominje autor, su postojeći sustavi loše kvalitete, nabavka koja se teško integrira i mjerenje tehnoloških motivacija.

Postojeći sustavi loše kvalitete

Poduzeće često poseže za ERP-om zbog problema i loše kvalitete sustava koji koriste. Često ti sustavi budu različite aplikacije bez ikakve povezanosti što nemali broj puta uključuje korištenje papira kao sredstvo prijenosa podataka iz jedne aplikacije u drugu. Ili se može dogoditi da se sustav krpao sa različitim zakrpama koje s vremenom rade samo probleme. ERP je svakako dobro rješenje jer je jedinstven za cijelo poduzeće.

Nabavka koja se teško integrira

Ovdje nastupa problem kada poduzeće ima više različitih podružnica i softvera koji svaki komunicira na svoj način. Tada postaje teško sve te aplikacije spojiti u jednu i očekivati da će sve raditi bez problema. Poduzeća posežu za ERP-om zbog njegove konzistentnosti i jedinstvenosti tako da sve podružnice nekog poduzeća mogu koristiti isti sustav te na taj način i raspolagati sa stvarnim podacima u realnom vremenu.

Mjerenje tehničkih motivacija

Tehničke motivacije su uglavnom mjerljive na način postoji ili ne postoji u sustavu odnosno može se jednostavno odgovoriti sa da ili ne. Također te tehničke motivacije nekada ne pružaju adekvatnu dokumentaciju povezanu sa konfiguriranjem sustava.

Motivacije poslovnih procesa

Za razliku od tehnoloških motivacija, motivacije poslovnih procesa koje navodi O'Leary D. E.(2000, str 89 – 100.) su suvremenske što znači da ne ovise o razdoblju. U nastavku slijede sve motivacije koje spominje autor.

Smanjivanje osoblja i troškova IT-a

Uvođenje ERP-a obično označava smanjenje osoblja pogotovo u računovodstvu, ali može se odnositi i u cijelom poduzeću. Što zapravo znači da ERP mijenja poslove koji su se radili neki puta ručno i on to sve sada odradi automatski

Poboljšavanje produktivnosti

Nije strano da ERP sustavi broje klijente upravo zbog ove motivacije. Poduzeća su često motivirana poboljšati određeni aspekt produktivnosti te zato se odlučuju za uvođenje ERP sustava.

Zatvaranje financijskog ciklusa

S obzirom da su pravovremene i potpune informacije ključne za pouzdano donošenje odluka, većina firma motivaciju u uvođenju ERP-a pronalazi u skraćivanju procesa zatvaranja financijskog ciklusa.

Mjerenje motivacija poslovnih procesa

Za razliku od tehničkih motivacija, ove motivacije uglavnom se mogu mjeriti na način jesu ili nisu isplative. Obično one uključuju i dodatne mjerljive ciljeve, a predvidljivost mjerenja ovisi o načinu kako je implementiran ERP sustav.

Strategijska motivacija

O'Leary D. E.(2000, str 89 – 100.) navodi da se strategijska motivacija fokusira na poboljšanje korisničkog odgovaranja i kvalitete uopće. Često se ERP uzima kao podloga za elektroničku trgovinu. Drugačije nego poboljšavanje poslovnih procesa ili transakcija.

Kompetitivna motivacija

O'Leary D. E.(2000, str 89 – 100.) spominje da je kompetitivna motivacija koja potiče poduzeće da implementira tehnologiju ili praksu koju implementira njegova konkurencija te na taj način želi održati svoju kompetentnost i poziciju na tržištu. U literaturi se takva motivacija referira sa „jer konkurencija ima“. Postoje dva ishoda ove motivacije:

1. poduzeće implementira jer ima konkurencija
2. poduzeće pokušava shvatiti kako bi tehnologija ili praksa utjecala na njezino poslovanje te koju sličnu tehnologiju ili praksu bi trebalo uvesti za slični efekt

~ ~ ~

O'Leary D. E.(2000, str 89 – 100.) navodi kako se navedene motivacije mogu koristiti da se odabere pristup dizajnu ERP-a. Kada se radi o motivaciji poslovnih procesa i strategijskoj motivaciji, lako je odrediti pristup dizajnu dok u slučaju kompetitivne i tehnološke motivacije nije moguće jer ne postoje konkretni pristupi dizajnu ERP-a za takve motivacije.

Uz motivacije, O'Leary D. E.(2000, str 89 – 100.) navodi da, postoje monetarni i nemonetarni razlozi. Drugim riječima, neka poduzeća na prvo mjesto stavljaju cijenu projekta uvođenja, a neka stavljaju na prvo mjesto strateške ciljeve i kako će taj projekt održati njihovu poziciju na tržištu.

U nastavku slijedi prijedlog podfaza koje bi se trebale razmatrati u tijeku odlučivanja za ili protiv uvođenja ERP sustava u poduzeće.

4.5.1.1.Prepoznavanje inovacije

Prepoznavanje inovacije je podfaza u kojoj poduzeće prepoznaje inovaciju, u našem slučaju to je ERP sustav. U toj podfazi, poduzeće postaje svjesno ERP sustava i koje su povlastice njegovog uvođenja. Također, se objasni koje probleme on rješava za poduzeće. Nakon toga je nužno da poduzeće vidi na koji bi način implementiralo ERP sustav što uvelike određuje iznos troškova kao slijedećeg koraka u postizanju odluke. Dodano je i računanje ROI-a (*eng. Return of Investments*) kako bi poduzeće dobilo sliku kada bi se cijeli taj proces isplatio.

U nastavku slijedi popis koraka:

1. prepoznavanje ERP-a na tržištu i problemi koje ERP rješava
2. analiza značaja ERP-a za poduzeće
3. odabir načina implementacije
4. analiza troškova implementacije i ROI
5. odluka o implementaciji

Prepoznavanje ERP-a na tržištu i problemi koje ERP rješava

Poduzeće je svjesno ERP sustava te se u ovom koraku radi prezentacija o ERP-ovima koji postoje. U tim prezentacijama se definira ERP te se prikazuje teorijski dio procesa implementacije po fazama i što koja faza radi. Predstavljaju se načini implementacije i okvirni period trajanja implementacije prema poznatoj statistici. Ključno je da prezentacija prikazuje okvirne cijene ERP sustava. U prezentaciju se uključuju i povlastice koje dolaze sa ERP-om i problemi koje on rješava. Ovaj korak služi za osvještavanje poduzeća o mogućnostima ERP sustava.

Analiza značaja ERP-a za poduzeće

U ovom koraku se fokus usmjerava na poduzeće. Prethodni korak je bio općenitog karaktera i odnosio se samo na informiranje poduzeća. Sada se izvodi analiza koji značaj bi imao ERP sustav za poduzeće. Koriste se procesi iz poduzeća i trenutna situacije te što se očekuje od ERP-a. Za analizu se značaja ERP-a za poduzeće. Nakon toga se kreće na odabir načina implementacije.

Odabir načina implementacije

Ovaj korak se odnosi na odluku hoće li se ERP sustav implementirati unutar poduzeća (*eng. In-House*) ili će se koristiti preko vanjskih suradnika (*eng. Outsourced*). O'Leary D. E.(2000, str 100) spominje način implementacije preko vanjskih suradnika te objašnjava da se na taj način poduzeće oslobađa infrastrukture (hardvera i softvera) jer je infrastruktura kod vanjskog suradnika, a poduzeće samo ima pristup tom ERP sustavu. Na taj način se poduzeće oslobađa troškova infrastrukture te sveukupna implementacija košta puno manje. To je prednost koju autor spominje, a nedostatak tog načina implementacije je da taj sustav nije nužno iskrojen po potrebama poduzeća.

Analiza troškova implementacija i ROI

Sada se radi okvirno predviđanje očekivanih troškova implementacije temeljenih na prezentacijama i informacijama dobivenih iz prezentacije. Kako bi poduzeće dobilo okvirnu sliku. Također se provodi ROI analiza na temelju tih informacija kako bi se uvidjelo kada će se ERP isplatiti. Nakon toga slijedi odluka o implementaciji.

Odluka o implementaciji

Odluka o uvođenju ovisi o motivacijama koje su navedene na početku poglavlja te o znanjima i informacijama koje poduzeće dobije izvođenjem prethodnih koraka. Ovaj korak je ključan jer o njemu ovisi hoće li poduzeće krenuti u stvarni proces realizacije ili neće. Ukoliko poduzeće odluči implementirati ERP tada se nastavlja proces na slijedeću podfazu.

4.5.1.2. Pokretanje implementacije

Pokretanje implementacije je faza koja je ključna za korake prije samog početka prave implementacije ERP sustava. U ovoj podfazi pripremaju se preduvjeti za prelazak na slijedeću fazu implementacije. U ovoj podfazi su slijedeći koraci:

1. podizanje svijesti o međusobnoj komunikaciji i suradnji
2. osnivanje projektnog tima
3. educiranje projektnog tima o procesu implementacije ERP-a
4. definiranje budžeta implementacije i potrebnih dokumenata

Podizanje svijesti o međusobnoj komunikaciji i suradnji

Kao što je spomenuto, važnost komunikacije tijekom implementacije ERP-a se ističe u znanstvenim člancima. Kritično je da se komunikacija u horizontali i vertikali odvija kako bi se postigao najbolji mogući rezultat. Ovim korakom bi trebalo osvijestiti ljude da će nedostatak komunikacije ugroziti uspješnost projekta implementacije.

Osnivanje projektnog tima

Projektni tim je ključan za izvođenje implementacije ERP-a jer je to projektni pothvat. Projektni tim je zadužen za izvođenje projekta bez da se narušava normalan rad poduzeća.

Educiranje projektnog tima o procesu implementacije ERP-a

Ovaj korak je nužan kako bi se projektni tim upoznao sa koracima tijekom implementacije ERP-a..

Definiranje budžeta implementacije i potrebnih dokumenata

Definicija budžeta implementacije je ključno kako bi se moglo mjeriti koji je napredak u projektu i je li se zadržao unutar granica budžeta. Definiranje svih potrebnih dokumenata bi bilo ključno da se zna točno što je potrebno kreirati u kojoj fazi.

4.5.2. Predimplementacijska faza

Predimplementacijska faza je razdoblje koje se odvija prije implementacije ERP-a u poduzeće. Početak faze označava odabiranje ERP-a u koji ulazi nekoliko različitih metoda analiza koje se mogu koristiti prilikom definiranja koji su zahtjevi poduzeća. Nakon odabira ERP-a, poduzeće kreće u projektno planiranje u kojem se definiraju parametri projekta kao što su trošak, trajanje, opseg i slično. Zadnji korak je dizajn i prilagodba koji uključuje analizu poslovnih procesa te prilagodbu samog ERP-a prije početka implementacijske faze.

4.5.2.1. Odabir ERP-a

Odabir ERP-a uključuje analize koje će dati konačnu listu potreba koja će se kasnije koristiti za nastavak procesa implementacije ERP-a. U ovom dijelu će se prikazati analize koje se mogu koristiti za izradu liste potreba. O'Leary D. E. (2000, str. 102 – 106) navodi analizu zahtjeva kao jednu od analiza pomoću kojih se može odabrati ERP.

Analiza zahtjeva

O'Leary D. E. (2000, str. 102 – 106) spominje da je analiza zahtjeva lista stavki koje poduzeće smatra važnim prilikom implementacije ERP -a. To je lista zahtjeva koji se trebaju razmatrati kako bi poduzeće ostvarilo stratešku prednost te poboljšalo poslovanje. Često se događa da lista zahtjeva postane dugačka, a to onda rezultira sa većim troškovima i duljim trajanjem analize prije odabira ERP-a. Često se događa da i ERP proizvođači ignoriraju poduzeće koje im pošalje ogromnu listu zahtjeva. Postoji dva načina kako se može definirati ova lista potreba:

1. Ručnim zapisivanjem svih zahtjeva u obliku liste koji se onda šalju ERP proizvođaču.
2. Korištenje demo ERP-a kako bi se stekao osjećaj koje su njegove mogućnosti. Ovo se odvija kroz dvije faze. U prvoj fazi se dobije demo ERP na kojem klijent isproba osnovne funkcije te nakon toga se prelazi na drugu fazu koja uključuje prikazivanje neke konkretne funkcije koja je zanimljiva klijentu. Ovaj pristup je koristan za shvaćanje razlika između aplikacije i stvarnog stanja u poduzeću.

Prednosti	Mane
<ul style="list-style-type: none"> • zahtjevi se mogu koristiti za provjeru softvera i provjeru koliko određeni softver zadovoljava te zahtjeve • izrada liste potreba potiče intrakomunikaciju i interkomunikaciju u poduzeću • otkriva područja gdje su potrebna poboljšanja • nudi dokaze da proizvod koji kupuju zadovoljava ove zahtjeve • poduzeće ima doka z da njegovi modeli, procesi i aktivnosti funkcioniraju. 	<ul style="list-style-type: none"> • vremenski zahtjevno • visok trošak • prevelika lista zahtjeva može rezultirati ignoriranjem ERP-proizvođača • zahtjevi mogu prisiliti poduzeće na zastarjele načine rada. Umjesto toga bi bilo bolje da se provede reinženjering procesa • zahtjevi mogu biti razrađeni na način da firma cilja na neprovjerene prakse.

Tablica 4.5.2.1.1. Prednosti i mane analize zahtjeva (Izvor: O'Leary, 2000, str. 104 – 105)

Prednosti i mane analize zahtjeva napisani su u tablici 4.5.2.1.1. O'Leary (2000, str. 102 – 106) ističe kako zaposlenici iz različitih pozicija iznose različite zahtjeve. Također je ključno uzeti u obzir da zahtjevi imaju različitu zrnatost. Što znači da neki zahtjevi mogu obuhvaćati cijelu domenu dok drugi mogu specificirati točno određenu akciju.

Gap analiza

Druga metoda koju O'Leary (2000, str 106.) je gap analiza. Gap analiza je prikaz razlika između sadašnjeg stanja i željenog stanja. Sadašnje stanje u gap analizi se označava sa „as is“ stanje, a željeno stanje se označava sa „to be“ stanje. Ova analiza daje uvid u promjene koje će se morati uraditi te isto tako omogućuje lakše upravljanje troškovima. Postoje dvije vrste analiza željenog stanja:

1. tehnološki neovisna analiza
2. tehnološki određena analiza

Ukoliko je analiza tehnološki neovisna, to označava da se to stanje razvijalo van svih softverskih paketa, a tehnološki određena analiza je ona koja se kroji prema nekom softverskom paketu. Ako je slučaj da taj softverski paket bude ERP, tada se to stanje obično takva analiza naziva analizom „najbolje prakse“.

Ova metoda nema specifično napisane prednosti i nedostatke, ali se oni mogu zaključiti. Prednosti ove analize jesu da poduzeće postaje svjesno razlika između trenutnog i željenog stanja te poznaje koje su potrebne promjene da se približe željenom stanju. Također u ovoj metodi se ne radi ogromna lista zahtjeva, pa analogno, ova metoda omogućuje bolju suradnju sa ERP proizvođačima. Negativne strane ove metode su nedostatak definicije što točno jest ova metoda i kako se mjeri uspješnost metode.

Ove dvije metode se fokusiraju striktno na funkcionalnosti. O'Leary (2000, str. 108 – 109) spominje da ove metode ignoriraju:

- troškove - troškove analize, kupnje ERP sustava
- vrijeme instalacije
- fleksibilnost ERP-a
- korisničko sučelje odnosno prilagođenost korisniku - koliko je ERP lak za korištenje
- nadogradivost na novu verziju - kako znati kada je dostupna nova verzija i koliko je lako nadograditi na novu verziju
- računalna okolina – ERP sustav nekad treba sučelje prema drugim sustavima što može biti važan faktor (Oracle je implementirao arhitekturu tankog klijenta što je zahtijevalo i sučelje prema drugim sustavima)
- implementacijsko osoblje – kvaliteta i dostupnost osoblja igra važnu ulogu prilikom odabira
- svakodnevno korištenje i održavanje – način učenja i održavanja se može razlikovati od proizvođača do proizvođača te je korisno to isto uključiti u analizu
- funkcionalnost - neka poduzeća mjere funkcionalnost umjesto da mjere zahtjeve koje te funkcionalnosti moraju realizirati.

AHP metoda

AHP metoda je još jedna metoda pomoću koje se može odabirati ERP. Wei CC., Chien CF. i Wang MJ. J. (2004) definiraju AHP metodu na sljedeći način:

AHP metoda je evaluacijska metoda nad više atributa koja uključuje tri faze: dekompozicija, komparativna procjena i sinteza prioriteta.

Wei CC., Chien CF. i Wang MJ. J. (2004) navode korake metode kroz faze:

1. dekompozicija – projektni tim izrađuje AHP hijerarhijski model iz temeljno objektivne hijerarhije

2. komparativna procjena – svaki donositelj odluke koristi uparene komparacije za atribute i alternative kako bi se izvukle matrice procjene na svakoj razini. Matrice sadrže vrijednosti od 1 do 9 kao relativne koeficijente važnosti.
3. sinteza prioriteta – proces uparenih usporedba se ponavlja za svaki atribut u problemu alternativne prioritetizacije koji se temelji na metodi najveće svojstvene vrijednosti. Nakon toga se važnost atributa i globalnu važnost alternativa može dobiti agregiranjem iznosa težina kroz hijerarhiju.

Procedura je prikazana sa primjerom u članku autora Wei CC., Chien CF i Wang MJ. J. (2004) koji ću samo objasniti u nastavku kako bi sve ovo bilo jasnije. Prvi korak je izrada hijerarhije koja na vrhu ima cilj poduzeća koji se dekompozira na specifičnije ciljeve. Nakon toga se uvode karakteristike i uvjeti ERP proizvođača te se isti povezuju sa ciljevima sa kojima su u vezi. Kada se dobije cijela slika kreće izrada matrica procjene koje sadrže koeficijente koji predstavljaju važnost određenog cilja. Koeficijent je broj od 1 do 9. Nakon toga se određenom procedurom izračunaju koeficijenti važnosti po jednom ERP proizvodu te nakon toga se zbrajanjem koeficijenata pojedinačno po ERP proizvodu. Uvidom u konačne koeficijente po ERP proizvodu, odabire se onaj koji je najveći i to znači da je taj ERP zapravo najbolja opcija za poduzeće.

Erazo J., Arboleda H. i Pino F. J. (2017) navode procese koji se odvijaju u fazi odabira ERP-a:

1. definiranje povlasticu i utjecaj uvođenja ERP sustava u organizaciju
2. pokretanje analize troškova i povlastica uvođenja ERP sustava
3. analiza potreba organizacije
4. definiranje korisničkih zahtjeva
5. sakupljanje informacija o proizvođačima ERP sustava
6. odabir prikladnog ERP sustava i njihove konzultantske usluge

Navedeni procesi ilustriraju tijek kako bi ovaj korak u predimplementacijskoj fazi trebao teći. Sada se službeno definiraju povlastice i utjecaj uvođenja ERP. Potrebno je da poduzeće postane svjesno troškova i povlastica koje ERP nosi sa sobom. Tada slijedi pokretanje prethodno spomenutih metoda koje kao izlaz daju listu potreba organizacije koja je zapravo lista prioriteta odnosno što bi ERP sustav trebao uključivati. Vrlo važna stavka u svakom softverskom projektu jesu korisnički zahtjevi. Oni pomažu u prikazu jasne slike što bi korisnici htjeli da taj sustav pruža te kako bi željeli upravljati sustavom. Korisnički zahtjevi su u nekoj mjeri podložni promjenama. Sada je potrebno i sakupiti informacije o dostupnim proizvođačima ERP sustava. Ova aktivnost omogućuje poduzeću da dozna ključne informacije o svim potencijalnim ERP

proizvođačima koje planira razmatrati za kasniju ponudu suradnje. Nakon toga se metodom odabira ERP rješenja provede postupak i kao izlaz tog postupka je prikladan ERP proizvođač kojemu se šalje ponuda. Nakon što oni odgovore na ponudu i izradi se ugovor, tada kreće nastavak procesa uvođenja.

4.5.2.2.Projektno planiranje

Erazo J., Arboleda H. i Pino F. J. (2017) navode procese koji se odvijaju u fazi projektnog planiranja implementacije ERP-a. U nastavku slijedi lista procesa uz manju izmjenju. S obzirom da će se projektni tim morati odabrati već prije procesa odabira ERP-a, to znači da nije potrebno ponovno definirati projektni tim u ovom koraku uvođenja. U ovom koraku uvođenja fokus je na kreiranje projektne dokumentacije i definiranje vodstva u timu. Također se izostavlja odabir eksternih konzultanata s obzirom da se pretpostavlja da će ERP proizvođač nuditi svoje konzultantske usluge. Popis procesa uz izostavljanje odabira projektnog tima te odabira eksternih konzultanata:

1. definiranje ciljeva, vizija i jasnih ishoda projekta
2. odabir projektnog menadžera
3. uspostavljanje upravnog odbora
4. definiranje obujma projekta
5. planiranje rasporeda projekta
6. definiranje strategije uvođenja ERP sustava
7. definiranje resursa, timova i sredstva za projekt
8. odabir i strukturiranje projektnog tima uzimajući u obzir tehničko i poslovno iskustvo
9. obuka projektnih timova
10. priprema plana implementacije

Poduzeće u ovoj fazi definira točno što se očekuje od ovog projekta implementacije što je nužno kako bi svi bili svjesni što se točno očekuje od projekta. Kako je projektni tim već definiran potrebno je samo definirati tko je vođa tima i upravni odbor koji će pratiti napredak projekta. Potrebno je također naznačiti opseg projekta te nekakvi vremenski okvir i raspored odvijanja faza u implementaciji. Ključno je također definirati način kako će se uvesti ERP, postoje dvije strategije uvođenja ERP sustava koje će se objasniti nakon ovog prikaza. Treba biti svjestan svih dostupnih resursa, timova i sredstava. Na taj način će se znati je li projekt uspješno završen i je li završen u određenom roku.

4.5.2.3.Dizajn i prilagodba

Ova podfaza se fokusira na usklađivanje poslovnih procesa sa onima u ERP sustavu i na kraju se i prilagođava ERP sustav kako bi bio spreman za slijedeću fazu. Erazo J., Arboleda H. i Pino F. J. (2017) navode slijedeće korake u ovoj podfazi:

1. analiza trenutnih poslovnih procesa kako bi se utvrdila potrebna razina reinženjeringa poslovnih procesa
2. analiza razlika između definiranih procesa u ERP sustavu i njihovih definicija u organizaciji
3. izvođenje reinženjeringa poslovnih procesa u svrhu dobivanja povlastica ERP tehnologije
4. izrada prototipa u svrhu testiranja dizajna sustava te integraciju novih procesa i sustava
5. prilagodba ERP sustava

Prvo je potrebno da se provede analiza poslovnih procesa da se utvrdi prate li najbolje prakse. Nakon toga se dodatno analiziraju razlike između procesa u poslovanju i procesa u ERP sustavu. Nakon analize razrađuje se plan reinženjeringa procesa da se vidi željeno stanje. Cilj inženjeringa je da se procesi slože kako bi ERP sustav mogao ponuditi najbolje od sebe i unaprijediti poduzeće. Sada slijedi izrada prototipa da se testira dizajn sustava te integracija novih procesa i sustava i na kraju se prilagođava i ERP sustav da zadovolji zahtjeve organizacije i da čim više odgovara procesima u poslovanju.

4.5.3. Implementacijska faza

Nakon provedenih svih potrebnih priprema i podešavanja, vrijeme je da se ERP sustav postavi kod klijenta. U ovoj fazi klijent vidi prvi puta ERP sustav u vlastitom poduzeću. U nastavku će se opisati podfaze koje su dio ove faze. Prije samog početka prve podfaze, je bitno provjeriti je li sve predviđeno odrađeno do sada.

4.5.3.1.Instalacija ERP-a

Provjera je li sklopovlje spremno za proces instalacije te radi li sve ispravno. Instalacija ERP-a se odnosi na donošenje softvera i pripremanje infrastrukture u poduzeću. Obavljanje instalacije te pokretanje softvera.

4.5.3.2.Konfiguracija ERP-a

Dodjeljivanje dozvola koja su potrebne za rad sa sustavom.. Postavljanje konfiguracionih varijabli na konkretne vrijednosti koje vrijede za poduzeće. Na ovaj način se ERP sustav podese da odgovara što je više moguće poduzeću.

4.5.3.3.Čišćenje i priprema podataka

Prijenos podataka iz starog sustava u novi te provjera ispravnosti podataka. Brisanje zaostalih podataka iz predimplementacijske faze.

4.5.3.4.Testiranje ERP-a

Zamjena testnih podataka sa stvarnim podacima. Izvršavanje sustavnog testa, korisničkog testa, testa prihvatljivosti i test integracije. Provjera jesu li uneseni podaci ispravni i radi li sustav ispravno.

4.5.3.5.Službena objava

Ovaj korak samo označava da su posljednje promjene prije službenog početka izvršene. Nakon što su izvršeni prethodni koraci do kraja, službeno se prolazi na slijedeću fazu.

4.5.4. Postimplementacijska faza (životni ciklus ERP-a)

Nakon što se novi sustav uspješno implementirao, sada je potrebno potpuno ga integrirati u sustav. Sada bi trebalo korisnike potpuno upoznati sa sustavom i održavati ga.

4.5.4.1.Upoznavanje ERP-a

Konzultanti educiraju korisnike kako bi bili spremni koristiti ERP sustav. Ovo je također jedan od dužih aktivnosti.

4.5.4.2.Korištenje ERP-a

Korisnici tijekom dužeg perioda korištenja ERP žele naučiti dodatne mogućnosti koje ERP pruža pa se kaže da započinje napredno korištenje ERP-a. Sada korisnici već znaju osnove koje su naučili u prošloj fazi životnog ciklusa ERP-a, ali im to nije dovoljno te žele proširiti svoje znanje.

4.5.4.3.Poboljšavanje ERP-a

Poboljšavanje se odnosi na uklanjanje problema i eventualno se traži podešavanje procesa ako se primijeti da je isti prespor. Također, ERP je dio tehnologije, a tehnologija se mijenja brzo i konstantno, pa je potrebno omogućiti ERP-u da se ažurira redovno kako bi ostao

u fokusu tehnologije i da ne zastari prebrzo. Ovo je faza životnog ciklusa ERP-a koja je zaslužna najviše za odgađanje prelaska na zadnju fazu životnog ciklusa.

4.5.4.4.Otkazivanje ERP-a

ERP ulazi u ovu fazu životnog ciklusa kada poduzeće shvati da ERP više ne rješava izazove koji su stavljeni pred poduzeće. Uočava se da ERP počinje zaostajati u odnosu na aktualnu tehnologiju – postaje zastarjela tehnologija. U ovoj fazi životnog ciklusa, poduzeće je primorano istraživati i otkrivati novu inovaciju koja će zamijeniti ovaj sustav. Kada poduzeće odluči mijenjati sustav, ponovno započinje predimplementacijska faza.

4.6. Strategije implementacije ERP sustava

Nagpal S., Khatri S. K. i Kumar A. (2015) navode da klasifikacija strategija implementacije je puno drugačije od klasifikacije na naglo uklanjanje i fazna implementacija. U nastavku će se spomenuti 2 klasifikacije koje spominju Nagpal S., Khatri S. K. i Kumar A. (2015).

Prva kategorizacija, koju koriste Nagpal S., Khatri S. K. i Kumar A. (2015) i po navodu Parr-a i Shank-a, slijedi:

- Opsežna implementacija (*eng. Comprehensive and Chocolate*) – Karakterizira ju opsežna prilagodba sustava te da su većina poslovnih procesa ERP-a usklađeni sa stvarnim procesima u poslovanju.
- Optimalna implementacija (*eng. Middle Road*) – Najpopularnija strategija koja se fokusira na čim manje prilagodbe sa gledištem da većina poslovnih procesa povezanih sa ERP-om su već primjeri najbolje prakse. Reinženjering procesa igra važnu ulogu kada se neki poslovni procesi moraju uskladiti sa ERP-ovim procesima.
- Ciljana implementacija (*eng. Vanilla*) – Korištena u slučaju da je glavno ograničenje implementacije novac. Karakteristika implementacije je nema ili ima minimalno prilagodbe. Glavni je fokus na konfiguraciji sustava prema zahtjevima poduzeća. Reinženjering procesa se odvija na način da su svi poslovni procesi poduzeća usklađeni sa procesima u ERP-u.

Druga kategorizacija, koju spominju Nagpal S., Khatri S. K. i Kumar A. (2015), je sljedeća:

- Prilagođena implementacija – Srž implementacije je ta da nema definirane procedure implementacije već si klijent može definirati svoj tijek projekta i korake. Ova

implementacija se koristi kod tradicionalne fazne implementacije. Ova implementacija se često koristi u agilnom razvoju.

- Implementacija specifična za proizvođača ERP-a – Glavni proizvođači se zalažu za takav način implementacije. Oni zalažu za interakciju između proizvođača i klijenta. Ovakva implementacija je specifična jer su definirani koraci.
- Implementacija specifična za konzultanta – Mnogo konzultanata se uključuje u implementaciju ERP-a i implementacija se smatra mješavinom implementacije specifične za proizvođača i najbolje prakse koja je rezultat konzultantskog iskustva.

Već dobro poznata tradicionalna klasifikacija, koju spominju Nagpal S., Khatri S. K. i Kumar A. (2015) i O'Leary D. E. (2000, str. 151), se sastoji od dva pristupa:

- Veliki prasak (*eng. Big Bang*)
- Fazna implementacija (*eng. Phased Implementation*)

U nastavku slijedi opisi tradicionalne klasifikacije i implementacija specifičnih za proizvođače ERP-a.

4.6.1. Veliki prasak

O'Leary D. E. (2000, str. 151) spominje da je "veliki prasak" način implementacije u kojem se više modula uvodi istovremeno te taj postupak zahtjeva vrlo mnogo testiranja prije nego se stari sustav potpuno ugasi. Ovaj pristup se često izvodi na svim lokacijama poduzeća. Autor spominje da se ova strategija obično izvodi kroz 3 koraka:

1. svi relevantni procesi i artefakti su odabrani ili razvijeni virtualno i implementirani u softver
2. svi moduli su testirani pojedinačno i njihova sučelja sa ostalim modulima
3. stari sustav je ugašen i novi sustav je službeno u upotrebi

Nakon trećeg koraka i dalje postoje neke manje promjene. U nastavku slijede prednosti i mane ovog načina implementacije, koje je spomenuo O'Leary D. E. (2000, str. 152).

Prednosti

- nije potrebno imati privremena sučelja – „Veliki prasak“ mijenja stare sustave odjednom pa nema potreba za privremenim sučeljima jer se stari sustav ne dira dok se u potpunosti ne promijeni.
- ograničena potreba za održavanjem i revidiranjem starog sustava – Kako se stari sustav u potpunosti mijenja sa novim ERP sustavom, onda nema potrebe za održavanjem

resursa ili mijenjanja starog sustava već se ti resursi mogu investirati u razvoj i testiranje novog sustava.

- manji rizik – „Veliki prasak“ je manje rizičan jer se prijelaz na novi sustav izvodi brzo i uključuje se kompletan tim. Manji je rizik da će zaposlenici otići prije nego završi implementacije i osigurava da su svi potrebni članovi tima uključeni, dok to kod fazne implementacije ne mora biti tako.
- povezanost funkcionalnosti – Kako su moduli u ovom pristupu implementirani odjednom, korisnik ne mora čekati na implementaciju nekih modula koji su potrebni za određenu funkcionalnost.
- nema povratka na stari sustav – S obzirom da se implementira novi ERP sustav i stari se odbacuje, to znači da nema povratka na stari sustav. Poduzeće će morati ostati raditi na novom sustavu iako im možda to ne odgovara.
- kraće vrijeme implementacije – Normalna implementacija ERP-a traje duže vrijeme koje se često broji u godinama. Kako je „veliki prasak“ puno brža metoda implementacije, smanjuje se mogućnost dodatnih promjena zahtjeva, smanjuje se potreba za resursima i osobljem koje mora odraditi implementaciju do kraja.
- manji trošak – Ovaj pristup košta manje kada nema problema prilikom implementacije. Razlog manjeg troška je ograničen rad na starom sustavu jer je fokus na novom ERP sustavu i tijekom implementacije je puno kraći.

Nedostaci „velikog praska“ su suprotni prednostima fazne implementacije.

4.6.2. Fazna implementacija

O'Leary D. E. (2000, str. 152) definira faznu implementaciju kao pristup u kojem se moduli implementiraju pojedinačno ili u grupama i često po jednoj lokaciji istovremeno. Fazna implementacije je sekvencijalna implementacija koja se sastoji od dizajniranja, razvoja, testiranja i instaliranja raznih modula. Za razliku od „velikog praska“, ovaj pristup zahtjeva pridavanje više pažnje starom sustavu kako bi se isti najbolje integrirao u ERP sustav. U nastavku će se preneti prednosti i mane koje je naveo O'Leary D. E. (2000, str. 154).

Prednosti

- vršni zahtjevi za resursima su manji nego kod „velikog praska“ – Metoda „velikog praska“ ima puno veće zahtjeve za resursima jer se sve odvija vrlo brzo. U faznoj implementaciji ti zahtjevi za resursima su raspoređeni kroz duže vrijeme i zato je uglavnom taj pristup bolji za poduzeća sa ograničenim resursima.

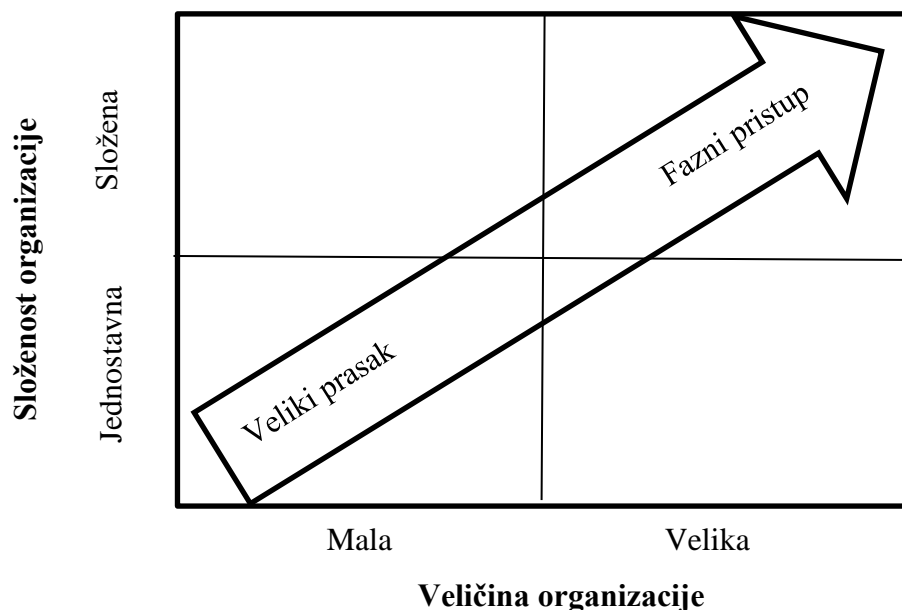
- više se resursa može posvetiti određenom modulu – U ovom načinu implementacije, poduzeće se može fokusirati na određene module obuhvaćeno fazom. Kod „velikog praska“ je potrebno da poduzeće osigura puno više resursa kroz više modula. Poduzeće je obvezno osigurati dovoljno resursa, jer nedostatak resursa može dovesti do neuspješnog projekta implementacije.
- manji rizik – „Veliki prasak“ je poznat kao „sve ili ništa“ pristup znači ako je jedan modul neispravan cijeli projekt propada. Zbog toga se poduzeća uglavnom odlučuju za fanu implementaciju jer je manje riskantna metoda.
- stari sustav je rezerva – „Veliki prasak“ implementacija označava da kada se jednom započinje postupak implementacije, ne postoji način da se poduzeće vrati na prijašnji sustav. Fazna implementacija se fokusira na implementiranje jednog po jednog modula prije nego se alternativa isključi. Fazni pristup nudi mogućnost da se projekt vrati na stari sustav ako ne uspije.
- osoblje dobiva znanje u svakoj fazi – Članovi projektnog tima stječu znanje radeći na projektu te to svoje znanje mogu prenositi na druge faze i koristiti ga za budući razvoj. Na taj način ERP sustav se razvija sa iskusnijim ljudima.
- projektni voditelji mogu prezentirati sustav koji radi – projektni voditelji mogu prezentirati dio projekta koji radi kao dokaz upravi da projekt napreduje. Ovdje se obično daje prioritet modulima koji se najlakše implementiraju.
- vrijeme između razvoja i korištenja je smanjeno – „Veliki prasak“ pristup implementacije je karakterističan po tome što korisnik ne vidi konačan proizvod jedno vrijeme. Fazna implementacija uklanja razlike između razvoja i implementacije, jer se modul odmah nakon implementacije je spreman za korištenje od strane klijenta.

Nedostaci faznog pristupa su zapravo suprotni prednostima „velikog praska“.

4.6.3. Organizacijske karakteristike i tradicionalne strategije implementacije

Odabir strategije implementacije nije samo odabrati neku koja se sviđa većini i implementirati. Postoje organizacijske karakteristike koje dodatno pokazuju strategiju implementacije koja bi se trebala koristiti. Također, O'Leary D. E. (2000, str. 156) navodi da nije lako smisliti analizu koja bi rekla koja je karakteristika sigurni odabir već postoje razne organizacijske karakteristike koje potpomažu odluku. U nastavku slijedi pregled organizacijskih karakteristika koje spominje autor.

Veličina i složenost organizacije



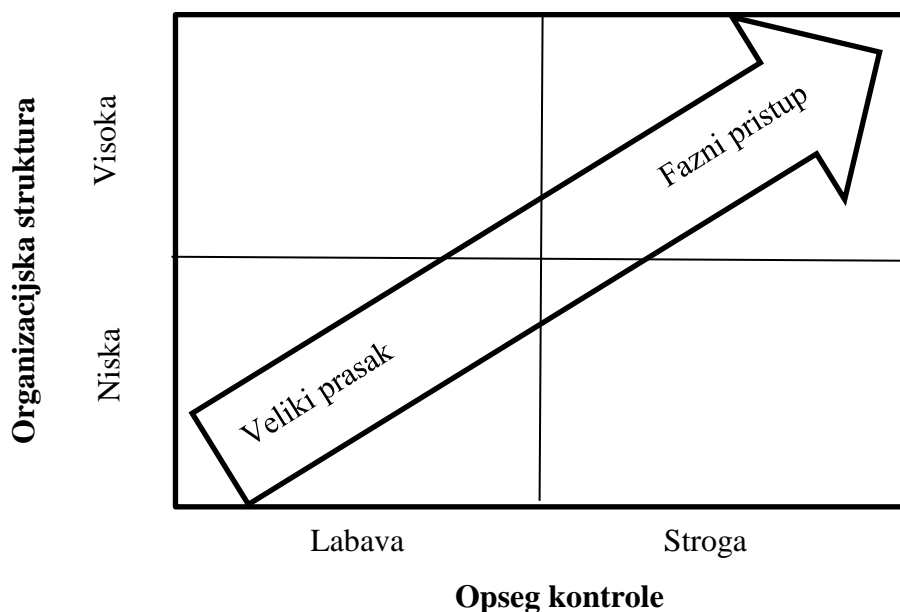
Slika 4.6.3.1 Odnos između veličine i složenosti organizacije te strategije implementacije (Izvor: O'Leary D. E., 2000, str. 157)

O'Leary D. E. (2000, str. 156) ističe da manje organizacije najčešće odabiru pristup „veliki prasak“, a veće organizacije biraju fazni pristup implementacije. Razlog je tome što je „veliki prasak“ vrlo teško izvesti ako ga izводи veća organizacija. Također se osvrće na složenost organizacije kao drugom parametru. Složenost organizacije se može mjeriti prema kupcima i proizvodnim linijama. Obično organizacija sa više kupaca ili više proizvodnih linija je složenija. Doduše, ti isti parametri se mogu koristiti za utvrđivanje veličine organizacije.

Odnos je prikazan na slici 4.6.3.1. Slika 4.6.3.1 prikazuje da manje i jednostavnije organizacije će prvo uzeti „veliki prasak“ kao pristup implementacije, dok će veće i složenije prije razmatrati fazni pristup implementacije.

Organizacijska hijerarhija i kontrola

O'Leary D. E. (2000, str. 158) navodi da organizacija sa već razrađenom hijerarhijom i strogim kontrolama je bolje opremljena za provođenje faznog pristupa, a organizacija, koja nema još hijerarhiju i slabe kontrole, može najbolje podnijeti „veliki prasak“ kao pristup implementaciji. Takva organizacija jednostavno nije sposobna izdržati dužnosti koje sa sobom nosi fazni pristup implementacije.

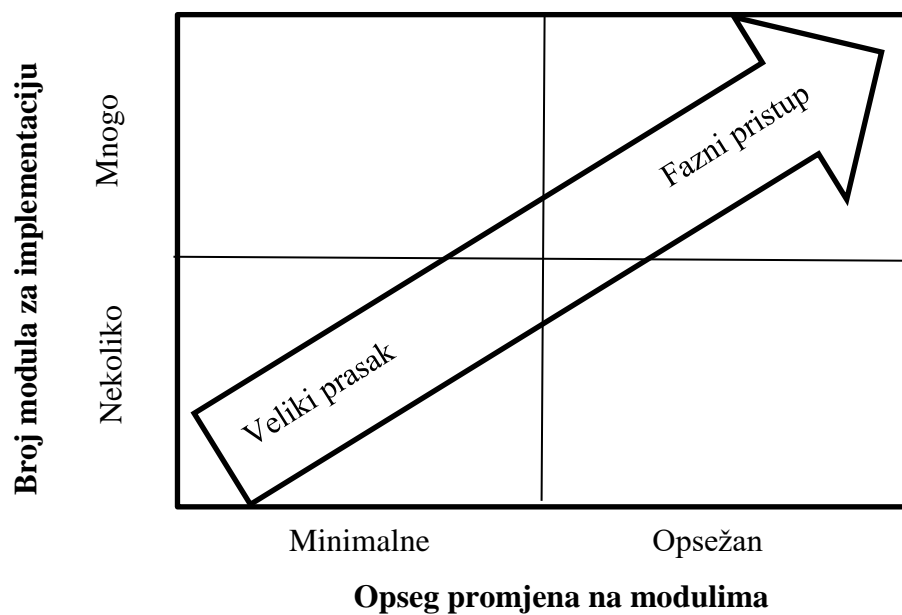


Slika 4.6.3.2 Odnos između organizacijske strukture i kontrole (Izvor: O'Leary D. E., 2000, str. 158)

Na slici 4.6.3.2 je prikazano da organizacije bez hijerarhijske strukture i slabim kontrolama su više podložne odabrati „veliki prasak“, a organizacije sa hijerarhijom i strožim kontrolama su više naklonjene faznom pristupu implementacije.

Opseg implementacije

O'Leary D. E. (2000, str. 159 - 160) da broj modula i količina modifikacija može utjecati na odluku o strategiji implementacije. Ukoliko poduzeće treba implementirati samo nekoliko modula veća je vjerojatnost da će odabrati pristup „veliki prasak“, a fazni pristup implementacije će biti bolji izbor ako se radi o više modula. Jer svaki novi modul traži više resursa kako bi bilo moguće implementirati. Više modula znači više resursa potrebnih da se ta implementacija provede. Drugi parametar, koji O'Leary D. E. spominje, je količina potrebnih modifikacija. Ukoliko je nije potrebno modificirati modul, tada je jasno da će poduzeće posegnuti za „velikim praskom“, dok se u slučaju više modifikacija na modulu odlučuje za fazni pristup implementacije. Više modifikacija označava da je potrebno razviti i testirati te novosti na modulima kako bi se osiguralo da ta promjena nije uzrok novih problema na modulu ili njegovoj interakciji sa drugim modulima.



**Slika 4.6.3.3 Odnos broja modula za implementaciju i opsega promjena na njima
(Izvor: O'Leary D. E., 2000, str. 159)**

Slika 4.6.3.3 prikazuje da nekoliko modula odabranih za implementaciju sa minimalnim modifikacijama uglavnom vodi ka odabiru „velikog praska“, a mnogo modula odabranih za implementaciju te opsežne promjene na istima vode ka odabiru faznog pristupa implementacije.

U nastavku slijede opisi pristupa implementacije koji su specifični za proizvođače ERP sustava.

4.6.4. ASAP (eng. *SAP's Accelerated SAP*)

Nagpal S., Khawtri S. K. i Kumar A. (2015) definiraju ASAP (eng Accelerated SAP) kao implementacijsku metodologiju koju zastupa SAP. Kažu da ona primjenjuje izvrnutu logiku analize i dizajna u implementaciji što znači da je dizajn praćen sa analizom. Ta metodologija se fokusira na kratke projekte te postoji nekoliko faza koje autori identificiraju, a to su:

1. faza: Priprema projekta (eng. *Project Preparation*)

Ova faza uključuje definiciju projekta, identifikaciju opsega i specifikaciju, iscertava se strategija implementacije, identifikacije vremenskog rasporeda projekta i slijeda implementacije, određivanje projektnog pravilnika za organizaciju i upravne odbore te na kraju i alokaciju resursa.

2. faza: Shematski plan poslovanja - Faza specifikacije zahtjeva (eng. *Business Blueprint - Requirement Specification Phase*)

Ova faza ovisi o količini podataka koji su predani tijekom sastanaka sa proizvođačem ERP-a. Jako je korisno da shematski plan sadrži poslovnu strategiju, organizacijsku strukturu, ulaze za glavne podatke i dokumentaciju poslovnih procesa.

3. faza: Konfiguracija sustava prema shematskom planu poslovanja (eng. *Configuration of the system according to the Business Blueprint*)

Ovo je procedura u 2 koraka koja se izvršava prem top-down pristupu. Prvo se definira općenita konfiguracija gdje se definiraju globalne postavke u organizaciji. Nakon toga se konfiguriraju poslovni procesi. Ova faza zahtjeva testiranje.

4. faza: Konačna priprema – testiranje sustava i korisničko testiranje (eng. *Final Preparation – System Testing & End-User Testing*)

U ovoj fazi se cilja na uklanjanje razlika te problema. Također se u ovoj fazi primjenjuje test izdržljivosti (eng. *stress testing*)

5. faza: Službena objava i faza podrške (eng. *Go Live and Support Phase*)

U ovoj fazi se isvodi instalacija i korisnici počinju koristiti sustav. U početku se pruža potrebna podrška, a s vremenom se sve manje pruža.

Sve te faze su podržane sa alatima koje nudi SAP. Svaka faza ima svoje alate i točno proceduru kako se određena faza izvodi. Tako da je SAP omogućio provjerenu metodologiju koja se može lako ponovno koristiti. Ova metoda je korisna jer omogućuje implementaciju ERP sustava u čim kraće vrijeme te zbog toga su manji troškovi.

4.6.5. OUM (eng. *Oracle Unified Method*)

Nagpal S., Khawtri S. K. i Kumar A. (2015) definiraju OUM (eng. *Oracle Unified Method*) kao metodologiju koju provodi Oracle. Ova metodologija je sljedbenik prethodne AIM (eng. *Oracle Applications Implementation Methodology*). Mnogo procesa iz AIM-a su preneseni u OUM. OUM se sastoji od pet faza umjesto šest kako je bilo u AIM-u. OUM je temeljna na principima zastupanih od strane vodećeg industrijskog standarda – UP (eng. *Unified Software Development Process*). OUM zapravo proširuje UP kako bi se obuhvatio cijeli opseg Oracle-ovih ERP projekata. Nagpal S., Khawtri S. K. i Kumar A. (2015) ističu 5 i objašnjavaju pet faza OUM-a:

1. Početak (eng. *Inception*) - Ciljevi ERP projekta su prilagođeni tako da su sve interesne strane usklađene. Definira se i opseg sa uključenim zahtjevima i rizicima.
2. Razrađivanje (eng. *Elaboration*) – Opseg se razradi na detaljne zahtjeve i razdvajanje rješenja. U ovoj fazi se može izraditi prototip kako bi bilo jasno svim stranama kako će izgledati i ovo pomaže kasnijoj izradi arhitekture.
3. Izrađivanje (eng. *Construction*) – Sada se koriste izrađeni modeli kako bi se konfigurirao sustav prema standardnoj funkcionalnosti koje nudi softver. Nakon čega slijedi razvoj korisničkih komponenata, testiranje i integracija. Sustav je spreman za prvu verziju i obično se smatra beta verzijom koja je dostupna za validaciju i ispitivanje prihvatljivosti.
4. Prijelaz (eng. *Transition*) – U ovoj fazi se sustav priprema za prihvatanje od strane klijentove organizacije. Izvršava se sustavno testiranje i validacija kako bi klijent prihvatio i krenuo koristiti novi sustav. Planiranje prelaska na novi sustav.
5. Produkcija (eng. *Production*) – U ovoj fazi sustav postaje u funkciji. Prati se te se pruža podrška za sve upite, podešavanje performansi i grešaka. Održava se procedura zahtjeva za promjenom kako bi se prijavili nedostaci i nove funkcionalnosti za nadogradnju za slijedeća izdanja.

Nagpal S., Khawtri S. K. i Kumar A. (2015) spominju da se UOM dijeli dalje na procese tako da tvore strukturu matrice gdje su faze redovi, a procesi stupci. Znači procesi se pišu ili iznad ili ispod, a faze lijevo ili desno. I autori navode sve procese: zahtjevi poslovanja, analiza zahtjeva, mapiranje i konfiguriranje, analiza, dizajn, implementacija, testiranje, upravljanje performansima, tehnička arhitektura, akvizicija podataka i konverzija, dokumentacija, upravljanje organizacijskim promjenama, edukacija, prijelaz, izrada i podrška.

Pozitivna strana ove metodologije je da uspješno razrađuje poslovanje na manje dijelove kojima je puno lakše upravljati i na taj način se lakše obavi proces implementacije.

4.6.6. Microsoft Dynamics' Sure Step

Nagpal S., Khawtri S. K. i Kumar A. (2015) obrađuju i ovu metodologiju te se njihova znanja koriste u nastavku ovog poglavlja. *Microsoft Dynamics Sure Step* metodologija koju provodi Microsoft. Uz metodologiju su još dostupni razni alati za upravljanje projektom, integrirane najbolje prakse koje potpomažu službeno pokretanje, migraciju, konfiguraciju i nadogradnju. Sastoji se od šest glavnih faza i dvije dodatne za optimizaciju i nadogradnju, a one su:

1. Dijagnoza (eng. Diagnostic) – Definiranje projektnog plana. Finaliziranje strategije projekta i opsega.
2. Analiza (eng. Analiza) – Razumijevanje procesa klijenta i ponuditelja ERP-a, njihovo modeliranje i dokumentiranje.
3. Dizajn (eng. Design) – Klijentski procesi se prilagođavaju procesima Dynamics-a. Ovdje se također može izraditi prototip ali on nije dio ove metodologije.
4. Razvoj (eng. Development) – Uključuje prilagodbu gdje se dodaju nove funkcionalnosti ili se mijenjaju postojeće. Također se radi migracija podataka i testiranje.
5. Instaliranje (eng. Deployment) – Instaliranje i konfiguriranje sustava kod korisnika. Uključuje i sustavno testiranje, ispitivanje opterećenja i ispitivanje prihvatljivosti.
6. Djelovanje (eng. Operation) – Sustav se službeno pokreće i koristi. Ovdje je uključena podrška za korisnika nakon službenog pokretanja.

Dodatne faze:

7. Optimizacija (eng. Optimization) – Optimiziranje rješenja pronalaženjem procesa koji se izvršavaju duže i njihovo fino podešavanje.

8. Nadogradnja (eng. Upgrade) – Nastavljanje iz postojećeg prema novim izdanjima. U ovoj fazi se mogu dodavati nove funkcionalnosti u rješenje.

~ ~ ~

MS Sure Step	<ul style="list-style-type: none"> Cultivate Customer Relationship Pre-Sales Support Execute Decision Accelerators Complete SOW 	<ul style="list-style-type: none"> Finalize Project Plan and Project Charter Execute Functional Requirements Workshop Execute Fit Gap Analysis Develop Test Plan 	<ul style="list-style-type: none"> Conduct Core Team Training Develop Functional Design Documents for Configurations (Fits) Develop FDD's for Customizations (Gaps) Demo CRP (For Rapid Project Type) 	<ul style="list-style-type: none"> Configure/ Setup solution Conduct Process Testing Conduct Integration Testing Manage Scope and Resolve Issues Complete Solution Design Document 	<ul style="list-style-type: none"> Conduct Train the Trainer (TTT) Training Conduct User Training Conduct User Acceptance Testing Perform Go-Live Readiness Activities Ready Solution for Production Deployment 	<ul style="list-style-type: none"> System Go-Live Post-Production Support
	Diagnostic	Analysis	Design	Development	Deployment	Operation
Oracle OUM	<ul style="list-style-type: none"> Define Scope Build Environments Confirm Resources Develop Work plan Develop Project Financial Plan 	<ul style="list-style-type: none"> Deliver End-to-End System Walkthrough Develop Business Process Model Develop Security Model Develop Integration Schema 	<ul style="list-style-type: none"> Execute Conversion Build Integrations Develop Test Scripts 	<ul style="list-style-type: none"> Deliver End User Training Execute Testing Build Production Environment Execute cutover 	<ul style="list-style-type: none"> Build Maintenance Contract 	
	Inception	Elaboration	Construction	Transition	Production	
SAP's ASAP	<ul style="list-style-type: none"> Initial Project Planning Project Procedures Training Kick Off Technical Requirements Quality Check 	<ul style="list-style-type: none"> Project Management Organizational Change Management Training Develop System Environment Organizational Structure Definition Business Process Analysis Business Process Definition Quality Check 	<ul style="list-style-type: none"> Project Management Organizational Change Management Training Baseline Configuration & Confirmation System Management Final Configuration & Confirmation Develop Programs, Interfaces etc Final Integration Test Quality Check 	<ul style="list-style-type: none"> Project Management Training System Management Detailed Project Planning Cutover Quality Check 	<ul style="list-style-type: none"> Migration to Production Environment Production Support Monitoring Performance Optimization 	
	Project Preparation	Business Blueprint	Realization	Final Preparation	Go Live & Support	

Slika 4.6.1. Usporedni pregled strategija implementacije specifičnih za proizvođače ERP sustava (Izvor: Nagpal S., Khawtri S. K. i Kumar A., 2015)

Slika 4.6.1. prikazuje usporedni pregled faza metodologija koje su bile spomenute prethodno. Ove metodologije su korisne jer pružaju točnu proceduru kako se napreduje kroz proces implementacije. S obzirom da su navedene metodologije specifične za proizvođače ERP sustava, to znači da svaka metodologija bi trebala biti primijenjena kod ERP sustava od vlasnika metodologije.

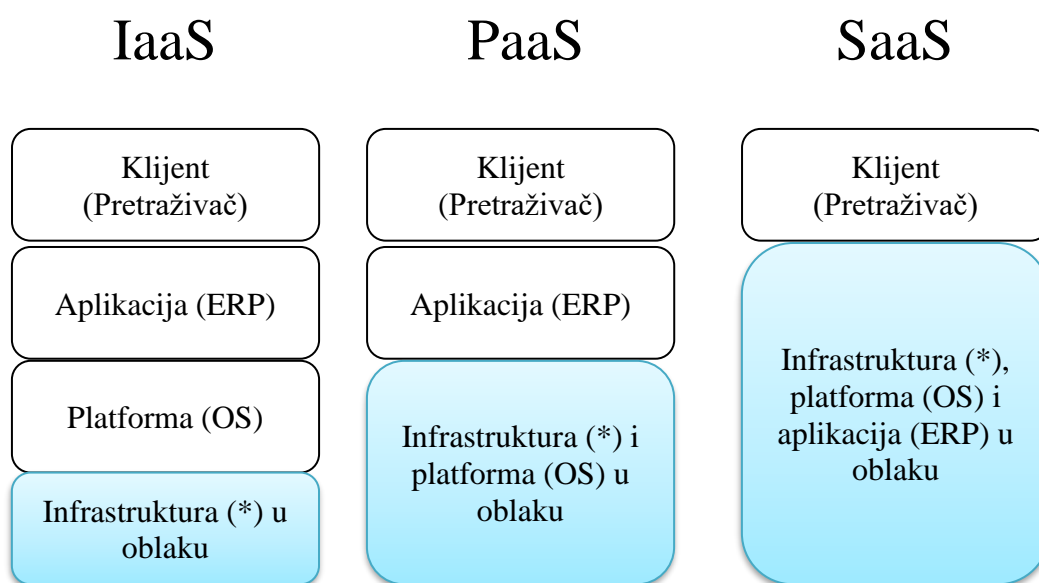
4.7. Trendovi u poslovanju

ERP se od samog početka konstantno razvija, mijenja i unaprjeđuje. Kako se razvija tehnologija i često se predstavljaju nove tehnologije koje na neki način postaju prekretnica u načinu na koji se poslovalo. Kada se pojavi tehnologija koja donosi povlastice i može potencijalno poboljšati poslovanje, poduzeća teže tome da tu tehnologiju i koriste u svome poslovanju. Pa tako će se u nastavku objasniti trendovi ERP-a.

4.7.1. Cloud ERP

Računalstvo u oblaku ili *cloud* je popularna tehnologija te je značajna za četvrtu industrijsku revoluciju. Specifično za ovu tehnologiju je da troškove održavanja infrastrukture i/ili aplikacije prebacuje na vlasnika prostora na oblaku. Poznato je do sada da su ERP sustavi skupa investicija i zahtjevna. S obzirom na to, računalstvo u oblaku se čini kao odlično rješenje da se rasterete organizacije nekih troškova. Slijedi kratak prikaz ovog trenda u ERP sustavima.

Raihana G. F. H. i College J. M. (2012) spominju da se Cloud ERP javlja kao rješenje za te troškove. Raihana G. F. H. i College J. M. (2012) navode da se Cloud ERP lako instalira i njegovo upravljanje u potpunosti ovisi o podršci proizvođača ERP-a. Što znači da organizacija ne mora brinuti o infrastrukturi i ostalim ovisnim troškovima. Proizvođač ERP-a može osigurati kontrolu umjesto poduzeća te takav ERP sustav zahtjeva internetsku vezu. Odnos opsega obaveza proizvođača i organizacije ovisi o tome koji tip Cloud ERP-a se implementira.



* Virtualiziran hardver

Slika 4.7.1.1 Prilagodba slike o servisima računalstva u oblaku ERP sustavima.

(Izvor: Al-Ghofaili, A. A., Al-Mashari, M. A., 2014, str 44.)

Al-Ghofaili A. A. i Al-Mashari M. A. (2014, str. 44) navode da postoji više vrsta Cloud ERP-a koje su ilustrirane na slici 4.7.1.1. Što znači da tipovi Cloud ERP-a su ekvivalentni servisima računalstva u oblaku. Uzevši to u obzir tipovo Cloud ERP-a se imenuju: IaaS ERP, PaaS ERP, SaaS ERP (još poznat pod nazivom EaaS – eng. *ERP as a Service*). Lako se može povući paralela sa objašnjenjima servisa računalstva u oblaku

U tablici 4.7.1.1 se nalazi prikaz povlastica i izazova pojedinačno za svaki tip Cloud ERP-a koje su spomenuli Al-Ghofaili A. A. i Al-Mashari M. A. (2014, str. 46).

Tip Cloud ERP-a	Povlastica	Izazov
<i>IaaS ERP</i>	Maksimizira fleksibilnost kako bi se zamijenili pružatelji <i>cloud</i> -a ili se premjestilo u zgradu poduzeća	Online servis sa manjom cijenom
<i>PaaS ERP</i>	Mješavina fleksibilnosti i uštede	Izazovi koordinacije – proizvođač ERP-a upravlja aplikacijom dok pružatelj <i>cloud</i> -a upravlja infrastrukturom.
<i>SaaS ERP</i>	Maksimizira efikasnost za univerzalne aplikacije	Proizvođač ERP-a zaključava servis. Korisnik nema mogućnost premještanja aplikacije na drugog pružatelja <i>cloud</i> -a

Tablica 4.7.1.1. Prikaz povlastica i izazova svih tipova Cloud ERP-a (Izvor: Al-Ghofaili, A. A., Al-Mashari, M. A., 2014, str 46.)

Cloud ERP, u odnosu na tradicionalni ERP, oslobađa firmu dijela troškova. Kako se u oblaku mogu držati i infrastruktura i aplikacija to uvelike olakšava poduzeću implementaciju ERP-a jer ne moraju ni imati infrastrukturu ni ne moraju provoditi instalaciju aplikacije u poduzeću već će ona biti dostupna online. Nedostatak je da ukoliko se uzme „čisti“ Cloud ERP (EaaS) onda je korisnik vezan za pružatelja *cloud*-a te je ERP dostupan preko interneta. Nedostaci ovog tipa ERP-a su ovisnost o internetu i visoki trošak najma ERP-a ukoliko se radi o dužem periodu. Možda se čini da je tradicionalni ERP skuplji, ali je EaaS puno skuplji na duži vremenski period.

4.7.2. Primjeri korištenja IoT-a u ERP-u

IoT postaje vrlo popularan pa tako postaje sve više korišten u poslovanju. Tako postoje neki primjeri korištenja IoT-a u ERP sustava koje ću navesti u nastavku.

Medicina: Aerocrine

Aerocrine je švedsko poduzeće koje proizvodi uređaje za dijagnozu i liječenje astme. Oni imaju dva uređaja koji mjere količinu izdahnutog dušikovog oksida, koji je važan indikator za upalu dišnih puteva. Ne da se ovako može detektirati je li astma uzrok pacijentovih simptoma nego se može i odrediti vjerojatnost steroidne reakcije. Oni koriste *Microsoft Azure Cloud* za sakupljanje podataka koje očitavaju njihovi uređaji i te podatke obrađuju koristeći *Azure Stream Analytics*. Oni podatke pokazuju prodaji i predstavnicima klijenata preko *Microsoft PowerBI*-a za *Office 365* radne ploče. Oni također koriste te podatke u svojem *Microsoft Dynamics NAV* ERP-u kako bi prodajnom timu omogućili jasnu sliku o uređajima. Koriste te podatke kako bi bolje upravljali sa lancem zaliha. Na ovaj način oni će moći isporučiti novi uređaj i prije nego ih klijent zatraži to praćenjem broja mjerenja. (Izvor: Olavsurd T. , 2017, Video: <https://youtu.be/elwDgxBcFI0>)

Šumarstvo: Peterson

Peterson je poduzeće koje se bavi šumarstvom od 1981. godine. Kako tehnologija napreduje tako su i oni osjetili potrebu za nabavom boljih strojeva, a naposljetku i integracijom IoT-a u svoje poslovanje. Oni koriste IoT za praćenje strojeva te im je glavni cilj da koristeći podatke mogu predvidjeti kada će koji dio stroja postati neupotrebljiv te poslati novi i prije nego se stari pokvari. Oni koriste *Azure IoT* i nedugo su implementirali *Dynamics 365* koji je još poznat kao web verzija NAV-a. (Izvor: <https://customers.microsoft.com/en-us/story/peterson-dynamics365-azure-cloud-iot-united-states-forestry-fishing-mazik> , pristupano 17.9.2018. Video: https://www.youtube.com/watch?v=yeCOyg_3Dg8)

Proizvodnja Pića: Coca-Cola

Coca-Cola je vodeći proizvođač i distributer pića. Coca-Cola je inkorporirala SAP sustav. Također je krenula koristiti IoT i ostale digitalne tehnologije u svrhu poboljšavanja iskustva korisnika, vjernosti i marketinga. Coca-Cola je koristila IoT od 1982. kada je imala prve aparate sa sokovima povezane na Internet. Coca-Cola je nudila usluge sa jednim klikom kako bi poboljšala ciljani marketing i uočili vršne slučajeve konzumiranja. Podaci iz aparata za piće su korisni i na taj način je Coca-Cola otkrila da su pića višnja Coca-Cole i vanilija Coca-cole bila popularne među korisnicima. Također povezani aparati omogućuju identifikaciju profitabilnosti aparata, stanje zaliha aparata te lokacije najveće frekventnosti. Coca-Cola aparati, koji su povezani Internet, omogućuju bezgotovinsko plaćanje i na taj način poboljšavaju vjernost nuđenjem bodova za svaku kupnju i iskorištavanje poklona korištenjem pametnog telefona.

Bitno je napomenuti da njihovi aparati također prihvaćaju mobilne novčanike (*eng. mobile wallets*) što je također korišteno za promicanje vjernosti. Slijedeći plan Coca-Cole je omogućiti kupnju sa studentskim identitetom i praćenje povezanih hladnjaka u svrhu praćenja zaliha i optimizacije održavanja hladnjaka. (Izvor: Pyatakova, S., 2018. Video: <https://www.youtube.com/watch?v=Vne4Y6UrUy0>)

4.8. Važnost ERP-a u poslovanju

Bez ERP sustava, u organizaciji se može javiti potreba za raznim aplikacijama u pojedinim dijelovima poduzeća. Nakon nekog vremena, poduzeće se sastoji od više različitih aplikacija koje služe svojoj svrsi, a ne koriste mnogo poduzeću. Naprotiv, čak i odmažu u nekim aspektima koji ne moraju odmah biti očiti. To utječe negativno na optimizirane funkcije . Abdullah i Ambedkar (2018) navode teškoće s kojima se poduzeće može susresti ako nema ERP sustav, a te teškoće su:

- Mnogo različitih softvera za različite procese.
- Bez lakog pristupa informacijama o poslovanju.
- Računovodstvo traje duže i teže je.
- Prodaja i korisničko iskustvo pate.
- IT je previše kompleksan i troši mnogo vremena.

Zbog tih teškoća, poduzeća uviđaju da je ERP sustav dobar za njih. ERP sustav sadrži centraliziranu bazu podataka koja sadrži podatke iz svih aspekata poduzeća. Ti podaci su uneseni samo jednom, ne događa se da se jedan podatak unese više puta. ERP sustav sa tom bazom može upravljati svim aspektima poduzeća kroz ERP sustav. ERP sustav omogućuje optimiziranje poslovnih procesa i procesa donošenja odluka. Pokazuje koji poslovni procesi su nepotrebni, koje bi trebalo mijenjati. Isto tako ERP je transparentan, što označava da svatko može vidjeti stanje kroz ERP sustav.

5. Analiza procesa integracije Internet stvari-a u ERP

IoT postaje vrlo popularna tehnologija i mnogi vide potencijal u korištenju iste. U poslovnom svijetu teži se jedinstvenosti i konzistentnosti pa se u poslovnom svijetu teži ka integriranju te nove tehnologije u svoj informacijski sustav. U ovom radu je priča usmjerena na ERP sustav pa će se u ovom poglavlju objasniti integracija općenito te važnost integracije IoT-a u poslovanje.

5.1. Važnost integracije Internet stvari-a u ERP u poslovanju

Haller, S., Karnouskos, S. i Schroth, C. (2008) spominju da postoje dva primjera iz kojih se mogu uočiti poslovne vrijednosti IoT-a. U nastavku će se prenijeti njihova vizija tih dva primjera.

Vidljivost stvarnog svijeta (*eng. Real-World Visibility*)

Poduzeća su bila svjesna vanjski promjena do sada, ali to se radilo ručno tako da se istraživalo, prikupljalo informacija putem anketa ili upitnika. Pomoću IoT-a se može sada puno brže i automatski reagirati na promjene vanjskih faktora odnosno promjene u vanjskoj okolini koja je detektirana putem IoT senzora. IoT senzori su sada „osjetila“ organizacije kojima ona može „vidjeti“ i „osjetiti“ pomoću senzora. Svaki „osjet“ se šalje u informacijski sustav u poduzeću koji može odraditi prikladne akcije ovisno o primljenom podatku. Kao primjeri za vidljivost navode se često praćenje stanja u dućanima, pa onda primjer gdje se putem IoT-a prati napredak u završavanju narudžbe pa onda kada je završena se javi ERP-u da je narudžba spremna. Vidljivost omogućuje poduzeću da pomoću praćenja ima bolju kontrolu i upravljanje nad poslovnim procesima te poboljšava kavalitetu poslovnih podataka.

Dekompozicija poslovnih procesa (*eng. Business Process Decomposition*)

Kako IoT izvršava svoje zadatke neovisno o glavnom sustavu, događa se dekompozicija poslovnih procesa. Tako da oni procesi koje izvode IoT uređaji se mogu izdvojiti iz prethodnog slijednog procesong niza. Tako dolazimo do distribuiranih sustava čija je prednost da uklanjaju nepotrebno slanje poruka. To se osobito događa ako IoT uređaj ima sposobnost obrađivati podatke i donjeti odluku – takav IoT uređaj se naziva pametni predmet (*eng. smart item*). Pametni predmeti odvajaju i poslovnu logiku i ona postaje distribuirana i razdijeljena na manje dijelove što ide u korist skalabilnosti. Kako ti pametni predmeti mogu spremati podatke u neki privremeno spremište, oni su mrežno neovisni te će raditi ako i nestane veze prema glavnom sustavu. Smatra se da ovakva dekompozicija sustava je evolucijski nastavak E-SOA (*Enterprise*

Service-Oriented Architecture). Haller, S., Karnouskos, S. i Schroth, C. (2008) definiraju E-SOA kao skup arhitekturnih komponenata koje izgrađuju autonomne ali interoperabilne strukture.

Da se sumira sve na jedno mjesto, slijedi lista pozitivnih učinaka IoT-a na poslovanje:

- bolja kontrola i upravljanje nad poslovnim procesima
- poboljšana kvaliteta poslovnih podataka
- bolja responzivnost cijelog sustava jer su eliminirane nepotrebne poruke
- skalabilnost – poslovna logika je distribuirana
- neovisnost o mreži – privremeni spremnik omogućuje da pametni predmet radi iako nema veze prema glavnom sustavu

IoT čini poduzeća izuzetno brzo reagiraju na promjene koje detektiraju pomoću senzora, također je korisna stvar kada se cijeli monolitni sustav počinje izvoditi djelomično ili potpuno distribuirano to bi značilo da se može jedna instance distribuiranog procesa promijeniti u izvođenju ali ne i u ulaznim i izlaznim podacima, tada cijeli sustav to neće ni osjetiti.

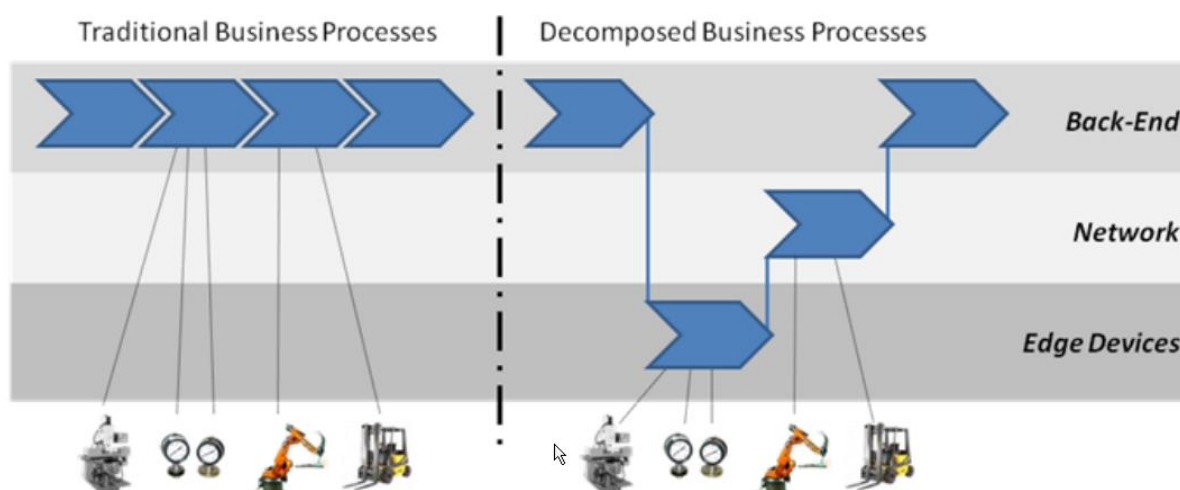
5.2. Analiza znanstvenih izvora

Početna točka u svakom prihvaćanju tehnologije, potrebno je shvatiti ima li uopće smisla truditi se oko nove tehnologije. Portal ERP News (2018) objašnjava da bi poduzeće trebalo prvo utvrditi ima li zahtjevane uvjete za rad s IoT tehnologijom. Kako hvatanje u koštac sa novim tehnologijama odnosno Industrijom 4.0 znači hvatanje sa tehnologijom koja se mijenja rapidno te ukoliko ERP sustav nije sposoban pratiti te promjene i one su skupa investicija za poduzeće, tada ne bi trebalo trčati za tom tehnologijom. Nakon toga je potrebno utvrditi način komunikacije u sustavu. Ako se radi sa novom tehnologijom kao što je IoT, potrebno je osigurati da su informacije dostupne kroz cijelo poduzećeu realnom vremenu. Dodatno pitanje, koje postavlja Kowalke P. (2018), je ima li ERP već integrirane funkcionalnosti za korištenje IoT-a bez da se radi sa eksternim platformama. Tada nema potrebe za korištenjem *cloud* tehnologija u svrhu dohvaćanja podataka. Treba se zapitati ima li ERP sustav podršku za povezivanje sa *cloud* alatima tako da ako ima onda postoji potencijal za korištenje IoT-a. I zadnja stavka koja je ključna je pitanje je ima li poduzeće fizičku opremu spremnu za takvu integraciju, nema smisla se truditi oko IoT-a dok u poduzeću nema ni jedan IoT senzor ili uređaj. Pravi potencijal IoT-a proizlazi iz komunikacije između IoT stvari i ERP sustava.

Iz navedenog odlomka se mogu lako izvaditi pitanja važna za donošenje odluke treba li integrirati IoT:

1. Je li poduzeće spremno na brze i česte promjene koje sobom donosi Industrija 4.0?
2. Jesu li podaci dostupni svima u poduzeću kroz ERP i u realnom vremenu?
3. Ima li ERP integrirane IoT mogućnosti ili mogućnost povezivanja sa *cloud* tehnologijama?
4. Koristi li poduzeće fizičke jedinice sa IoT sposobnostima ili je to buduća namjera?

Pretpostavi li se da je poduzeće spremno za integraciju ERP-a, potrebno je znati na koji način će se te IoT stvari dokumentirati putem BPM alata. Meyer, S., Ruppen, A. i Magerkurth, C. (2013) navode da bi se IoT stvari trebale odvojiti od ostatka dijagrama u zasebnu liniju jer IoT stvari radi neovisno od ostatka sustava i samo šalje poruke. Na slici 5.2.1 je prikazana slika koja ilustrira razliku između sustava sa i bez IoT stvari.



Slika 5.2.1 Razlika između tradicionalnog procesa i dekomponiranog procesa sa IoT stvarima (Izvor: Haller, S., Karnouskos, S. i Schroth, C., 2008)

Slika 5.2.1 se lako može pretočiti u UML prikaz tako da IoT stvari imaju zasebnu liniju u procesu.

Sada bi slijedio proces integracije. Pretraživanje baza podataka nije dalo neku konkretnu metodologiju, ali se naišlo na članke prema kojima je zaključeno da zapravo ni ne postoji neka konkretna metodologija trenutno. Kowalke P. (2018) navodi da bi poduzeća trebala gledati da čim lakše prođu proces integracije i da koriste pomoć *cloud* tehnologije kako bi se povezale IoT stvari sa ERP sustavom. Dok Carlton R. (2016) spominje 3 načina na koje se može integrirati IoT u ERP sustav:

1. Proširenje aktivnosti pogonjene ERP-om koristeći RFID ili NFC uređajima.
2. Pобољшanje upravljanja poduzetničkim prihodima iskorištavanjem matrice ERP / IoT uređaja - svaki trošak ili prihod se šalje u modul financija u realnom vremenu
3. Proširiti ERP-ove nadzorne operacije koristeći IoT kontrolne sustave – praćenje i obavještanje u realnom vremenu (obavijesti, greške)

Sada slijedi prijedlog procesa integracije kako nije bilo konkretnih članaka na internetu.

5.3. Prijedlog procesa integracije

<i>Faze</i>	<i>Koraci</i>
<i>Otkrivanje</i>	<ul style="list-style-type: none"> a. identificiranje IoT stvari b. definiranje problema koji se rješava c. definiranje skupa podataka d. proširivanje skupa podataka svojstvima e. modeliranje IoT procesa u BPM-u f. definiranje korelacija među IoT stvarima g. definiranje vizualizacija korelacija IoT stvari h. definiranje svih slučajeva korištenja IoT stvari
<i>Implementacija</i>	<ul style="list-style-type: none"> a. programiranje IoT stvari b. korištenje podataka IoT stvari c. testiranje IoT stvari d. instaliranje IoT stvari
<i>Korištenje</i>	<ul style="list-style-type: none"> a. kontrola IoT stvari b. poboljšanje IoT stvari c. otkazivanje IoT stvari

Tablica 5.3.1 Prijedlog faza i koraka integracije IoT-a u ERP (rad autora)

Kako nije pronađena ni jedna metodologija sa koracima i fazama, navesti ću sada svoj prijedlog. Prije nego se krene na faze, važno je napomenuti da integracija IoT se može izvršiti generalno na dva načina: preko *cloud* tehnologija ili ako ERP podržava IoT stvari. Čitanjem

članaka došao sam do prijedloga faza koje bi se mogle koristiti kao pomoć do kraja procesa integracije.

Važno je napomenuti da fazama prethodi donošenje odluke o integraciji IoT stvari. Načine donošenja odluka su opisali autori članaka u prethodnom poglavlju. Integracija bi se mogla definirati u 3 faze i svaka faza ima svoje korake koji su prikazani u tablici 5.3.1. Uvidom u redoslijed može se i reći da su te faze ujedno životni ciklus jedne IoT stvari te je vrlo važno napomenuti da se navedeni koraci moraju izvršiti onoliko puta koliko ima IoT stvari ako je to prikladno.

U nastavku će se objasniti što se smatra pod kojom fazom i korakom. Kako bi bilo jasno koja je svrha pojedine faze i koraka.

1. faza - Otkrivanje

U prvoj fazi poduzeće identificira odnosno otkriva potencijal IoT stvari. U ovoj fazi se izvršavaju sve potrebne radnje u svrhu pripreme prije implementacije. Koraci u ovoj fazi su:

- identificiranje IoT stvari – Radi se popis svih potencijalnih IoT stvari u poduzeću ili koje se žele nabaviti.
- definiranje problema koji se rješava – Stvaranje popisa problema koje rješavaju IoT stvari grupirano prema pojedinačnoj IoT stvari.
- definiranje skupa podataka – Definira se koje podatke bi IoT stvar trebala slati i stvara se entitet za bazu podataka gdje će se podaci spremati.
- proširivanje skupa podataka svojstvima – Dodaju se granični slučajevi i podaci koji pokreću određene akcije. Ova svojstva ne moraju nužno biti u tablici ali mogu se spremati.
- modeliranje IoT procesa u BPM-u – Svaka promjena u strukturi organizacije potrebno je dokumentirati pa tako i sve IoT stvari koje su dodane u poduzeće.
- definiranje korelacija među IoT stvarima – Svrha IoT stvari je prikazivati neke zaključke. Potrebno je definirati odnose odnosno korelacije među IoT stvarima i njihovim podacima kako bi se točno znalo što sve bi se trebalo prikazati u izveštajima odnosno pogledu u ERP sustavu.
- definiranje vizualizacija korelacija IoT stvari -Definiranje izgleda radnih površina. Definiranje tipa prikaza izvještaja koji može biti u stvarnom vremenu, statistički i prediktivni. Statistički je izvještaj koji sadrži trenutne podatke i ne mijenja se u realnom

vremenu. Prediktivni izvještaj je izvještaj koji radi predviđanja za buduća razdoblja. Također se definiraju tipovi vizualizacije za određene podatke i korelacije podataka.

- definiranje svih slučaja korištenja IoT stvari – Uvođenje IoT stvari u poslovne procese dodaje distribuirane procese koji rade neovisno od tradicionalnih procesa. Potrebno je pokriti sve slučajeve korištenja kako bi se znalo kako sustav mora reagirati u kojoj situaciji te kakve će posljedice te reakcije ostaviti na cijeli sustav.

Sada slijedi prijelazak na implementaciju IoT stvari.

2. faza - Implementacija

Druga faza je faza implementacije i svi koraci ove faze su usredotočeni na postavljanje IoT stvari u funkciju. Koraci u ovoj fazi su:

- programiranje IoT stvari – Implementacija zadaće koja je zamišljena za tu IoT stvar uz uzimanje u obzir poslovne logike i vođenje brige o sigurnosti.
- korištenje podataka IoT stvari – Postavljanje *cloud* okoline ili definiranje ERP definirane logike za spremanje i konverziju podataka. Ovdje se postavlja način spremanja podataka koji dolaze iz IoT stvari.
- testiranje IoT stvari – Izvršavanje testova kako bi se ustanovilo radi li IoT stvar kako treba i radi li točno ono što treba raditi.
- instaliranje IoT stvari – Posljednje pripreme prije postavljanja IoT stvari na predviđeno mjesto koje uključuju brisanje testnih podataka. U ovoj fazi, IoT stvar se sprema na postavljanje na predviđeno mjesto i puštanje u pogon.

Nakon što se postavila IoT stvar, kreće se u posljednju fazu.

3. faza - Korištenje

Treća faza se odnosi na razdoblje nakon implementacije, tj. na razdoblje rada IoT stvari. U toj fazi se samo poboljšava i ažurira stanje ukoliko je to potrebno. Koraci u ovoj fazi su:

- kontrola IoT stvari – Nadzor u svrhu praćenja pojavljuju li se neke nepredviđene situacije.
- poboljšanje IoT stvari – Poboljšavanje performansa IoT stvari i ažuriranje sa inovativnim načinima ukoliko se procijeni da su ažuriranja potrebna.
- otkazivanje IoT stvari – IoT stvar više ne služi svrsi te se polako razmatra zamjena ili uklanjanje iste.

6. Praktični primjer – Integracija IoT modula na Azure i ERP (Microsoft Dynamics NAV)

Slijedi praktični primjer pomoću koji će prikazivati *Proof of Concept (PoC)* implementaciju IoT modula sa Microsoft Dynamics NAV sustavom.

6.1. Opis i konceptualizacija

Praktični primjer se sastoji od 2 dijela: izradu modula te slanje podataka na *cloud*, izrada izvještaja i prikaz tog izvještaja u Microsoft Dynamics NAV-u.

Zadatak praktičnog primjera je pokazati da je moguće prikazati podatke iz Internet stvari koji se dohvaćaju u realnom vremenu i šalju u NAV.

Ovaj zadatak se može riješiti na više primjera sa različitim arhitekturama. Odabrana arhitektura je da jedan ESP modul, koji sadrži senzore za mjerenje temperature, vlage i intenziteta svjetlosti, šalje u određenom vremenskom rasponu trenutna stanja na *cloud*.

6.2. Korišteni alati

Navedeni alati su korišteni samo za realizaciju praktičnog primjera. Također ima dodatnih procedura koje su se odradile u svrhu uklanjanja grešaka i pripremu okoline za realizaciju praktičnog primjera. Alati koji su se koristili u realizaciji rješenja jesu:

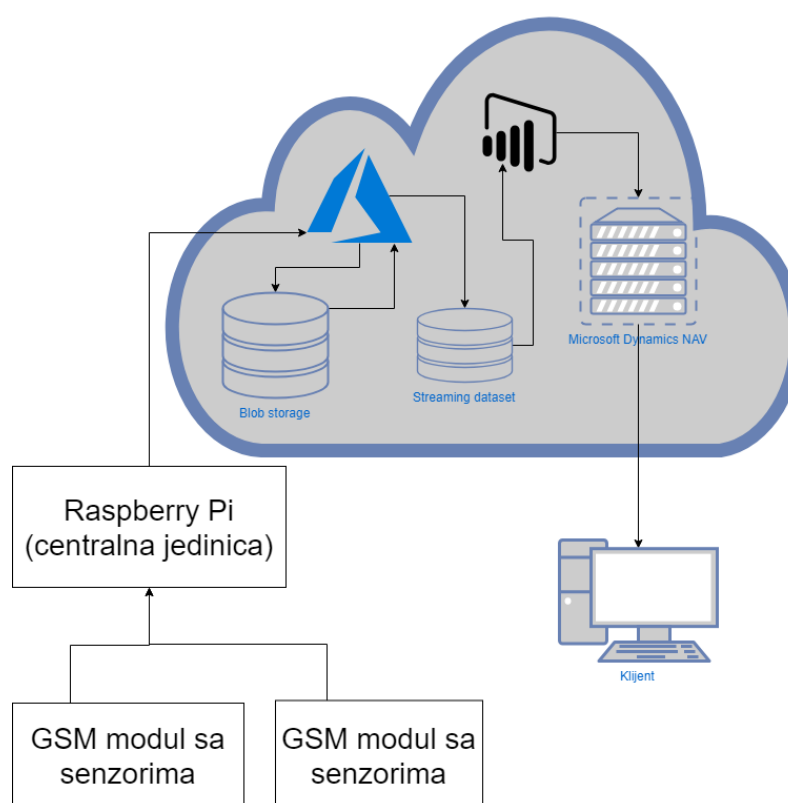
- Arduino IDE – Korišten je za programiranje ponašanja modula.
- Azure – Azure kao *cloud* platformu koja omogućuje rad s ostalim alatima
- Azure IoT Hub – Služio je za primanje podataka od Internet stvari i prosljeđivanje tih podataka za izradu izvještaja.
- Azure Stream Analytics Job – Ovaj alat se koristio za pretvorbu neobrađenih podataka u tablicu i druga uloga mu je omogućivati kontinuiran protok podataka u svrhu prikaza podataka u realnom vremenu.
- Microsoft Power BI – Služi za vizualizaciju podataka koji se primaju od Internet stvari. (Potrebno je kreirati račun prije početka izrade)
- Microsoft Dynamics NAV 2017 – Ovo je sustav u koji se integrira IoT funkcionalnosti

- Microsoft Dynamics Development IDE – Alat pomoću kojega se rade promjene u samom NAV-u. U radu se koristio zbog dodavanja akcije koja prikazuje izvještaj.
- Fritzing – alat za vizualizaciju IoT stvari.

6.3. Mogućnosti realizacije

Postoji nekoliko mogućnosti realizacije koje su se razmatrale u fazi planiranja diplomskog rada. U nastavku će se prikazati sve mogućnosti realizacije te će se jedna i odabrati.

GSM moduli i centralna jedinica šalju podatke



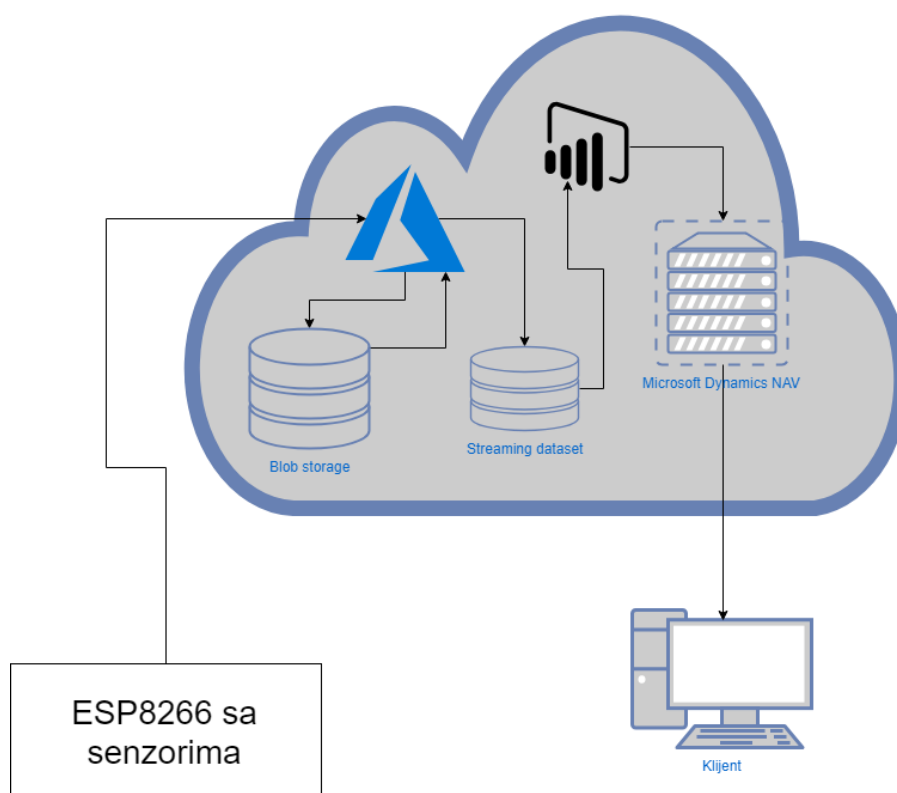
Slika 6.3.1. Prikaz arhitekture sa modulima i centralnom jedinicom (rad autora)

Na slici 6.3.1 je prikazana inicijalna ideja o arhitekturi praktičnog primjera. Zamisao je bila da se IoT dio izvede centralizirano na način da bi centralna jedinica bila Raspberry Pi koji bi bio zaslužan za skupljanje podataka i slanje istih prema serveru. Osim centralne jedinice, tu bi bili i sporedni moduli koji bi se sastojali od GSM modula i senzora te bi bili napajani baterijom. Ti sporedni moduli bi periodički slali očitavanja prema centralnoj jedinici koja bi te podatke agregirala i slala u *cloud*.

Prednost ove realizacije je da su sporedni moduli neovisni o bežičnoj mreži te se mogu smjestiti gdje god treba, zbog centraliziranosti se na *cloud* šalju agregirani podaci što je važnije nego da svaki sporedni modul šalje svoja očitavanja direktno na *cloud*.

Nedostaci ove realizacije je ovisnost o centralnom modulu, provedba dodatnih operacija uparivanja centralnog modula sa sporednim modulima, sporedni moduli su napajani baterijom što može uzrokovati nedostupnost sporednog modula kada se baterija potroši.

ESP8266 modul sa senzorima šalje podatke



Slika 6.3.2 Prikazuje arhitekturu kada ne postoji centralizirani pristup kod IoT stvari (rad autora)

Slika 6.3.2 prikazuje arhitekturu kada nema centraliziranog pristupa kod IoT modula. Ova realizacija pogodna za *Proof of Concept (PoC)* implementaciju da se demonstrira praktični primjer. ESP8266 modul se može povezati sa bežičnom mrežom pa onda taj modul može poslati periodička očitavanja na *cloud*.

Prednosti ove realizacije je neovisnost o jednom modulu (ako pretpostavimo da se koristi više instance modula prikazanog na slici), ovaj modul se napaja sa električnom energijom.

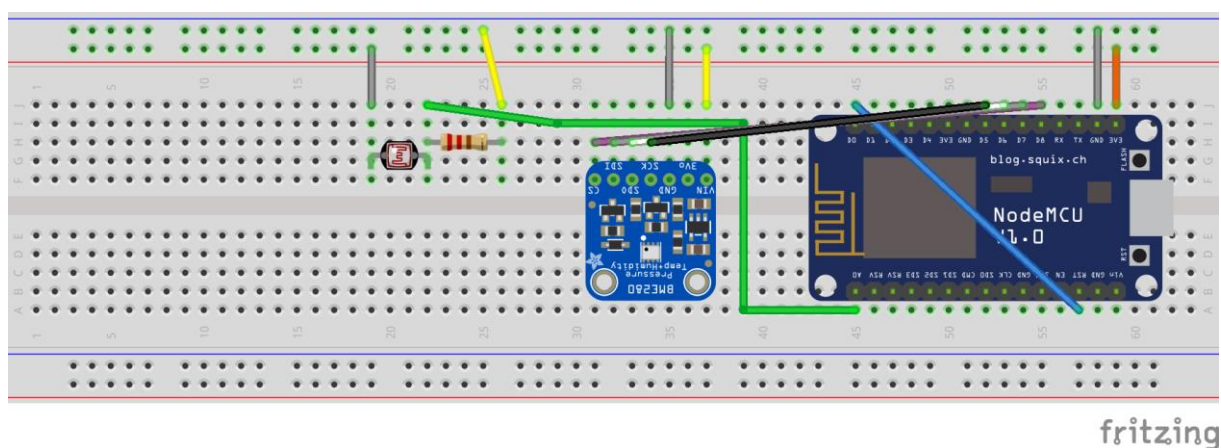
Nedostaci ove realizacije su da instance modula moraju biti u dosegu bežične mreže i izvora električne struje, što onemogućava slobodno postavljanje modula.

Zajednički dio – cloud

Podaci dolaze u *cloud* i ulaze u Azure-ov *IoT hub*. *IoT hub* uz pomoć *Azure Stream Analytics Job*-a pretvara podatke iz spremišta podataka u spremište tablica kako bi se isti podaci mogli koristiti u alatu za izvještavanje *Power BI*. *Microsoft Dynamics NAV* koristi ugrađeni izvještaj *Power BI*-a u svrhu prikaza vizualizacije podataka.

Postoji još jedan *Azure Stream Analytics Job* koji se koristi za kontinuirano slanje podataka prema *Power BI*-u u njegov *Streaming dataset* iz kojeg se onda čitaju podaci za radnu površinu sa stanjem u realnom vremenu, izvještaje i izvještaje u realnom vremenu (potrebno je kupiti *Power BI PRO*).

6.4. Scheme i opisi izvedbe IoT modula



Slika 6.4.1 Prikaz implementiranog modula (rad autora)

Slika 6.4.1 prikazuje modul koji je implementirani u radu i on se sastoji od (slijeva na desno) analognog senzora za intenzitet svjetla, otpornika, senzora za temperaturu, vlagu i tlak te ESP8266 modula. Slika prikazuje i način povezivanja da taj modul radi.

6.5. Spremanje podataka

Učitava se temperatura, vlaga, broj jačine svjetla praćen sa tekstualnom reprezentacijom intenziteta svjetla (*dim*, *light*, *bright*). To su podaci koji se inicijalno čitaju. U nastavku slijedi JSON čija svojstva su zapravo stupci u *Streaming dataset*-u.

```
[
  {
    "t": "2018-09-17T03:16:19.889Z",
    "deviceid": "AAAAA555555",
    "messageid": "AAAAA555555",
    "temperature": 98.6,
    "humidity": 98.6,
    "light": 98.6,
    "lightstring": "AAAAA555555"
  }
]
```

Slika 6.5.1 prikazuje izgled JSON-a čija svojstva su stupci u tablici (rad autora)

Na slici 6.5.1 se vidi izgled JSON-a koji ujedno predstavlja i stupce u *Streaming dataset*-u i taj se koristi za generiranje izvještaja u *Power BI*. Nakon očitavanja se dodaje „t“ što je podatak koji sadrži konkretno vrijeme kada je poruka poslana kako bi se mogli izrađivati izvještaji prema vremenu. Svojstvo „deviceid“ je jedinstveni identifikacijski tekst potreban za identifikaciju Internet stvari koja šalje poruke i koristi se za spremanje poruka u spremište podataka u *IoT Hub*-u. Svojstvo „messageid“ jedinstven identifikacijski tekst za poruku a sastoji se od elemenata trenutnog datuma i vremena koristeći sekunde kao najmanju jedinicu i na taj način se osigurava da je to svojstvo jedinstveno u svakom trenutku. To se može reći tako jer modul šalje podatke periodički tako da uvijek barem sekunda prođe prije nego se pošalju novi podaci.

6.6. Priprema Azure-a

U ovom koraku je potrebno imati Azure račun, ako se nema potrebno je instalirati ili besplatnu verziju ili odabrati neki plan plaćanja. Bitno je da se na Azure spaja sa poslovnim e-mailom kako *Power BI* ne dozvoljava osobne e-maile, a potreban je kasnije u primjeru.

Prvo je potrebno kreirati *IoT Hub*. Nakon kreacije *IoT Hub*-a trebali bi postojati sljedeće: *storage container*, *routes* i *endpoints*. Nakon toga je potrebno kreirati *Azure Stream Analytics Job* prema uputama te definirati ulaze i izlaze. Dobro je napomenuti da se kod kreiranja outputa i *Partition key* i *Row key* moraju biti u JSON-u koji šalje IoT modul. Pomoću *Stream Analytics Job*-a se podaci iz JSON-a spremaju u tablicu na Azure-u. Nakon toga je potrebno dodati fizički uređaj na *IoT Hub*, a to se radi preko izbornika sa lijeve strane i opcije *IoT devices*. Potrebno je dati ID uređaju što je obično tekst koji jednoznačno opisuje uređaj. Obično se ovaj korak revidira kada se postavi IoT modul te se prilagode potrebne stvari da cijela cijelina proradi.

6.7. Programiranje IoT modula i slanje podataka u Cloud

Nakon što se modul sastavio, potrebno je programirati ponašanje modula. Programiralo se upotrebom razvojne okoline Arduino IDE. U nastavku će se prikazati kod koji se koristi u programskom primjeru za IoT modul.

```

/*****
* Include Libraries
*****/
#include <ESP8266WiFi.h>
#include <ConfigManager.h>
#include <WiFiClientSecure.h>
#include <DNSServer.h>          //Local DNS Server used for redirecting all requests to
the configuration portal
#include <ESP8266WebServer.h>    //Local WebServer used to serve the configuration
portal
#include <WiFiManager.h>        //https://github.com/tzapu/WiFiManager WiFi
Configuration Magic
#include <time.h>
#include <Wire.h>
#include <SPI.h>
#include <Adafruit_Sensor.h>
#include <Adafruit_BME280.h>

#include <AzureIoTHub.h>
#include <AzureIoTProtocol_MQTT.h>
#include <AzureIoTUtility.h>

/*****
* Define Constants
*****/

// Physical device information for board and sensor
#define DEVICE_ID "ESP8266"
#define DHT_TYPE DHT22

// Pin layout configuration
#define LED_PIN 5
#define DHT_PIN 2

#define TEMPERATURE_ALERT 30

// Interval time(ms) for sending message to IoT Hub
#define INTERVAL 10000

// If don't have a physical DHT sensor, can send simulated data to IoT hub
#define SIMULATED_DATA false

// EEPROM address configuration
#define EEPROM_SIZE 512

// SSID and SSID password's length should < 32 bytes
// http://serverfault.com/a/45509
#define SSID_LEN 32
#define PASS_LEN 32
#define CONNECTION_STRING_LEN 256

```

```

#define CONNECTION_STRING_LEN 256

#define MESSAGE_MAX_LEN 256

#define BME_SCK D5
#define BME_MISO D6
#define BME_MOSI D7
#define BME_CS D8

#define SEALEVELPRESSURE_HPA (1013.25)

namespace{
    const int CONNECTION_RETRY_NUMBER = 5;
    const int CONNECTION_SUBTRIES_NUMBER = 5;
}

struct Config {
    char name[20];
    bool enabled;
    int8 hour;
} config;

static WiFiClientSecure sslClient; // for ESP8266

const char *onSuccess = "\"Successfully invoke device method\"";
const char *notFound = "\"No method found\"";

int photocellPin = 0; // the cell and 10K pulldown are connected to a0
int photocellReading; // the analog reading from the analog resistor divider

/*****
 * Initialize a global instance
 *****/
WiFiClient espClient;
ConfigManager configManager;
WiFiManager wifiManager;
static IOTHUB_CLIENT_LL_HANDLE IoTHubClientHandle;

static bool messagePending = false;
static bool messageSending = true;

static char *connectionString = "*****";
static char *ssid;
static char *pass;

static int interval = INTERVAL;

//Adafruit_BME280 bme; // I2C
Adafruit_BME280 bme(BME_CS); // hardware SPI
//Adafruit_BME280 bme(BME_CS, BME_MOSI, BME_MISO, BME_SCK); // software SPI

/*****
 * Auxiliar Functions
 *****/

```

```

/*****
* Auxiliar Functions
*****/

void printAvailableNetworks() {
    WiFi.scanNetworks();
    delay(500);
    int number = WiFi.scanComplete();

    Serial.printf("%d Networks found!\n", number);
    for(int i = 0; i < number; i++) {
        Serial.printf("%d: %s, Ch:%d (%dBm) %s\n", i + 1, WiFi.SSID(i).c_str(), WiFi.channel(i),
        WiFi.RSSI(i), WiFi.encryptionType(i) == ENC_TYPE_NONE ? "open" : "");
    }
}

void blinkLED()
{
    digitalWrite(LED_PIN, HIGH);
    delay(500);
    digitalWrite(LED_PIN, LOW);
}

boolean connectionSucceeded() {
    int subtries_count = 0, tries_count = 1;

    while(WiFi.status() != WL_CONNECTED && tries_count <=
    CONNECTION_RETRY_NUMBER) {
        subtries_count = 0;
        tries_count++;
        Serial.printf("[%d] Connectiong", tries_count);
        while(WiFi.status() != WL_CONNECTED && subtries_count <
    CONNECTION_SUBTRIES_NUMBER) {
            delay(500);
            Serial.print(".");
            subtries_count++;
        }
    }

    return WiFi.status() == WL_CONNECTED;
}

boolean wifiInitiated() {
    printAvailableNetworks();
    wifiManager.autoConnect("AutoConnectAP");
    if(connectionSucceeded()){
        String ssid = WiFi.SSID();
        Serial.printf("Connected to network: %s\n", ssid.c_str());
        return true;
    } else {
        /* Access Point configuration */
        configManager.setAPName(AP_NAME);
        configManager.addParameter("name", config.name, 20);
        configManager.addParameter("enabled", &config.enabled);
        configManager.addParameter("hour", &config.hour);
        configManager.begin(config);
        return false;
    }
}

```

```

    configManager.begin(config);
    return false;
}
}

static void sendCallback(IOTHUB_CLIENT_CONFIRMATION_RESULT result, void
*userContextCallback)
{
    if (IOTHUB_CLIENT_CONFIRMATION_OK == result)
    {
        Serial.println("Message sent to Azure IoT Hub.");
        blinkLED();
    }
    else
    {
        Serial.println("Failed to send message to Azure IoT Hub.");
    }
    messagePending = false;
}

static void sendMessage(IOTHUB_CLIENT_LL_HANDLE iotHubClientHandle, char *buffer,
bool temperatureAlert)
{
    IOTHUB_MESSAGE_HANDLE messageHandle =
IoTHubMessage_CreateFromByteArray((const unsigned char *)buffer, strlen(buffer));
    if (messageHandle == NULL)
    {
        Serial.println("Unable to create a new IoTHubMessage.");
    }
    else
    {
        MAP_HANDLE properties = IoTHubMessage_Properties(messageHandle);
        //Map_Add(properties, "temperatureAlert", temperatureAlert ? "true" : "false");
        Serial.printf("\nSending message: %s.\r\n", buffer);
        if (IoTHubClient_LL_SendEventAsync(iotHubClientHandle, messageHandle, sendCallback,
NULL) != IOTHUB_CLIENT_OK)
        {
            Serial.println("Failed to hand over the message to IoTHubClient.");
        }
        else
        {
            messagePending = true;
            Serial.println("\nIoTHubClient accepted the message for delivery.");
        }

        IoTHubMessage_Destroy(messageHandle);
    }
}

void start()
{
    Serial.println("Start sending temperature and humidity data.");
    messageSending = true;
}

void stop()

```



```

void start()
{
    Serial.println("Start sending temperature and humidity data.");
    messageSending = true;
}

void stop()
{
    Serial.println("Stop sending temperature and humidity data.");
    messageSending = false;
}

IOTHUBMESSAGE_DISPOSITION_RESULT
receiveMessageCallback(IOTHUB_MESSAGE_HANDLE message, void *userContextCallback)
{
    IOTHUBMESSAGE_DISPOSITION_RESULT result;
    const unsigned char *buffer;
    size_t size;
    if (IoTHubMessage_GetByteArray(message, &buffer, &size) != IOTHUB_MESSAGE_OK)
    {
        Serial.println("Unable to IoTHubMessage_GetByteArray.");
        result = IOTHUBMESSAGE_REJECTED;
    }
    else
    {
        /*buffer is not zero terminated*/
        char *temp = (char *)malloc(size + 1);

        if (temp == NULL)
        {
            return IOTHUBMESSAGE_ABANDONED;
        }

        strncpy(temp, (const char *)buffer, size);
        temp[size] = '\0';
        Serial.printf("Receive C2D message: %s.\r\n", temp);
        free(temp);
        blinkLED();
    }
    return IOTHUBMESSAGE_ACCEPTED;
}

int deviceMethodCallback(const char *methodName, const unsigned char *payload, size_t size,
unsigned char **response, size_t *response_size, void *userContextCallback)
{
    Serial.printf("Try to invoke method %s.\r\n", methodName);
    const char *responseMessage = onSuccess;
    int result = 200;

    if (strcmp(methodName, "start") == 0)
    {
        start();
    }
    else if (strcmp(methodName, "stop") == 0)
    {
        stop();
    }
}

```

```

else if (strcmp(methodName, "stop") == 0)
{
    stop();
}
else
{
    Serial.printf("No method %s found.\r\n", methodName);
    responseMessage = notFound;
    result = 404;
}

*response_size = strlen(responseMessage);
*response = (unsigned char *)malloc(*response_size);
strncpy((char *)(*response), responseMessage, *response_size);

return result;
}

void initTime()
{
    time_t epochTime;
    configTime(0, 0, "pool.ntp.org", "time.nist.gov");

    while (true)
    {
        epochTime = time(NULL);

        if (epochTime == 0)
        {
            Serial.println("Fetching NTP epoch time failed! Waiting 2 seconds to retry.");
            delay(2000);
        }
        else
        {
            Serial.printf("Fetched NTP epoch time is: %lu.\r\n", epochTime);
            break;
        }
    }
}

bool readMessage(int messageId, char *payload)
{
    float temperature = getTemperatureValue();
    float humidity = getHumidityValue();
    int light = getLightReadingValue();
    String lightString = getLightReadingString(light);
    DynamicJsonBuffer jsonBuffer;
    JsonObject &root = jsonBuffer.createObject();
    root["deviceId"] = DEVICE_ID;

    time_t rawtime;
    struct tm * timeinfo;

    time ( &rawtime );
    timeinfo = gmtime ( &rawtime );

```

```

time ( &rawtime );
timeinfo = gmtime ( &rawtime );

char id[15];

int year = (timeinfo->tm_year + 1900);
for(int i = 0; i < 4; i++) {
    int current = year / pow(10, 3 - i);
    int temp = int(year) % int(pow(10, 3 - i));
    id[i] = current + 48;
    year = temp;
}

int month = (timeinfo->tm_mon + 1);
if(month < 10) {
    id[4] = '0';
} else {
    id[4] = (month / 10) + 48;
    month %= 10;
}
id[5] = month + 48;

int day = timeinfo->tm_mday;
if(day < 10) {
    id[6] = '0';
} else {
    id[6] = (day / 10) + 48;
    day %= 10;
}
id[7] = day + 48;

int hour = timeinfo->tm_hour;
if(hour < 10) {
    id[8] = '0';
} else {
    id[8] = (hour / 10) + 48;
    hour %= 10;
}
id[9] = hour + 48;

int minute = timeinfo->tm_min;
if(minute < 10) {
    id[10] = '0';
} else {
    id[10] = (minute / 10) + 48;
    minute %= 10;
}
id[11] = minute + 48;

int sec = timeinfo->tm_sec;
if(sec < 10) {
    id[12] = '0';
} else {
    id[12] = (sec / 10) + 48;
    sec %= 10;
}

```

```

        sec %= 10;
    }
    id[13] = sec + 48;
    id[14] = '\0';

    root["messageId"] = id;

    // NAN is not the valid json, change it to NULL
    if (std::isnan(temperature))
    {
        root["temperature"] = NULL;
    }
    else
    {
        root["temperature"] = temperature;
    }

    if (std::isnan(humidity))
    {
        root["humidity"] = NULL;
    }
    else
    {
        root["humidity"] = humidity;
    }

    if(std::isnan(light)) {
        root["light"] = NULL;
    } else {
        root["light"] = light;
    }

    root["lightstring"] = lightString;

    root.printTo(payload, MESSAGE_MAX_LEN);
    return false;
}

String getLightReadingString(int photocellReading) {

    String lightString;
    if (photocellReading < 10) {
        lightString = "dark";
    } else if (photocellReading < 200) {
        lightString = "dim";
    } else if (photocellReading < 500) {
        lightString = "light";
    } else if (photocellReading < 800) {
        lightString = "bright";
    } else {
        lightString = "very bright";
    }

    return lightString;
}

```

```

    lightString = "very bright";
}

return lightString;
}

int getLightReadingValue() {
    photocellReading = analogRead(photocellPin);
    return photocellReading;
}

float getTemperatureValue() {
    return bme.readTemperature();
}

String getTemperatureSign() {
    return "°C";
}

float getPressureValue() {
    return bme.readPressure() / 100.0F;
}

String getPressureSign() {
    return "hPa";
}

float getHumidityValue() {
    return bme.readHumidity();
}

String getHumiditySign() {
    return "%";
}

/*****
* Main Functions
*****/

void setup() {
    Serial.begin(9600);
    pinMode(LED_PIN, OUTPUT);
    if(wifiInitiated()) {
        initTime();
        iotHubClientHandle = IoTHubClient_LL_CreateFromConnectionString(connectionString,
MQTT_Protocol);
        if (iotHubClientHandle == NULL)
        {
            Serial.println("Failed on IoTHubClient_CreateFromConnectionString.");
            while (1);
        }
        Serial.println("Connected to IoTHub");
        IoTHubClient_LL_SetMessageCallback(iotHubClientHandle, receiveMessageCallback,
NULL);
        IoTHubClient_LL_SetDeviceMethodCallback(iotHubClientHandle, deviceMethodCallback,
NULL);

```

```

    IoTHubClient_LL_SetDeviceMethodCallback(iotHubClientHandle, deviceMethodCallback,
    NULL);

    int status = bme.begin();
    if (!status) {
        Serial.println("Could not find a valid BME280 sensor, check wiring!");
        while (1);
    }

    Serial.println("-- Default Test --");

    Serial.println();

    delay(1000); // let sensor boot up
}

}

static int messageCount = 1;
void loop() {
    if(WiFi.status() != WL_CONNECTED) {
        configManager.loop();
    }
    if (!messagePending && messageSending)
    {
        char messagePayload[MESSAGE_MAX_LEN];
        bool temperatureAlert = readMessage(messageCount, messagePayload);
        Serial.printf("JSON: %s\n", messagePayload);
        sendMessage(iotHubClientHandle, messagePayload, temperatureAlert);
        messageCount++;
        delay(interval);
    }
    IoTHubClient_LL_DoWork(iotHubClientHandle);
    delay(10);
}

```

Navedeni programski primjer funkcionira na sljedeći način. Prilikom pokretanja modula, prvo se provjerava ima li modul bežičnu mrežu. Ukoliko modul nema bežičnu mrežu, onda se pokreće postavljanje bežične mreže. Postava bežične mreže kreira mrežu zvanu „AutoConnectAP“ te je potrebno odabrati tu mrežu na nekom uređaju i odabrati bežičnu mrežu koja za povezivanje zahtjeva isključivo samo lozinku. Ova postava trenutno ne podržava način prijave kakav je kod eduroam mreže. Nakon što se spoji na „AutoConnectAP“, trebao bi se pokrenuti Internet preglednik sa stranicom za podešavanje, a ukoliko se ne pokrene samo od sebe: potrebno je u Internet pregledniku utipkati IP adresu: 192.168.4.1. Nakon toga se odabire *Wifi scan* te u sljedećem prikazu imamo popis dostupnih mreža. Odabere se jedna i unese se

lozinka te se započne spajanje. Modul se ponovno pokrene te se spoji na Internet sa bežičnom mrežom koja je bila odabrana.

Nakon toga slijedi povezivanje na Azure, bitno je napomenuti da je potrebno zalijepiti *connection string* kako bi povezivanje bilo uspješno. Koristi se vrijednost polja *Connection String Primary Key*. (Napomena: radi sigurnosnih razloga moj *connection string* je skriven.) Povezivanje na Azure se pokreće te kada je povezivanje uspješno, modul ulazi u petlju te šalje očitavanja u periodičkim intervalima. Očitavanja u obliku JSON-a odlaze na Azure i spremaju se u spremištu podataka.

6.8. Podešavanje povezivanja Azure-a sa Power BI -om

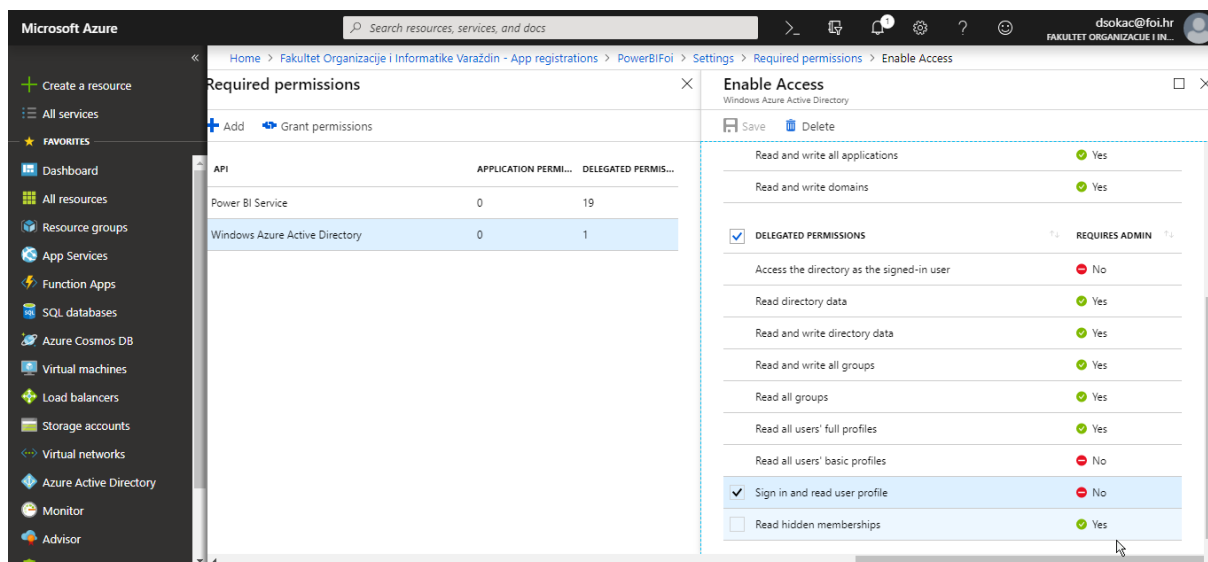
Ideja oko ovog povezivanja je da se omoguće podaci u realnom vremenu. Iako je već kreirani *job* na Azur-u koji to sprema u tablicu na Azuru. Sada je potrebno kreirati novi *Azure Stream Analytics Job*. U tom novom *job-u*, ulaz je tipa *Stream*, a izvor podataka mu je *IoT Hub*. Izlaz u tom novom *job-u* mora imati odabrano *Power BI* pri kreiranju te se jednostavno unesu podaci koje traži te se autorizira sa *Power BI*-om. I nakon što se pokrene i uspješno prođe, trebalo bi biti moguće vidjeti tu novu tablicu u *Power BI*-u i onda se izvještaji mogu kreirati po toj tablici.

6.9. Podešavanje povezivanja Azure-a, Power BI-a i Microsoft Dynamics NAV-a

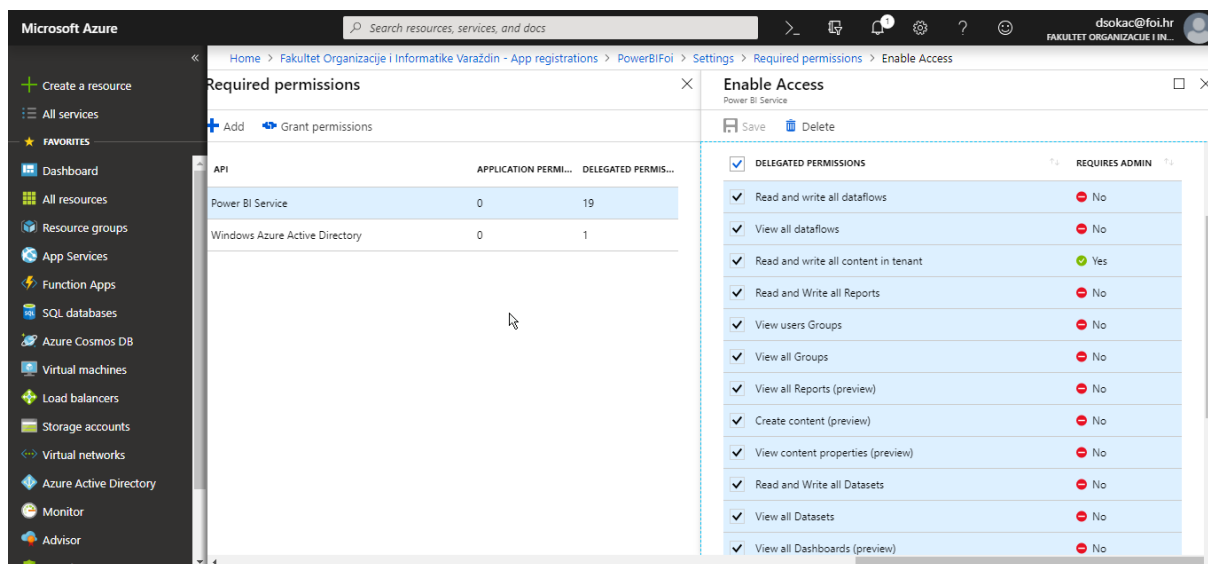
NAV podržava mogućnost prikazivanja *Power BI* izvještaja. Isto tako nudi i vođeno postavljanje veze prema Azure-u do kojeg se može doći na dva načina:

1. preko početnog zaslona i *Power BI Reports* prozorom koji je prikazan ukoliko je NAV na tvorničkim postavkama
2. preko akcije „Vođeno postavljanje i zadaci“ (*eng. Assisted Setup and Tasks*) te se onda odabere „Postava Azure aktivnog direktorija“ (*eng. Setup Azure Active Directory*)

Kada se pokrene čarobnjak za postavljanje, samo se prate koraci koje on nudi i sve će uspješno proći. Na slikama 6.9.1 i 6.9.2 se nalaze prava koja zahtjeva aplikacija za Windows Azure aktivni direktorij i za *Power BI* servis.



Slika 6.9.1 Prikaz potrebnih prava za Microsoft Azure aktivni direktorij (rad autora)

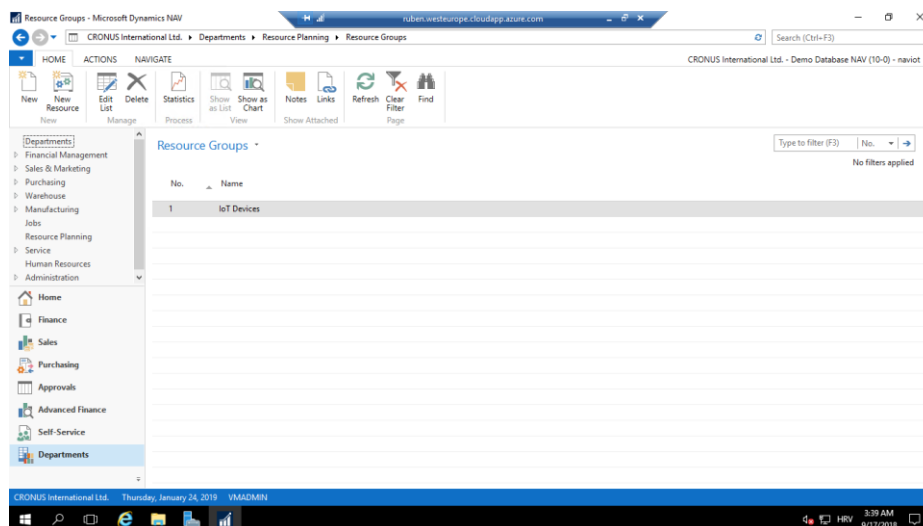


Slika 6.9.2 Prikaz potrebnih prava za Power BI servis (rad autora)

Nakon uspješnog postavljanja, čarobnjak može javiti da trenutno prijavljen korisnik nije administrator pa ne može dodijeliti prava aplikaciji te kaže da treba javiti administratoru da dodjeli prava aplikaciji kako bi se nastavilo. U ovom koraku, ako niste administrator, nikako ne dozvolite administratoru da se prijavi u NAV-u i dozvoli prava. Administrator mora dodijeliti prava preko Azure aktivnog direktorija i u opciji Registrirane aplikacije (*eng. App registrations*). Mora odabrati aplikaciju kojoj želi dodijeliti prava, nakon tog u slijedećem prozoru odabrati „Postavke“ (*eng. Settings*) i na kraju „Potrebne dozvole“ (*eng. Required Permissions*) te na slijedećem prozoru odabrati „Dodjeli prava“ (*eng. Grant Permissions*).

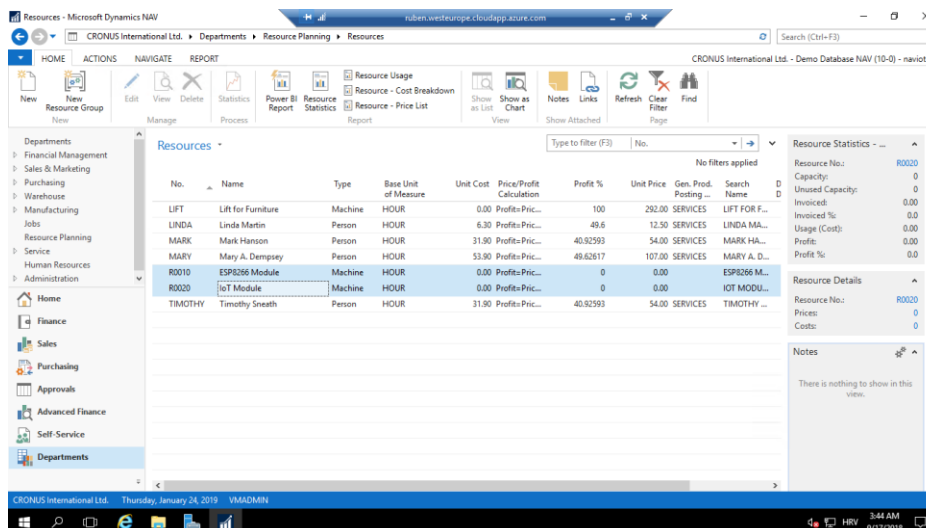
6.10. Integracija modula u IS

Integracija modula u IS se vrši korištenjem razvojne okoline *Microsoft Dynamics NAV*-a. Ideja oko integracije je dodati novu grupu resursa vezanu za IoT uređaje te u tu grupu resursa dodati specifične IoT uređaje. Nakon toga je potrebno kreirati izvještaj u *Power BI*-u koji ima isti naziv kao ID dodanog resursa Internet uređaja.



Slika 6.10.1 Prikaz grupe resursa (rad autora)

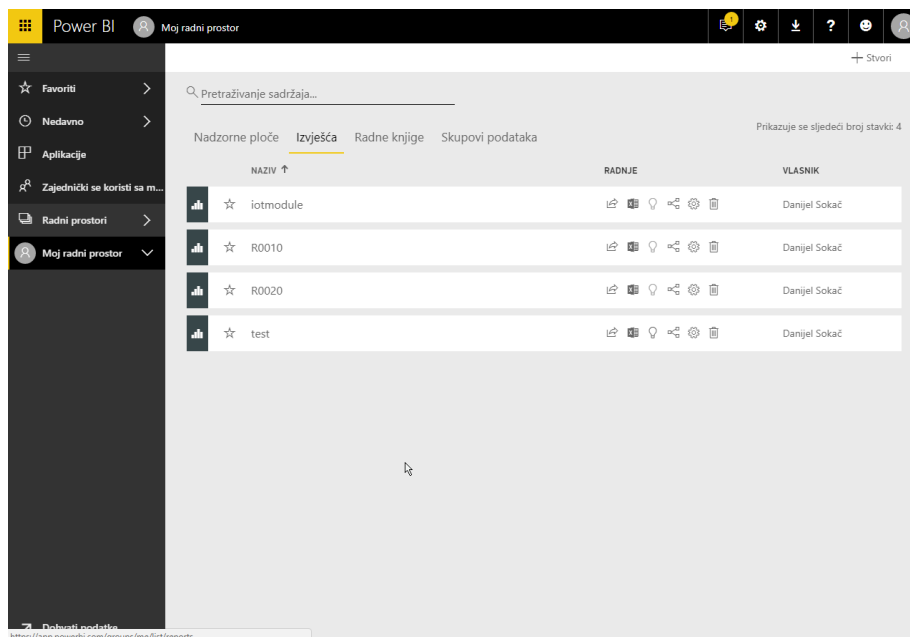
Slika 6.10.1 prikazuje prikaz trenutno kreiranih grupa resursa. Ta tablica je u početku prazna pa je potrebno da se doda grupa resursa prije nego se dodaju resursi IoT stvari.



Slika 6.10.2 Prikaz resursa sa označenim resursima prethodno kreirane grupe resursa (rad autora)

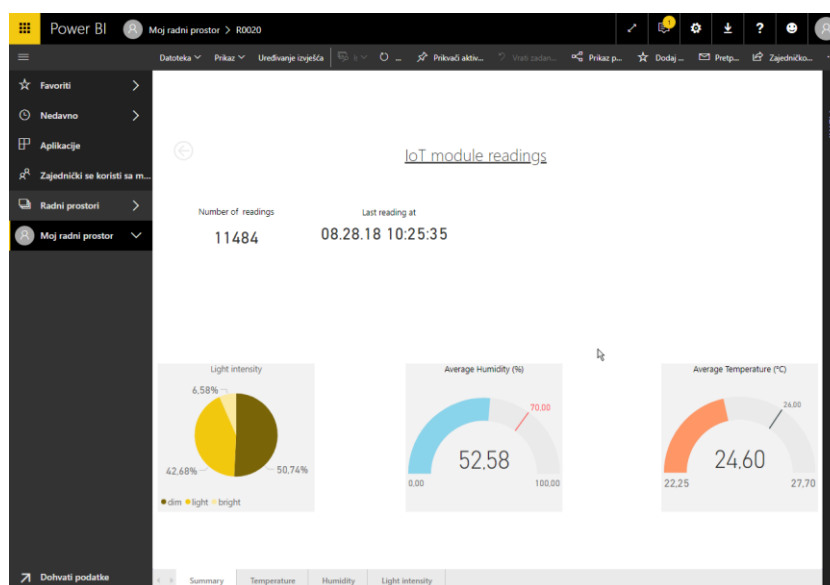
Na slici 6.10.2 se vide dostupni resursi u NAV-u. Označeni resursi su novi resursi kreirani sa grupom resursa koja se izradila u prethodnom koraku. Ovo je vrlo bitno jer funkcionalnost

promatra grupu resursa Internet stvari prilikom provjera. Potrebno je također i zapamtiti ID novokreiranog uređaja.

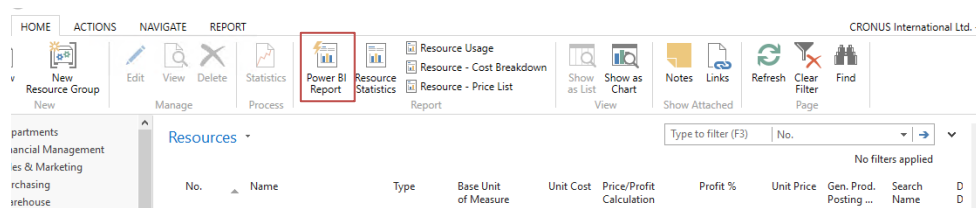


Slika 6.10.3 Izgled *Power BI* radnog prostora sa izvještajima (rad autora)

Na slici 6.10.3 se nalaze izvještaji koji se koriste u praktičnom primjeru. Potrebno je primijetiti da postoje neki izvještaji, koji imaju naziv jednak identifikacijskom stupcu u ERP-u, i oni se koriste u implementaciji primjera. Akcija je dizajnirana na način da ispituje ima li identifikacijska vrijednost iz tablice resursa kao naziv iz popisa izvještaja. Na slici 6.10.4 se može vidjeti izgled izvještaja „R0020“ koji se prikazuje u ERP-u. Postoji još jedan izvještaj „R0010“ koji je prazan kako bi se demonstriralo da ERP traži točno izvještaj koji mu treba.

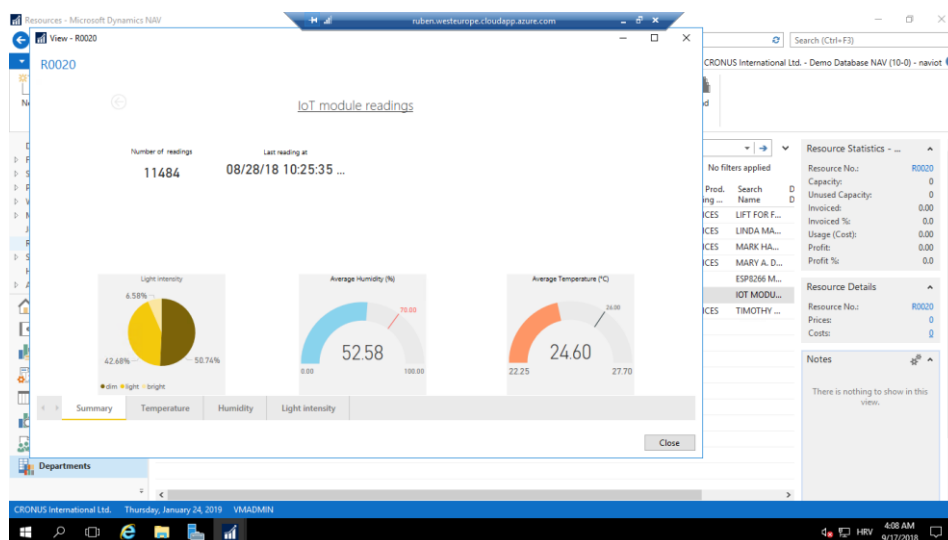


Slika 6.10.4 Primjer izvještaja (rad autora)



Slika 6.10.5 Prikaz trake s akcijama kod tablice resursa (rad autora)

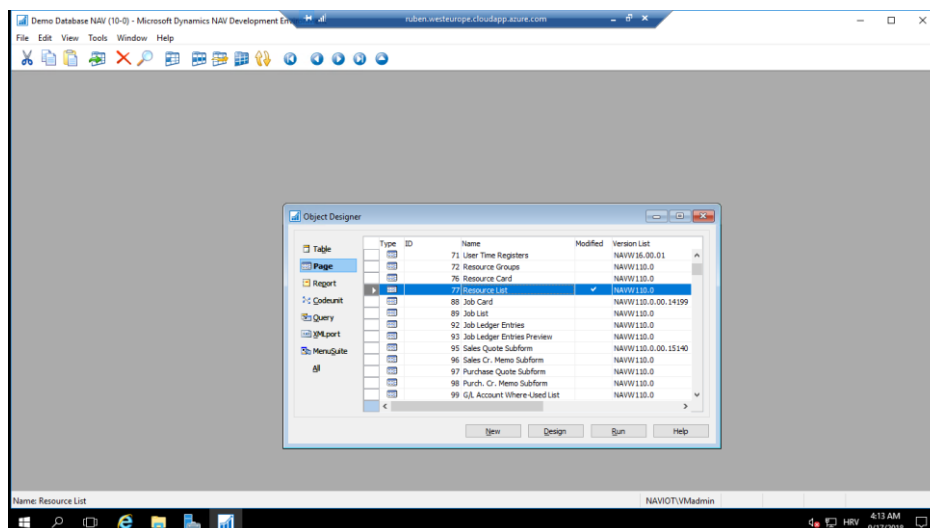
Na slici 6.10.5 je označena nova akcija koja je kreirana za potrebu praktičnog primjera, a njezina zadaća je otvoriti *Power BI* izvještaj za označeni IoT uređaj u tablici. Što znači da je potrebno odabrati resurs čiji se izvještaj želi prikazati prije pokretanja akcije. Nakon ispravno provedenih koraka trebali bi vidjeti izvještaj sličan onome prikazanom na slici 6.10.6.



Slika 6.10.6 Prikaz izvještaja otvorenog u ERP-u (rad autora)

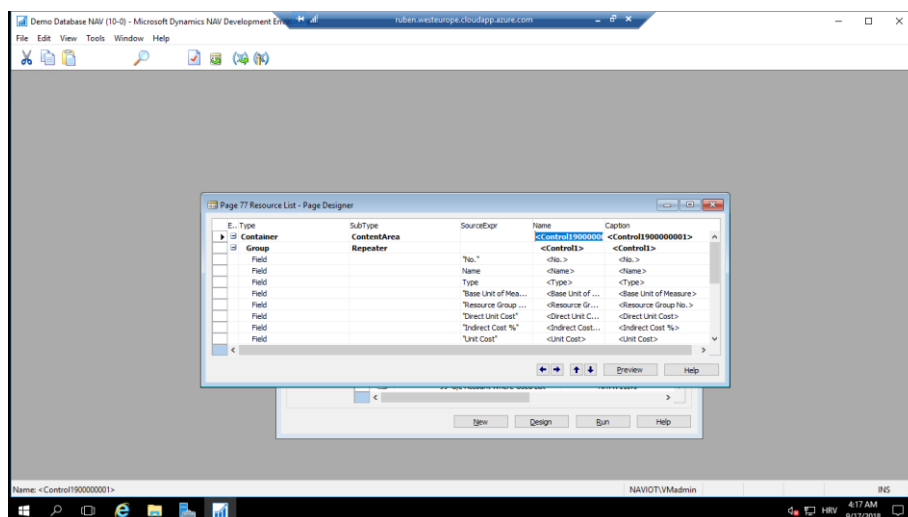
Akcija „Power BI Report“

Sada će se detaljno objasniti procedura kreiranja akcije za pokretanje izvještaja. Akcija je kreirana pomoću razvojne okoline NAV-a.

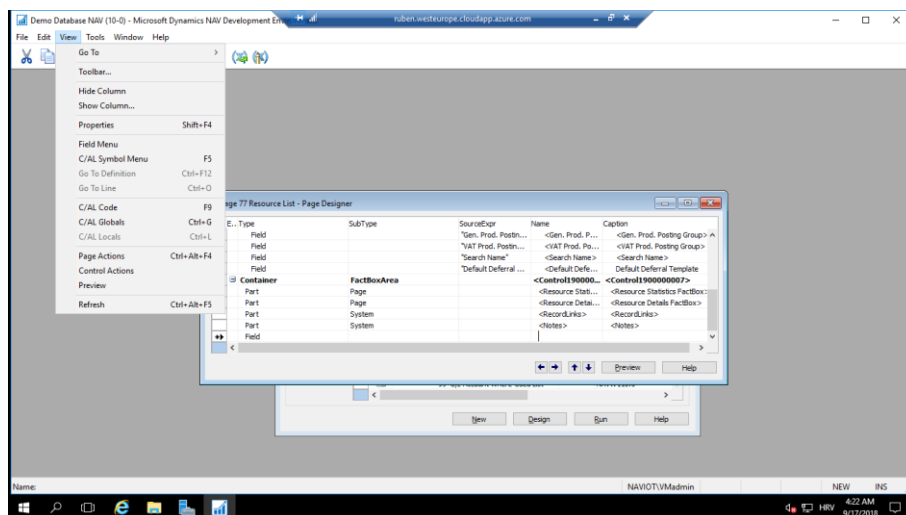


Slika 6.10.7 Prikaz popisa stranica u razvojnoj okolini NAV-a (rad autora)

Prvo je potrebno pronaći koja stranica predstavlja stranicu sa listom resursa. Na slici 6.6.7 je ta stranica označena trenutno te uz sebe ima oznaku da je promijenjena. Ta oznaka se dodaje samo na one elemente koji se naknadno modificiraju. Kada se klikne na gumb „Dizajn“ (*eng. Design*), onda se prikazuje slijedeći prozor kao na slici 6.10.8. Na slici 6.10.8 se vide elementi stranice te se tu oni mogu mijenjati. S obzirom da trenutno se ne rade promjene u dizajnu stranice nego se programira akcije pa stoga ni ne treba odabirati ništa u ovom prozoru i prije nastavka treba pozicionirati kursor na novi red na dnu.

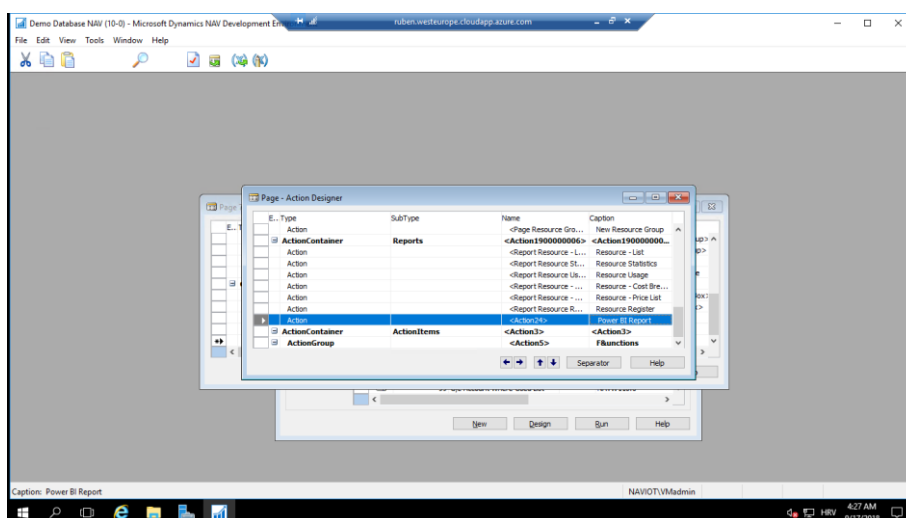


Slika 6.10.8 Prikaz elemenata stranice sa popisom resursa (rad autora)



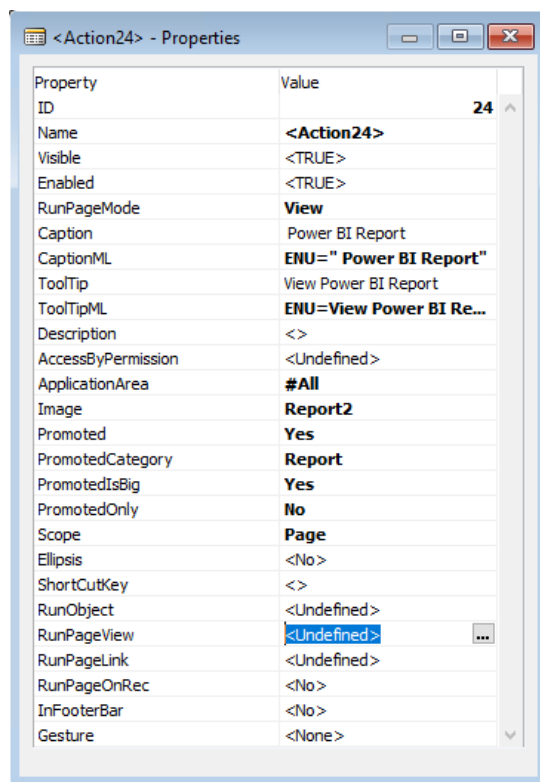
Slika 6.10.9 Prikaz izbornika „Izgled“ (*eng. View*) (rad autora)

Nakon toga se odabire izbornik „Izgled“ kako je prikazano na slici 6.10.9 i u tom izborniku treba se odabrati opcija „Akcije stranice“ (*eng. Page Actions*) koje će prikazivati sve akcije definirane za tu stranicu kao što je prikazano na slici 6.10.10. Trenutno je na slici već kreirana akcija koja treba, ali će se prikazati postupak kreiranja nove akcije.



Slika 6.10.10 Prikaz akcija stranice (rad autora)

Akcije su grupirane u grupe te se na taj način i prikazuju u ERP-u stoga je potrebno odabrati dobru grupu i dodati akciju u tu grupu. Za dodavanje akcije je potrebno pritisnuti desnu tipku miša kako bi se pozvao kontekstualni izbornik na kojem se odabere opcija „Novo“ (*eng. New*) i onda se kreira nova akcija. Drugi način kreiranja je otići na dno prikaza i dodati akciju pa ju odvući gdje je potrebno. Slijedeći korak je podešavanje te novo kreirane akcije čiji je prozor prikazan na slici 6.10.11.

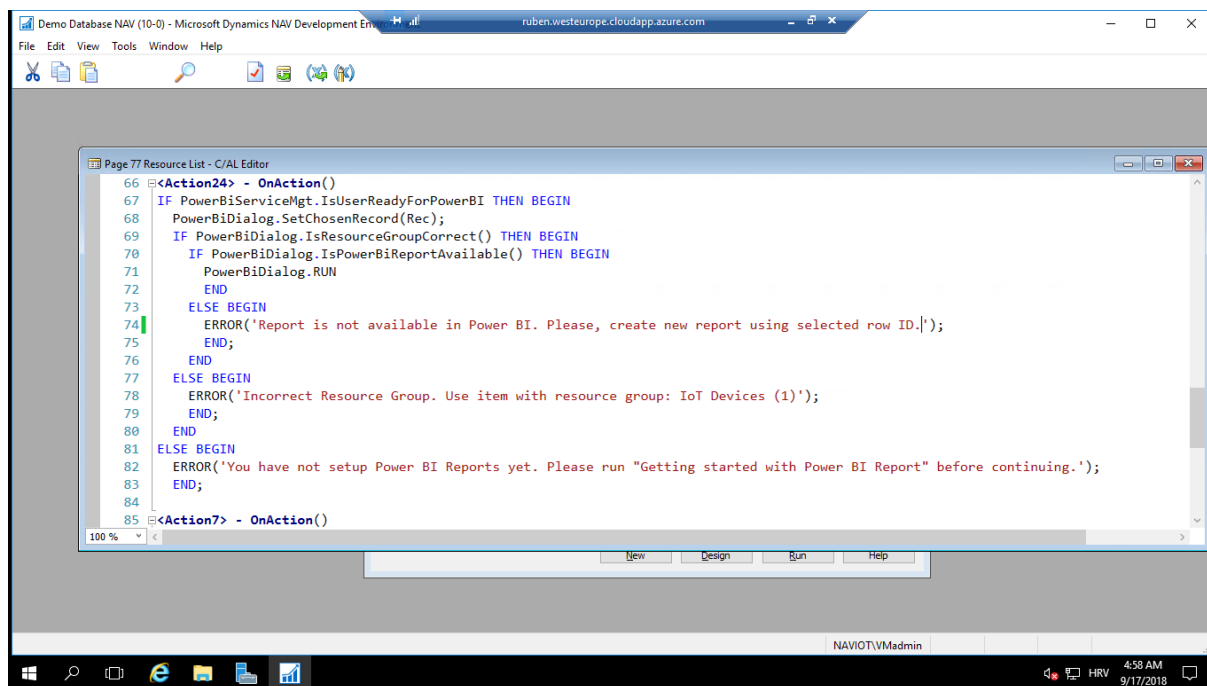


Slika 6.10.11 Prikaz postavki akcije za prikaz *Power BI* izvještaja (rad autora)

Ovo su izvorne postavke za akciju koja je implementirana trenutno. Kada se kreira nova akcije, potrebno je preslikati izgled ovog prozora na vaš prozor da bi sve radilo. Može se primijetiti redom neke opcije koje su se mijenjale. *RunPageMode* je opcija koja označava način na koji će se pokretati stranica koju otvara akcija. U našem slučaju je idealan način pregledavanje (eng. *View*). Slijedi naslov akcije (eng. *Caption*) koji sadrži tekst vidljiv ispod ikone akcije. Nakon toga slijedi dodatno objašnjenje (eng. *Tooltip*) koji prikazuje kratak opis da korisniku bude jasno što ta akcija radi. *ApplicationArea* je opcija koja definira kada će se ta akcija prikazivati. Za potrebe diplomskog rada, ta opcija nosi vrijednost „#All“ što označava da će se prikazivati uvijek neovisno o korisničkim ulogama. Slijedi odabir slike ikone i tu se odabire neka slika koja je prikladna za akciju. Na žalost u desktop verziji ERP-a, nije moguće dodati vlastite slike. Slijedi opcija *Promoted* koja označava hoće li se akcija prikazati na prozoru ili neće. *PromotedCategory* je opcija koja sadrži grupu gdje će se prikazati akcija i za potrebe diplomskog je odabrana grupa izvještaja što je i logično. *PromotedIsBig* opcija označava hoće li akcija biti istaknuta ili neće. Istaknute akcije su veće i zadržavaju takav oblik neovisno o veličini prozora dok se ostale smanjuju. Na kraju se postavlja svojstvo opsega (eng. *Scope*) koji je postavljen na vrijednost da je opseg ove akcije trenutna stranica. Ova opcija također utječe na prikaz akcije u prozoru. Obično se definiraju i objekti koje ta akcija mora pokrenuti, ali za potrebe diplomskog ne treba jer se prosljeđuju podaci prema akciji i izvršavaju kontrole prije

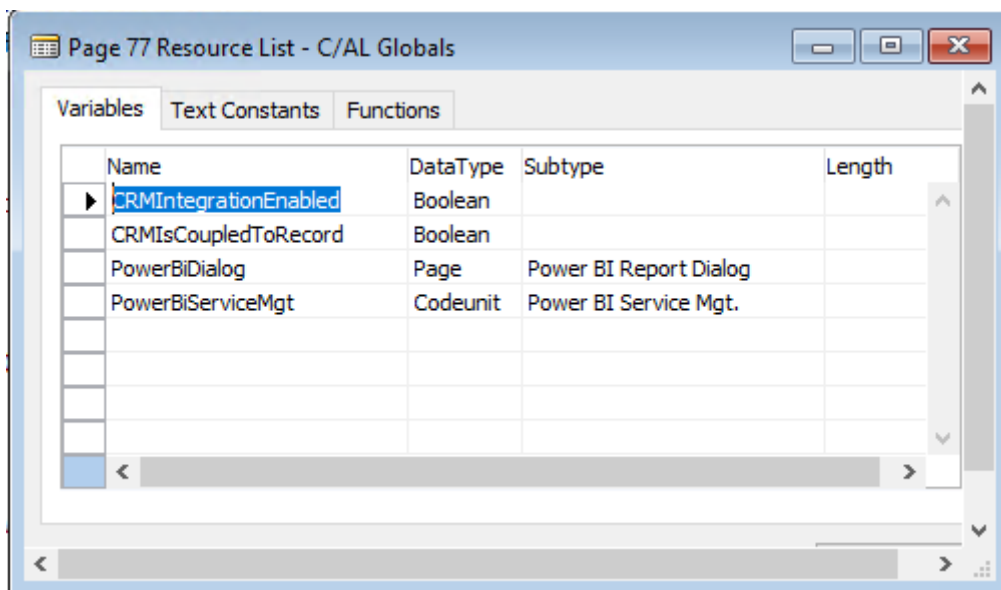
samog pokretanja izvještaja. Na kraju je potrebno spremite postavke pritiskom na prečicu Ctrl + S.

Akcija se može pokrenuti i programski, a da bi vidjeli kod od akcije potrebno je označiti akciju pa ići na izbornik „Izgled“ (*eng View*) i odabrati opciju *C/AL Code* te će se otvoriti prozor prikazan na slici 6.10.12.



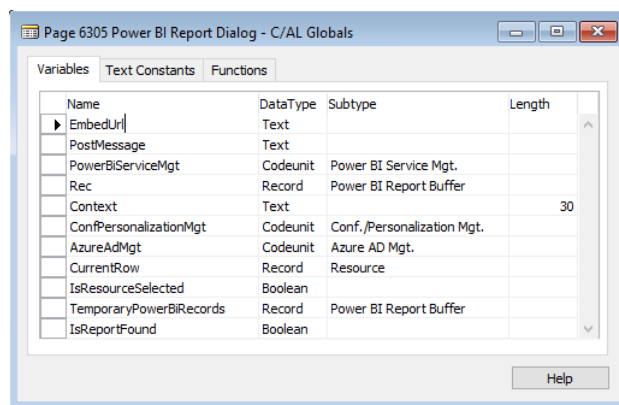
Slika 6.10.12 Prikaz programskog koda akcije (rad autora)

Kada se kreira nova akcija onda neće biti koda nikakvog već prazni red te je potrebno dodati kod sa slike 6.6.12 kako bi sve radilo. U kodu postoje neke funkcije koje će se morati dodati naknadno pa se nakon kreiranih funkcija vratimo na ovu sliku za dovršavanje akcije. Kada se izvrše promijene koje se žele spremite, potrebno je odabrati prečicu Ctrl + S za spremanje. Obično je teško identificirati što koja varijabla radi i odakle je. Stoga je odabrati opciju *C/AL Globals* u izborniku „Izgled“ (*eng. View*). Nakon toga se otvara prozor prikazan na slici 6.10.13.

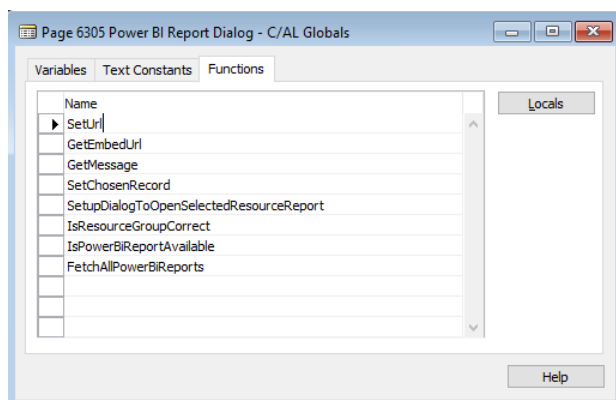


Slika 6.10.13 Prikaz globalnih varijabli za programiranje akcije (rad autora)

Na slici 6.10.13 se nalaze globalne varijable koje se koriste kod programiranja. Kada se kreira nova akcija, samo prve dvije varijable će biti dostupne, a druge dvije se moraju dodati. *PowerBiDialog* je varijabla koja drži referencu na stranicu *Power BI Report Dialog* koja će se otvoriti. Ta stranica je već u ERP-u pa ju je samo potrebno definirati i kasnije malo modificirati da radi što treba. *Power Bi Service Mgt.* je klasa (*codeunit*) koja sadrži operacije vezane za *Power BI*. Ukoliko su učinjene neke promjene, tada je potrebno spremi pritiskom na prečicu Ctrl + S. Sada je potrebno ići pronaći stranicu *Power BI Report Dialog* kako bi se dodale funkcije prije nego se mogu koristiti u programskom kodu akcije. Potrebno je zatvoriti sve prozore osim dizajnera objekta (*eng. Object Designer*) i onda se klikne na sivu površinu kako bi se maknula selekcija i počne se pisati naziv stranice. Razvojna okolina automatski započinje pretragu pa se stranica prikaže u prozoru ispod. Odabere se red sa stranicom i klikne se na gumb dizajn (*eng. Design*). Kako nema potrebe mijenjati strukturu stranice, treba ponovno otvoriti programski kod (*C/AL Code*) uz pomoć izbornika „Izgled“ (*eng. View*). Prvo je potrebno definirati sve varijable prikazane na slici 6.10.14. i funkcije prikazane na slici 6.10.15. Funkcije se mogu definirati uz pomoć prikaza dijelova koda koji se trebaju dodati, pa tada treba svaku funkciju, koja nije prikazana, kreirati da bi bila prisutna u kodu.

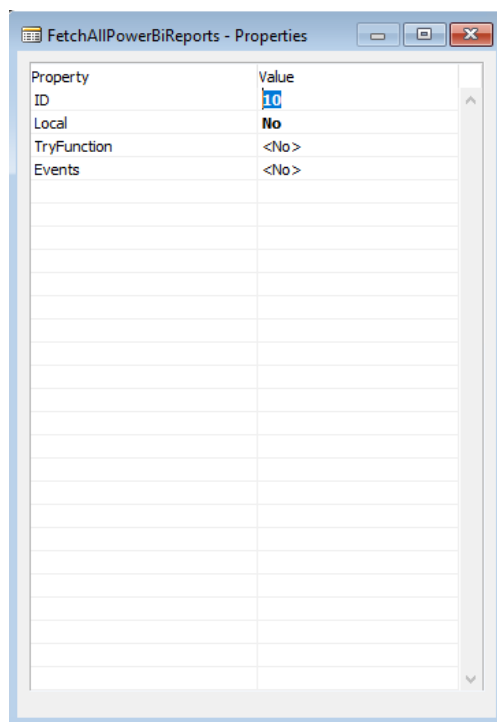


Slika 6.10.14 Prikaz varijabli od stranice *Power Bi Dialog*.(rad autora)



Slika 6.10.15 Prikaz funkcija od stranice *Power Bi Dialog* (rad autora)

Kreiranje funkcije se izvršava na način da se u prvo slijedeće slobodno mjesto u prikazu na slici 6.10.15 dodaje naziv funkcije te se odabere gumb *Locals*. Nakon toga se otvara prozor sa više kartica koje označavaju što se sve može definirati. U kartici parametara (*eng. Parameters*) se definiraju parametri funkcije. U kartici *Return Value* se definira koji tip metoda vraća te je moguće i imenovati taj tip sa poljem „Naziv“ (*eng. Name*), ali to neće biti potrebno u ovom primjeru. Slijedeća kartica je kartica varijabli (*eng. Variables*) u njoj se definiraju lokalne varijable funkcije, ali ta kartica se neće koristiti u primjeru. I zadnja kartica je kartica tekstualnih konstanti (*eng. Text Constants*) koja služi za definiranje tekstualni konstanti i ona se također neće koristiti u ovom primjeru.



Slika 6.10.16 Prikaz postavki funkcije (rad autora)

Inicijalno je funkcija definirana kao lokalna (*eng. Local*) te ukoliko se želi koristiti izvan trenutnog pogleda, potrebno je promijeniti tu definiciju na globalna. Kada se pozicionira na pogled sa slike 6.10.15, onda je potrebno označiti red funkcije čije postavke se žele promijeniti i odabire se opcije „Postavke“ (*eng. Properties*) koja se nalazi u izborniku „Izgled“ (*eng. View*). Nakon što se klikne, otvara se prozor prikazan na slici 6.10.16. I u tom pogledu je potrebno promijeniti vrijednost postavke *Local* na ne (*eng.No*) kako je i prikazano na slici 6.10.16.

Sada će se prvo definirati funkcije koje se koriste prije otvaranja stranice.

```
SetChosenRecord(ChosenRecord : Record Resource)
CurrentRow := ChosenRecord;
IsResourceSelected := TRUE;
FetchAllPowerBiReports();
```

Funkcija *SetChosenRecord* sprema označeni redak te postavlja indikator da je resurs označeni. Kao zadnji korak dohvaća sve dostupne izvještaje iz *Power BI*-a.

```

FetchAllPowerBiReports()
  IF Context = '' THEN
    Context := ConfPersonalizationMgt.GetCurrentProfileID;
  Rec.DELETEALL;
  IF Rec.ISEMPTY THEN
    PowerBiServiceMgt.GetReports(Rec,Context);

```

Funkcija *FetchAllPowerBiReports* postavlja trenutni kontekst te briše spremljene izvještaje iz *Power BI*-a. Nakon toga ponovno dohvaća izvještaje koristeći dostupnu funkciju.

```

IsResourceGroupCorrect() : Boolean
EXIT (IsResourceSelected AND (CurrentRow."Resource Group No." = '1'));

```

Funkcija *IsResourceGroupCorrect* provjerava je li uopće resurs označen i ima li označen red definiranu grupu resursa sa identifikacijskim brojem 1 što označava grupu resursa koja je kreirana za IoT uređaje.

```

IsPowerBiReportAvailable() : Boolean
IsReportFound := FALSE;
IF Rec.FINDSET THEN BEGIN
  REPEAT
    IF Rec.ReportName = CurrentRow."No." THEN
      IsReportFound := TRUE;
      TemporaryPowerBiRecords := Rec;
  UNTIL (Rec.NEXT = 0) OR IsReportFound;
END;
Rec := TemporaryPowerBiRecords;
EXIT(IsReportFound);

```

Funkcija *IsPowerBiReportAvailable* prolazi kroz sve izvještaje koji su spremljeni te uspoređuje je li naziv izvještaja jednak identifikacijskom stupcu resursa u tablici i ako jest onda se postavlja taj izvještaj u memoriju i vraća se je li nađen ili nije. Na taj način se održava kontrola i izbjegavaju se nepredvidive situacije. Sada bi bilo dobro krenuti sa funkcijama da se definira ponašanje stranice.

```

OnOpenPage()
IF IsResourceSelected THEN
    IF IsResourceGroupCorrect() THEN
        SetupDialogToOpenSelectedResourceReport(CurrentRow);

```

Funkcija *OnOpenPage()* je puna selekcijskih konstrukta jer se ova stranica koristi i za prikaz *Power BI* izvještaja što je već implementirano u ERP. Kako bi se izbjegli problemi prvo se provjerava je li označeni resurs i ako jest onda se provjerava je li odabrana dobra grupa resursa te ako i to odgovara, onda se pokreće funkcija za otvaranje stranice za potrebe akcije koju definiramo.

```

LOCAL SetupDialogToOpenSelectedResourceReport()
CurrPage.CAPTION := Rec.ReportName;
IF EmbedUrl = '' THEN
    EmbedUrl := Rec.EmbedUrl;
    EmbedUrl := GetEmbedUrl;
    CurrPage.UPDATE;
IF PostMessage = '' THEN
    PostMessage := GetMessage;

```

Lokalna funkcija *SetupDialogToOpenSelectedResourceReport* dodjeljuje naslov prozoru te si dohvaća potrebne podatke za slanje zahtjeva prema *PowerBI*-u. Dobro je napomenuti da se funkcije mogu pisati i bez zagrada ako nemaju parametre kao što je prikazano u kodu iznad. Znači *GetEmbedUrl* i *GetMessage* su zapravo funkcije. To su bile funkcije koje se pokreću prilikom otvaranja prozora, a dalje odrade predefinirane funkcije za *WebPageViewer* koji prikazuje izvještaj.

Nakon što su sve funkcije napisane, potrebno je spremi pritiskom na prečicu Ctrl + S. I potvrditi odabir u dijaloškom okviru da se spremi. Ako nema grešaka onda je postupak završio i potrebno je pokrenuti NAV i provjeriti je li sve kao što je bilo prikazano na slikama.

6.11. Zaključna analiza

Ovaj primjer je izrađen kao rezultat motivacije o pametnoj poljoprivredi. Primjer možda nije najbolje arhitekturno osmišljen i implementiran. Bilo bi bolje kada moduli koji izvode mjerenja da nisu ovisni o električnoj energiji i bežičnoj mreži. Čim se makne napajanje putem električne energije, odmah bežična mreža postane prevelik teret za modul napajan baterijom te bi se trebao koristiti način komunikacije putem GSM-a. Dobro bi bilo da postoji centralni modul kako je i bilo zamišljeno inicijalno te bi taj modul sakupljao podatke od ostalih i slao ih na

cloud. Uključujući centralne module i centraliziranu organizaciju, potrebno je osmisлити način na koji će se implementirati „upoznavanje“ modula i kako će centralni modul znati da je dobio podatke od tog modula koji stvarno i jest tu odnosno koji je „poznaniк“ centralnom modulu. Kada se razmišlja o skalabilnosti, potrebno je još proširiti definiciju poruka ako se radi o više različitih lokacija i slično. Jedina funkcionalnost, koja nije prikazana, a mogla bi biti korisna za pametnu poljoprivredu, je upravljanje modulima preko Azure-a što znači slanje signala uređaju da pokrene neku akciju.

Power BI je odličan alat, ali sam primjetio da je malo ograničeno kreiranje radne površine koja može prihvaćati podatke u realnom vremenu i to je nedostatak jer se nude samo neke opcije vizualizacije te su podaci striktno vezani za određene vizualizacije. *Power BI* nudi samo neke podatke i ne daje opciju da se mogu odabrati ili unesti neki drugi podaci. Također, je nedostatak to da NAV ne podržava prikaz tih radnih površina sa podacima u realnom vremenu. Korištenje izvještaja je zadovoljavajuće rješenje, ali to nije rješenje u kojem se mogu vidjeti podaci u realnom vremenu.

Ovaj primjer bi se mogao koristiti kao temelj za pametnu poljoprivredu i bilo koju drugu djelatnost koja uključuje ERP sustav kao temelj poslovanja. Naravno da obrtnici neće sada implementirati ERP sustav da bi pratili primjer. Moguće je također izraditi web aplikacije i klijentske aplikacije koje mogu sasvim dovoljno podržavati djelatnost za koju su kreirane. Ideja primjera je bila prikazati da je moguće integrirati IoT u ERP sustav.

7. Zaključak

Tehnologija se mijenja iz dana u dan velikom brzinom. Kratko se pogledao pregled do danas da bi se došlo do digitalne transformacija koja je još poznata kao i četvrta industrijska revolucija. S obzirom da je to relativno novi trend on obuhvaća i novije tehnologije poput IoT, *Cloud*, 3D modeliranje i slično. Digitalna transformacija je prepoznata kao sredstvo postizanja prednosti na tržištu te poduzeća ciljaju na to da uvedu ovaj trend u svoja poduzeća. Kao jedna od značajnih tehnologija za ovaj rad je Internet stvari (IoT). IoT je među popularnijim tehnologijama digitalne transformacije te omogućuje poduzeću bolju vidljivost izvan njegovih granica i poboljšava poslovanje kao i kompetentnost na tržištu.

Slijedeća tema u radu je poslovni informacijski sustav poznat kao ERP. ERP se godinama razvijao do današnjeg dana. Isprva nije imao prevelik utjecaj, ali je bio prepoznati kao korisna tehnologija da bi kasnije rapidno se raširio na području proizvodne industrije pa i na ostalim područjima. ERP je prevelik zalogaj za manja i srednja poduzeća, međutim veća poduzeća također moraju dobro promisliti o implementaciji ERP sustava. Implementacija ERP sustava je proces koji traje i košta. ERP se sastoji od 3 faze: predimplementacijska, implementacijska i postimplementacijska. Faze prate ERP od planiranje, preko razvoja, prilagodbe i konfiguracije do održavanja te na kraju i otkazivanja. Otkazivanje ERP označava samo da je ERP zastario i ne odgovara poduzeću te ono traži novu zamjenu. Spomenute su bili trendovi ERP-a kao što su Cloud ERP koji cilja na smanjivanje troškova jer poduzeće ne mora brinuti ni o infrastrukturi ni o aplikaciji već ju samo pokrene preko interneta. I drugi trend je IoT ERP za koji se navedeni samo neki primjeri.

Tema, po važnosti nakon ERP-a, je integracija IoT-a u ERP. Generalno IoT ugrađen u ERP označava ogroman uspjeh ako je pravilno upravljan. IoT olakšava poslovanje, smanjuje troškove, povećava responzivnost poduzeća, skalabilan je i slično. Postoje mnoštvo članaka koji kažu slično, ali nema ni jedan da definira neku metodologiju implementacije IoT-a u ERP sustav. Zbog toga su predložene faze: faza otkrivanja, faza implementacije i faza korištenja. Te faze se sastoje od svojih koraka i pridržavanje istih bi trebalo rezultirati integriranim IoT uređajem u ERP sustavu. Na kraju svega, izrađen je bio primjer implementacije IoT-a u ERP *Dynamics NAV* koji pokazuje *Power BI* izvještaj za odabrani IoT uređaj.

IoT je vrlo korisna stvar u ovom dobu, pogotovo kada je integriran u ERP. Za daljnja istraživanja bi trebalo izraditi metodologiju uključivanja IoT stvari u ERP u svrhu maksimiziranja poslovnog rezultata.

8. Literatura

- [1] Abdullah A. M. A, Ambedkar B (2018.) *Evolution of Enterprise Resource Planning*. Excel Journal of Engineering Technology and Management Science (An International Multidisciplinary Journal), Vol.1 N.11, December – January, 2017-2018. Preuzeto 4. ožujka 2018. s <https://fardapaper.ir/mohavaha/uploads/2017/10/Evolution-of-Enterprise-Resource-Planning.pdf> .
- [2] Al-Ghofaili, A. A., Al-Mashari, M. A. (2014). *ERP System Adoption - Traditional ERP system vs. Cloud-based ERP systems*. Fourth Edition of the International Conference on the Innovative Computing Technology (INTECH 2014), str. 39 - 50. doi: 10.1109/intech.2014.6927770. Pristupano 16.9.2018 na <https://ieeexplore.ieee.org/abstract/document/6927770/> .
- [3] Al-Mashari, M. (2003). A Process Change-oriented Model for ERP Application. *International Journal of Human-Computer Interaction*, 16(1) 39-55, doi: 10.1207/s15327590ijhc1601_4. Pristupano 11.9.2018 na https://www.tandfonline.com/doi/abs/10.1207/S15327590IJHC1601_4 .
- [4] Armbrust M., Fox A., Griffith R., Joseph A. D., Katz R., Konwinski A., Lee G., Patterson D., Rabkin A., Stoica I., Zaharia M. (2010.) *A View of Cloud Computing*. Published at Communication of the ACM in April 2010, Vol. 53 No.4 . Preuzeto 24. rujna 2017. s https://s3.amazonaws.com/academia.edu.documents/34578652/a_view_of_cc.pdf?AWSAccessKeyId=AKIAIWOWYYGZ2Y53UL3A&Expires=1506236815&Signature=mrmkBmEk86ahB69Y5XKJ1MPyto0%3D&response-content-disposition=inline%3B%20filename%3DA_View_of_Cloud_Computing_Clearing_the_c.pdf .
- [5] Brettel M., Friederichsen N., Keller M., Rosenberg M. (2014.) *How Virtualization, Decentralization and Network Building Change the Manufacturing Landscape: An Industry 4.0 Perspective*. World Academy of Science, Engineering and Technology, International Journal of Mechanical, Aerospace, Industrial, Mechatronic and Manufacturing Engineering Vol:8, No:1. Preuzeto 6. kolovoza 2017. s <http://www.waset.org/publications/9997144> .
- [6] Carlton, R. (2016). *Three ways you can integrate IoT into your ERP*. Pristupano 18.9.2018. na <https://www.erpfocus.com/three-ways-you-can-integrate-the-iot-into-your-erp-4448.html> .
- [7] Chen H., Chiang R. H. L., Storey V. (2012) *Business Intelligence and Analytics: From Big Data to Big Impact*. MIS Quarterly Vol. 36 No. 4 pp. 1165-1188/December 2012. Preuzeto 20. kolovoza 2017 s https://s3.amazonaws.com/academia.edu.documents/32970305/FROM_BIG_DATA_TO_BIG_IMPACT.pdf?AWSAccessKeyId=AKIAIWOWYYGZ2Y53UL3A&Expires=1503195051&Signature=4JPq3HhCybb4Qk0iKH1%2F4al0Mg%3D&response-content-disposition=inline%3B%20filename%3DSPECIAL_ISSUE_BUSINESS_INTELLIGENCE_RESE.pdf .
- [8] Chi H., Kang S., Wang X. (2013) *Research trends and opportunities of augmented reality applications in architecture, engineering, and construction*. Preuzeto 29. rujna 2017 s https://www.researchgate.net/profile/Shih_Chung_Kang/publication/257371819_Research_trends_and_opportunities_of_augmented_reality_applications_in_architecture_engineering_and_construction/links/53f458440cf2155be35502a3.pdf .

- [9] Chofreh, A. G., Goni, F. A. i Jofreh M. G.(2011). Enterprise Resource Planning (ERP) Implementation Process: Project Management Perspective. *Advanced Materials Research*, 338(2011), 152-155, doi: 10.4028/www.scientific.net/AMR.338.152. Pristupano 11.9.2018 na https://www.researchgate.net/publication/266908912_Enterprise_Resource_Planning_ERP_Implementation_Process_Project_Management_Perspective .
- [10] Conner B. P., Manogharan G. P., Martof A. N., Rodomsky L. M., Rodomsky C. M., Jordan D. C., Limperos J. W. (2014.) *Making sense of 3-D printing: Creating a map of additive manufacturing products and services*. Preuzeto 24. rujna 2017. sa https://www.researchgate.net/profile/Guha_Manogharan/publication/265388307_Making_sense_of_3-D_printing_Creating_a_map_of_additive_manufacturing_products_and_services/links/5596951e08ae99aa62c85794.pdf .
- [11] Erazo, J., Arboleda, H., Pino, F. J. (2017). Analysis of the Software Implementation Process for ERP Systems. *Advances in Computing*, 297 - 312. doi: 10.1007/978-3-319-66562-7_22. Pristupano 7.9.2018. na https://link.springer.com/chapter/10.1007%2F978-3-319-66562-7_22 .
- [12] Eremia M., Toma L., Sanduleac M. (2017.) *The Smart City Concept in the 21st Century*. 10th International Conference Interdisciplinary in Engineering, INTER-ENG 2016. Procedia Engineering 181 (2017) 12 – 19. Preuzeto 22. siječnja 2018. s https://ac.els-cdn.com/S1877705817309402/1-s2.0-S1877705817309402-main.pdf?_tid=53c9c116-ffc2-11e7-9371-00000aab0f26&acdnat=1516659763_7529806184cff4220cae5b26a2c45450 (<https://www.sciencedirect.com/science/article/pii/S1877705817309402#>).
- [13] ERP News (2018). *Is Your Business ERP Ready for Industry 4.0?* Pristupano 18.9.2018. na <http://www.erpnews.com/business-erp-ready-industry-4-0/> .
- [14] Guiora A. N. (2017.) *Cybersecurity – Geopolitics, Law and Policy*, First published 2017 by Routledge 711 Third Avenue, New York, NY 10017. Pregled knjige, stranice koje se koriste su bile dostupne na dan pristupa linku. Preuzeto 23. rujna 2017. s <https://books.google.hr/books?id=3LE0DgAAQBAJ&pg=PA18&dq=cybersecurity+definition&hl=hr&sa=X&ved=0ahUKEwiSi924w7TWAhVBWhoKHRyBAHAQ6AEIODAC#v=onepage&q=cybersecurity%20definition&f=false> .
- [15] Haller, S., Karnouskos, S. i Schroth, C. (2008). Internet of Things in an Enterprise Context. Pristupano 17.9.2018 na <https://www.alexandria.unisg.ch/46642/1/fis2008-haller-final.pdf> .
- [16] Haller S., Magerkurth C. (2011) *The Real-time Enterprise: IoT-enabled Business Processes*. Preuzeto 31. kolovoza 2018. s <http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.227.9881&rep=rep1&type=pdf> .
- [17] Harper R. (2003.) *Inside the Smart House*, Dostupno 6. siječnja 2018. na https://www.researchgate.net/profile/Sue_Smith25/publication/226566868_Emotional_Context_and_Significancies_of_Media/links/584dbe0308aed95c25032549.pdf#page=31 .
- [18] Huang, T., Yasuda, K. (2016). Reinventing ERP Life Cycle. *Journal of Enterprise Resource Planning Studies*, 2016(2016), doi: 10.5171/2016.331270. Pristupano 8.9.2018. na https://www.researchgate.net/profile/Tingting_Huang2/publication/312218523_Reinventin_g_ERP_Life_Cycle_Model_From_Go-Live_To-Withdrawal/links/5877264a08ae8fce492debdf/Reinventing-ERP-Life-Cycle-Model-From-Go-Live-To-Withdrawal.pdf .

- [19] Ijaz, A., Malik, R. K., Lodhi, R. N., Habiba, U., Irfan, S. M. (2014) *A Qualitative Study of the Critical Success Factors of ERP System - A Case Study Approach*, Proceedings of the 2014 International Conference on Industrial Engineering and Operations Management, January 7-9, 2014. Pristupano 7.9.2018. na <http://ieom.org/ieom2014/pdfs/547.pdf> .
- [20] Ijspeert A. J., Martinoli A., Billard A., Gambardella L. M. *Collaboration through the Exploitation of Local Interactions in Autonomous Collective Robotics: The Stick Pulling Experiment*. Istituto Dalle Molle di Studi sull'Intelligenza Artificiale (IDSIA), Lugano Laboratoire de Microinformatique (LAMI), EPFL, Lausanne. Preuzeto 24. kolovoza 2017. s https://infoscience.epfl.ch/record/28057/files/AR_01.pdf .
- [21] Jacobs R., Weston 'Ted' Jr. (2006.) *Enterprise resource planning (ERP) – A brief history*. Journal of Operations Management 25(2007) 357-363. Dostupno online od 12.12.2006 na www.sciencedirect.com (<https://www.sciencedirect.com/science/article/abs/pii/S0272696306001355> , Preuzeto 17.6.2018)
- [22] Jagoda, K., Samaranayake, P. (2017). An integrated framework for ERP system implementation, *International Journal of Accounting & Information Management*, 25(1), 91 - 109. doi: 10.1108/ijaim-04-2016-0038. Pristupano 7.9.2018. na <https://www.emeraldinsight.com/doi/abs/10.1108/IJAIM-04-2016-0038> .
- [23] Kowalke, P. (2018). *Six Tips for ERP Integration with IoT*. Pristupano 18.9.2018. na <https://it.toolbox.com/blogs/peterkowalke/six-tips-for-erp-integration-with-iot-010218> .
- [24] Metaxiotis, K., Zafeiropoulos, I., Nikolinakou, K., i Psarras, J. (2005). Goal directed project management methodology for the support of ERP implementation and optimal adaptation procedure. *Information Management & Computer Security*, 13(1), 55–71. doi:10.1108/09685220510582674 . Pristupano 11.9.2018. na <https://www.emeraldinsight.com/doi/abs/10.1108/09685220510582674> .
- [25] Meyer, S., Ruppen, A. i Magerkurth, C. (2013). *Internet of Things-aware Process Modeling: Integrating IoT Devices as Business Process Resources*. Pristupano 18.9.2018. na <https://pdfs.semanticscholar.org/a600/8e5f29bab2699302608273f3bfd5babf073a.pdf> .
- [26] Nagpal, S., Khatri, S. K. i Kumar A. (2015). *Comparative Study of ERP Implementation Strategies*. Pristupano 13.9.2018. na https://www.researchgate.net/profile/Dr_Sunil_Kumar_Khatri/publication/283561125_Comparative_Study_of_ERP_Implementation_Strategies/links/568a627d08ae1e63f1fbc208.pdf .
- [27] Ning H., Wang Z. (2011.) *Future Internet of Things Architecture: Like Mankind Neural System or Social Organization Framework?*. IEEE Communications Letters, Vol. 15, No. 4. April 2011. Preuzeto 27.12.2017. s <http://www.cybermatics.org/images/Future%20Internet%20of%20Things%20Architecture%20Like%20Mankind%20Neural%20System%20or%20Social%20Organization%20Framework.pdf> .
- [28] Olavsurd, T. (2017). *10 Internet of things success stories*. Pristupano 17.9.2018 na <https://www.cio.com/article/3229671/internet-of-things/10-internet-of-things-success-stories.html> .
- [29] O'Leary D. E. (2000). *Enterprise Resource Planning Systems*. New York, NY, USA: Cambridge University Press.

- [30] Pyatakova, S. (2018). *4 Real-Life IoT Examples Proven to transform business*. Pristupano 18.9.2018 na <https://www.itransition.com/blog/4-real-life-iot-success-cases-proven-to-transform-business> .
- [31] Raihana, G. F. H., College, J. M. (2012). CLOUD ERP - A SOLUTION MODEL. *IRACST - International Journal of Computer Science and Information Technology & Security*, 2(1), issn: 2249-9555. Pristupano 16.9.2018 na <http://esl.checkbeta.com/upload/whitepaper1.pdf> .
- [32] Rüßmann M., Lorenz M., Gerbert P., Waldner M., Justus J., Engel P., Harnisch M. (2015.) *Industry 4.0: The Future of Productivity and Growth in Manufacturing Industries*. Preuzeto 8. kolovoza 2017. s http://www.inovasyon.org/pdf/bcg.perspectives_Industry.4.0_2015.pdf.
- [33] Shatat, A. S. (2015). Critical Success Factors in Enterprise Resource Planning (ERP) System Implementation: An Exploratory Study in Oman. *The Electronic Journal of Information Systems Evaluation*, 15(1). Pristupano 7.9.2018 na <http://www.ejise.com/issue/download.html?idArticle=947> .
- [34] Stier, D. I. (2016). Digitalna ekonomija i hrvatsko gospodarstvo: Smjer prema gore? *OPEN InfoTrend: Informatika za gospodarstvo znanja*, 25(202), 16-17, dostupno 31.8.2018. na <http://www.infotrend.hr/files/pdf/casopis/2016/202/InfoTrend-202.pdf> .
- [35] Tallam A. (2013.) *Evolution of SAP (System, Applications and Products)*. Preuzeto 7. ožujka 2018. s <https://www.esds.co.in/blog/evolution-of-sap-system-applications-and-products/#sthash.KeCh8QeE.dpbs> .
- [36] TongKe F. (2013.) *Smart Agriculture Based on Cloud Computing and IOT* Journal of convergence Information Technology, Volume 8, Number 2, Jan 2013. Preuzeto 25. veljače 2018. s <https://pdfs.semanticscholar.org/62ee/b701c40626811a1111ca5d1db37650f1ea0b.pdf> .
- [37] Wei, CC., Chien, CF., Wang MJ. J. (2004). An AHP-based approach to ERP system selection. *International Journal of Production Economics*, 96(2005), 47-62, doi: 10.1016/j.ijpe.2004.03.004. Pristupano 10.9.2018 na <https://www.sciencedirect.com/science/article/pii/S0925527304000520> .
- [38] Xu L.D., He W., Li S. (2014) *Internet of Things in Industries: A Survey*, IEEE Transactions On Industrial Informatics, Vol. 10, No. 4, November 2014. Preuzeto 17. rujna 2017. s https://www.researchgate.net/profile/Wu_He2/publication/270742269_Internet_of_Things_in_Industries_A_Survey/links/55fc355a08aec948c4b189f6/Internet-of-Things-in-Industries-A-Survey.pdf .
- [39] Žitnik B. (2016), Digitalni rascjep: Na kojoj smo strani? *OPEN InfoTrend: Informatika za gospodarstvo znanja*. 25(202), 10-15, dostupno 31.8.2018 na <http://www.infotrend.hr/files/pdf/casopis/2016/202/InfoTrend-202.pdf> .