Primjena proširene stvarnosti u stvaranju dodane vrijednosti ERP sustavu

Bakale, Toni

Master's thesis / Diplomski rad

2023

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://urn.nsk.hr/urn:nbn:hr:211:571398

Rights / Prava: Attribution-NonCommercial-ShareAlike 3.0 Unported/Imenovanje-Nekomercijalno-Dijeli pod istim uvjetima 3.0

Download date / Datum preuzimanja: **2024-05-16**

SVEUČILIŠTE U ZAGREBU
FAKULTET ORGANIZACIJE I INFORMATIKE
VARAŽDIN

Repository / Repozitorij:

Faculty of Organization and Informatics - Digital Repository

zir.nsk.hr

DIGITALNI AKADEMSKI ARHIVI I REPOZITORIJI

SVEUČILIŠTE U ZAGREBU

FAKULTET ORGANIZACIJE I INFORMATIKE

V A R A Ž D I N

Toni Bakale

PRIMJENA PROŠIRENE STVARNOSTI U STVARANJU DODANE VRIJEDNOSTI ERP SUSTAVU

DIPLOMSKI RAD

Varaždin, 2023.

SVEUČILIŠTE U ZAGREBU

FAKULTET ORGANIZACIJE I INFORMATIKE

V A R A Ž D I N

Toni Bakale

Matični broj: 0016135426–R

Studij: Organizacija poslovnih sustava

PRIMJENA PROŠIRENE STVARNOSTI U STVARANJU DODANE VRIJEDNOSTI ERP SUSTAVU

DIPLOMSKI RAD

Mentor:

prof. dr. sc. Ruben Picek

Varaždin, kolovoz 2023.

Toni Bakale

Izjava o izvornosti

Izjavljujem da je moj završni/diplomski rad izvorni rezultat mojeg rada te da se u izradi istoga nisam koristio drugim izvorima osim onima koji su u njemu navedeni. Za izradu rada su korištene etički prikladne i prihvatljive metode i tehnike rada.

Autor potvrdio prihvaćanjem odredbi u sustavu FOI-radovi

_____

Sažetak

Tema ovog rada je primjena proširene stvarnosti u stvaranju dodane vrijednosti ERP sustavu. Kroz teorijsku obradu rad pokušava približiti moderne i inteligentne ERP sustave, industriju 4.0 i načine na koje se primjenom tehnologija industrije 4.0, pretežito proširene stvarnosti, može stvoriti dodana vrijednost. ERP sustav kojem se želi stvoriti dodana vrijednost je BORM Business ERP, sustav specijaliziran za drvoprerađivačku industriju. Stoga je za način stvaranja dodane vrijednosti odabran razvoj aplikacije za prikaz namještaja u stvarnom prostoru koristeći proširenu stvarnost. Praktični dio rada prikazuje proces razvoja aplikacije, od planiranja, dizajna, razvoja i testiranja. Na kraju rada je prikazan koncept implementacije aplikacije u BORM Business ERP sustav.

Ključne riječi: ERP sustavi; industrija 4.0; 3D modeli; računalna grafika; mobilna aplikacija; AR; Unity

Sadržaj

Sadržaj iii
1. Uvod 1
2. Metode i tehnike rada 2
3. Moderni ERP sustavi 3
3.1. ERP sustavi treće generacije 4
3.2. Utjecaj digitalne transformacije na ERP sustave 5
3.3. Inteligentni ERP sustavi 6
3.4. Industrija 4.0 u sklopu ERP sustava 7
3.4.1. Umjetna inteligencija i iERP 8
3.4.2. Internet stvari i iERP 9
3.5. Stvaranje dodane vrijednosti ERP sustavu 11
3.5.1. Analiza tržišta 11
3.5.2. Dodana vrijednost kroz tehnologije industrije 4.0 i digitalnu transformaciju 12
4. Proširena stvarnost 13
4.1. Pojedinosti tehnologije 13
4.1.1. Uređaji za prikaz proširene stvarnosti 16
4.1.2. Razvoj aplikacija proširene stvarnosti 19
4.2. Proširena stvarnost u poslovanju 20
4.2.1. Primjer proširene stvarnosti u školstvu i edukaciji u poslovanju 21
4.2.2. Primjer proširene stvarnosti u turizmu 22
4.3. Proširena stvarnost u ERP sustavima 23
4.3.1. Područja primjene 23
4.3.2. Izazovi i problemi 25
5. Integracija proširene stvarnosti u ERP sustav 26
5.1. Integracija u proizvodnji 26
5.2. Integracija u skladištu 27

6. Praktičan primjer razvoja mobilne aplikacije proširene stvarnosti za ERP sustav 29
6.1. BORM Business ERP 29
6.2. Pointline CAD 31
6.3. Mobilna aplikacija – *ARoom* 33
6.3.1. Metodika razvoja – Osobno ekstremno programiranje 33
6.3.2. Planiranje 35
6.3.3. Razvojno okruženje – *Unity* 36
6.3.4. Dizajn 37
6.3.5. Implementacija 41
6.3.6. Testiranje funkcionalnosti aplikacije, problemi i izazovi tokom razvoja 57
6.3.6.1. Testiranje detekcije ploha i stabilnosti modela 58
6.3.6.2. Testiranje manipulacije modelima 58
6.3.6.3. Testiranje korisničkog sučelja 61
7. Koncept integracije mobilne aplikacije u ERP sustav 63
7.1. Arhitektura sustava 63
7.2. Problemi integracije 65
7.3. Monetizacija 66
7.4. Mjerenje uspješnosti 66
7.5. Moguće nadogradnje aplikacije 67
7.6. Mogućnost integracije u druge ERP sustave 68
8. Zaključak 69
Popis literature 70
Popis slika 74

1. Uvod

ERP sustavi su postali ključan dio svakodnevnog poslovanja i njihova primjena je sve prisutnija. Od velikih poduzeća koja ih koriste već desetljećima, do malih koja ih, neovisno o svojoj veličini, počinju koristiti u sve većem broju, dijelom zbog sve veće pristupačnosti, ali i potrebe, u današnjem svijetu su postali jedan od, ako ne i najvažniji resurs u poslovanju. Kvaliteta podataka i njihovo maksimalno iskorištavanje mogu biti razlika između uspjeha i neuspjeha poduzeća, stoga je bitno što bolje njima gospodariti te tu ulaze ERP sustavi. ERP sustavi pokušavaju objediniti sve procese i podatke unutar organizacije. Osnovni cilj im je svima isti, ali ono što ih razlikuje su dodatne mogućnosti koje nude. Od funkcionalnosti koje se bez njih ne bi mogle raditi do funkcionalnosti koje i njima, i poduzeću stvaraju dodanu vrijednost.

Ideja ovog rada je prikazivanje načina na koji se ERP sustavima može stvoriti dodana vrijednost, konkretno kroz primjer mobilne aplikacije za prikaz 3D modela namještaja koristeći proširenu stvarnost. Uzevši u obzir ranije navedeno, rad se sastoji od dva dijela: teorijskog i praktičnog. Teorijski dio će uključivati pojašnjavanje koncepta modernog i inteligentnog ERP sustava, značajke i specifičnosti sustava, tehnologije industrije 4.0, njihov opis sa naglaskom na proširenu stvarnost (AR), kako njima stvoriti dodanu vrijednost ERP sustava, i koncept integracije AR aplikacije u ERP sustav. Praktični dio uključivat će detaljan opis izrade aplikacije.

2. Metode i tehnike rada

Metode i tehnike rada korištene u ovom radu se mogu podijeliti u dvije kategorije: metode i tehnike korištene za teoretsku obradu, te metode i tehnike korištene za izradu mobilne aplikacije.

Za teorijsku obradu je korištena stručna literatura, prikupljena iz knjiga i stručnih članaka, te pouzdanih web stranica vezanih uz ERP sustave, moderne i inteligentne ERP sustave, industriju 4.0 i proširenu stvarnost. Također, dio teorijske obrade je vezan uz analizu tržišta. Većina tih informacija pronađena je na javno dostupnim izvorima najvećih proizvođača ERP sustava. Informacije vezane uz BORM-ove aplikacije su preuzete sa službenih stranica poduzeća i autorovog iskustva rada tijekom studentske prakse.

U praktičnom dijelu rada je korišten alat *Unity* za izradu programskog rješenja, *GitHub* i *GitHub* Desktop za verzioniranje kôda, te različite službene dokumentacije vezane uz *Unity*, *ARCore*, C# programski jezik, i slično. Također su korišteni alati poput *Figme*, za dizajniranje korisničkog sučelja, i *draw.io*, za izradu dijagrama. Metodika korištena u razvoju aplikacije je *personal extreme programming - PXP*.

3. Moderni ERP sustavi

ERP sustavi (eng. *Enterprise Resource Planing systems*) postoje već desetljećima i prolazili su kroz razne promjene, ali im je glavni cilj ostao isti - „…povezati sve poslovne procese organizacije u jedan informacijski sustav koji radi s centraliziranom bazom podataka u koju se spremaju podaci iz različitih modula/poslovnih funkcija.“ (Picek, R. ,2023). Moderni ERP sustav podrazumijeva sustav koji je riješio sljedeće probleme:

1. Zastarjela tehnologija – ERP sustav nije moderan ako koristi zastarjele tehnologije, ne iskorištava maksimalno dostupan hardver i softver za unaprjeđenje
2. Ograničen broj funkcionalnosti – ERP sustav nije moderan ako ne zadovoljava rastuće funkcionalne potrebe organizacija
3. Loše korisničko iskustvo – ERP sustav nije moderan ako koristi korisničko sučelje koje je zastarjelo ili teško za koristiti
4. Loša integracija – ERP sustav nije moderan ako ima problema sa integracijom drugih sustava
5. Slabosti sigurnosti podataka – ERP sustav nije moderan ako konstantno ne unaprjeđuje sigurnost podataka, česti razlog hakiranja sustava organizacija je preko zastarjelih, takozvanih *legacy* aplikacija
6. Poslovni izazovi – zastarjeli ERP sustavi narušavaju produktivnost vještih IT zaposlenika koji bi trebali stvarati vrijednosti, a umjesto toga su zaduženi za održavanje sustava
7. Troškovi održavanja – zastarjeli ERP sustavi često imaju velike troškove održavanja, dijelom zbog hardvera, a dijelom zbog manjka stručnjaka koji mogu rješavati probleme u zastarjelim sustavima

Sustav koji je riješio gore navedene probleme još možemo nazvati i ERP sustavom treće generacije. Takav sustav radom u oblaku, fokusom na kupce, informacijskom sigurnosti, korisničkim iskustvom i korištenjem raznih novih tehnologija stvara tako zvanu „organizaciju bez granica“. (Rawat, C. 2023.)

3.1. ERP sustavi treće generacije

Jedna od glavnih značajnosti treće generacije ERP sustava je migracija prema arhitekturi u oblaku (eng. *Cloud*). Za razliku od tradicionalnih, lokalnih (eng. *on-premise*), sustava koji su često zahtijevali značajna ulaganja u hardver i infrastrukturu, moderni ERP sustav koristi oblak kako bi organizaciji pružao neusporedivu skalabilnost, fleksibilnost i cjenovnu dostupnost. Te značajke su učinile ERP sustave pristupačnijima manjim organizacijama jer je oblak riješio česte probleme koji su vezali najmoćnije mogućnosti ERP sustava uz velike infrastrukturne troškove. Fleksibilnost je prikazana kroz veliku razinu konfiguracije ERP sustava. Naime, u prošlosti su se primarno pojavljivale dvije vrste ERP sustava: gotovi ERP sustav kojem se organizacija treba prilagoditi, i ERP sustav „po narudžbi" koji se radi za organizaciju, ali je cjenovno nepristupačan. Danas se dobar dio ERP sustava nalazi na pola puta, gdje postoji gotov sustav ali ga je moguće do neke razine prilagoditi potrebama organizacije, što je omogućeno modularnim razvojem ERP sustava. Kako god bilo, određena razina reinženjeringa poslovnih procesa je i dalje neizbježna, ali je cijeli proces ovime ubrzan i olakšan. Međutim, uz sve te prednosti, neki ERP sustavi su i dalje dijelom, ili čak i u potpunosti lokalni. Jedan od tih razloga je sigurnost. Oblak je puno sigurniji nego lokalna implementacija, ali ljudi vole imati stvari „pod svojim krovom", misleći da su na taj način sigurniji, te se javlja sukob ljudske percepcije sigurnosti i realne sigurnosti okruženja u oblaku. Usprkos tome sve više organizacija ipak prelazi na okruženje u oblaku. (Tomasz, T., Damian, D. , 2018.)

Osim okruženja u oblaku, veliki je naglasak na nove tehnologije u modernim ERP sustavima, te sustave možemo zvati i inteligentnim ERP sustavima (iERP). Najviše je naglaska na umjetnoj inteligenciji (*AI – aritificial inteligence*), koja je „eksplodirala" u proteklih godinu dana, i za vrijeme pisanja ovog rada postala glavna poštapalica (eng. *buzzword*) u IT-u i iz dana u dan se sve brže razvija. Tehnologije koje također imaju veliki naglasak su strojno učenje (eng. *ML - machine learing*) i Internet stvari (eng. *IoT - Internet of Things*), ali i mnoge druge, poput proširene stvarnosti (eng. *AR - augmented reality*). (Tomasz, T., Damian, D. , 2018. ; Picek, R., & Androcec, D. 2022.) O njima će biti govora više u sljedećim poglavljima.

Veliki naglasak u razvoju modernih ERP sustava je i na korisničkom iskustvu (eng. *UX - user experience*). Dizajn korisničkog iskustva je usredotočen na intuitivnost, odzivnost i mogućnost prilagodbe kako bi se što više pojednostavilo korištenje ERP sustava korisnicima svih razina unutar organizacije. Unaprijeđeno korisničko iskustvo ubrzava vrijeme prilagodbe kod integracije sustava, obuku novih zaposlenika i povećava efikasnost. Na korisničko iskustvo se nadovezuje i mobilni pristup ERP sustavu, tj. pristup sa različitih uređaja, a okruženje u

oblaku to uvelike olakšava, te uz razvoj višeplatformskih aplikacija koje omogućuju pristup svim sustavima bilo gdje i bilo kada. Na taj način se ključnim informacijama može pristupiti u najkraćem mogućem periodu i odluke se mogu donositi brzo, i time ubrzati korisnička podrška. (Tomasz, T., Damian, D. , 2018. ; Picek, R., & Androcec, D. 2022.)

Treća generacija ERP sustava predstavlja značajan iskorak u preoblikovanju načina na koji organizacije posluju i upravljaju svojim resursima. Kroz sve ranije navedene prednosti moderni ERP sustavi obogaćuju organizacije i dižu ih na višu razinu, te im omogućuju ostati kompetitivnima na tržištu. Moderni ERP sustavi su osmišljeni sa konstantnim unaprjeđivanjem sustava, a time i organizacije kroz modularni pristup razvoju. (Tomasz, T., Damian, D. , 2018.)

3.2. Utjecaj digitalne transformacije na ERP sustave

Organizacije se kontinuirano trebaju prilagođavati novim tehnologijama i trendovima digitalnog doba kako bi ostali kompetitivni i relevantni na tržištu. Iz toga razloga digitalna transformacija je postala glavni prioritet organizacija svih industrija. Žele iskoristiti digitalne tehnologije kako bi unaprijedile svoje procese, povećale efikasnost i ostale korak ispred konkurencije. (Rawat, C. 2023.)

ERP sustavi mogu biti ključni u omogućavanju digitalne transformacije pružajući organizaciji jedinstvenu, integriranu platformu za upravljanje ključnim poslovnim procesima. ERP sustavi su uvijek bili prisutni u velikim poduzećima. Digitalna transformacija je „otvorila oči“ i manjim poduzećima i povećala potražnju za ERP sustavima. Pokazuje važnost digitalizacije, informatizacije i automatizacije poslovanja, a ERP sustavi to omogućuju na unificiran način.

Postoje razni razlozi zašto organizacije odlučuju implementirati ERP sustave. Najčešći razlog je potreba za unificiranjem svi informacijskih sustava organizacije u jedno rješenje koje će pokriti sve poslovne funkcije organizacije. Budući da ERP sustavi pružaju podršku cijelog sustava logično je zaključiti da su ERP sustavi i digitalna transformacija usko povezani. ERP sustav je zapravo platforma na kojoj bi rezultati implementacije digitalne transformacije trebali biti vidljivi. Čak i kad ti rezultati nisu odmah vidljivi stvaraju prostor za jednostavniju buduću nadogradnju. (Špatenka, J., & Koch, M. 2021)

Razvoj novih tehnologije uvelike utječe na ERP sustave i logično je zaključiti da će biti implementirani u novim verzijama ERP sustava. Rad u oblaku, umjetna inteligencija, mobilne aplikacije, internet stvari, prepoznavanje glasa, robotska automatizacija procesa, strojno

učenje, prediktivna analitika i mnoge druge tehnologije postaju dio ERP sustava. Nove generacije ERP sustava mogu premostiti jaz između novih tehnologija i poslovnih zahtjeva. Novi, moderni ERP sustavi su logičan izbor kod implementacije digitalne transformacije. Najvažnija stvar za kretanje u digitalnu transformaciju su ERP sustavi u oblaku. ERP sustavi u oblaku su postali pokretači digitalne transformacije i donose sa sobom niz prednosti i novih funkcionalnosti u usporedbi sa tradicionalnim lokalnim ERP sustavima. Prelazak na ERP sustave u oblaku više nije rijetkost budući da takva rješenja pomažu u smanjivanju troškova razvoja, ubrzavaju implementaciju i eliminiraju potrebu za internim softverom i hardverom. Uz to stvaraju i platformu za razne moguće nadogradnje i implementacije komplementarnih tehnologija, poput tehnologija industrije 4.0. (Špatenka, J., & Koch, M. 2021; Picek, R., & Androcec, D. 2022, Rawat, C. 2023)

ERP sustavi su izrasli iz statičnih i velikih sustava koji su se fokusirali na poslovne funkcije i bili su jako dobro organizirani u optimizaciji podataka i poslovnih procesa u strateški važne dijelove organizacije.

3.3. Inteligentni ERP sustavi

Sve više i više organizacija prelazi iz lokalnih ERP sustava u ERP sustave u oblaku jer su ERP sustavi u oblaku tehnološki napredniji i ekonomski isplativiji. Inteligentni ERP sustavi, takozvani iERP sustavi, su nova generacija ERP sustava u oblaku koja je nastala iz utjecaja digitalne transformacije. Uz novu generaciju ERP sustava u oblaku su integrirane i nove tehnologije digitalne transformacije. Najvažnije od tih tehnologija su one koje svoje korijene vuku iz industrije 4.0: internet stvari, umjetna inteligencija, strojno učenje, virtualna i proširena stvarnost, tehnologije lanca blokova (eng. *blockchain*) i velikih podataka (eng. *big data*). Osim tih tehnologija ključna je i njihova međusobna suradnja. Internet stvari prikuplja podatke, velikim podatcima se analiziraju, strojnim učenjem se iz tih podataka izvlače zaključci koji se prikazuju pomoću proširene stvarnosti korisniku ERP sustava kako bi lakše donosio odluke. (Jayender, P., & Kundu, G. K. 2021; Picek, R., & Androcec, D. 2022, Jenab, K., Staub, S., Moslehpour, S., & Wu, C. 2019)

Tehnologije industrije 4.0 pružaju poduzećima mogućnosti poboljšanja svojih poslovanja i razvoja novih proizvoda i usluga u sklopu digitalne ekonomije. Inteligentni ERP sustavi u kontekstu poduzeća u proizvodnoj industriji su temelj za „pametne tvornice.“ Zahvaljujući tehnologijama digitalne transformacije ERP sustavi su transformirani u fleksibilne i pametne sustave prikladne za današnja moderna poduzeća. Inteligentni sustavi su

karakterizirani kroz tri dimenzije: korisničko iskustvo, proces i podatke. Podatci, prikupljeni iz različitih povezanih uređaja i analizirani pomoću algoritama su ključan dio inteligentnog ERP sustava i direktno su vezani uz procese. Današnji klijenti više vole inteligentne ERP sustave koji pružaju kombinaciju tih triju dimenzija u dostavljenim proizvodima i uslugama koje se mogu konfigurirati ovisno o trenutnim učenjima i zatim primijeniti ta učenja kako bi poboljšali procese i kvalitetu korisničkog iskustva. Nova generacija ERP sustava donosi više mogućnosti za povećanje poslovanja. Inteligentni ERP sustavi pružaju mogućnosti za integraciju i/ili optimizaciju poslovnih procesa, integraciju drugačijih aktivnosti koje stvaraju nadzornu ploču poslovanja u stvarnom vremenu i pronalaženje pravih odluka pomoću automatizacije donošenja odluka kroz analizu prilika i alternativa. (Jayender, P., & Kundu, G. K. 2021; Picek, R., & Androcec, D. 2022; Jenab, K., Staub, S., Moslehpour, S., & Wu, C. 2019)

Velika količina podataka koju inteligentni ERP sustavi generiraju treba biti obrađena. gore navedeni algoritmi su širok pojam i kombinirani sa nekim drugim tehnologijama industrije 4.0 mogu se definirati kao višeagentni sustavi obrade podataka. Agenti u kontekstu informacijskih znanosti predstavljaju računalne programe koji izvršavaju razne aktivnosti kontinuirano i samostalnu u ime nekog pojedinca unutar organizacije. U kontekstu inteligentnih ERP sustava to je ponajviše za obradu podataka, ali i za njihovo rudarenje, što iz velike količine prikupljenih podataka što iz javno dostupnih izvora. Rad u oblaku to maksimalno olakšava i smanjuje hardverske troškove na minimum. (Bouchemal, N., & Bouchemal, N. 2019; Sathiyamoorthi, V., & Bhaskaran, V. M. 2009)

ERP sustavi su postali ključni dijelovi gotovo svake uspješne organizacije za upravljanje kompleksnim poslovnim procesima. I oni i organizacije se međusobno unaprjeđuju, poboljšavaju i guraju jedni druge u napredak. Time se stvaraju inteligentni ERP sustavi i „organizacije bez granica“ u kojima se vide pravi potencijali rada u oblaku u kombinaciji sa novim tehnologijama poput tehnologija industrije 4.0. (Amini, M., & Abukari, A. M. 2020)

3.4. Industrija 4.0 u sklopu ERP sustava

ERP sustavi su postali ključan dio svake industrije i stoga je bitno objasniti odnos njih i industrije 4.0, kako su povezani i kako utječu jedno na drugo, ranije potpoglavlje je detaljno opisalo inteligentne ERP sustave. Ono što čini inteligentne ERP sustave inteligentnima su tehnologije industrije 4.0 pa će ovo potpoglavlje detaljnije objasniti njezine tehnologije i kako su primijenjene u sklopu inteligentnih ERP sustava.

Prvo je potrebno definirati industriju 4.0. Najlakša i najbanalnija definicija je: industrija 4.0 je četvrta industrijska revolucija. Pojam se primarno koristi za trendove u proizvodnji, ali ti trendovi nisu nužno vezani samo uz proizvodnju. Industriju 4.0 obilježavaju integracija digitalnih tehnologija, razmjena podataka i automatizacija u procesima proizvodnje. Cilj joj je stvoriti pametne tvornice (eng. *Smart factories*) i međusobno povezati sustave gdje strojevi, procesi i podatci besprijekorno komuniciraju i međusobno se unaprjeđuju. Već iz tog dijela se može vidjeti poveznica industrije 4.0 i inteligentnih ERP sustava. Inteligentni ERP sustavi omogućuju povezivanje procesa i podataka, a industrija 4.0 svojim tehnologijama nadograđuje i optimizira strojeve, te međusobno čine organizaciju još boljom i povezanijom. (sap.com, 2023)

Tehnologije industrije 4.0 su ključne tehnologije koje omogućuju povezanost u proizvodnji, ali one ne moraju biti nužno vezane samo za proizvodnju. Njihova je primjena izuzetno opširna i može se koristiti u raznim dijelovima ERP sustava, ne samo onima vezanim uz proizvodnju. (sap.com, 2023)

U tehnologije industrije 4.0 se mogu svrstati sve tehnologije čija primjena ima potencijal donijeti znatne i inovativne promjene. Njihovo definiranje i izdvajanje uvelike ovisi o industriji o kojoj se govori. Budući da se ovaj rad primarno odnosi na ERP sustave i industrije vezane uz prodaju i proizvodnju, tehnologije o kojima će biti govora u radu su umjetna inteligencija i strojno učenje, internet stvari i proširena stvarnost. Te tehnologije su ključne u svim industrijama i njihova primjena može obogatiti bilo koji sustav. Umjetna inteligencija, strojno učenje i Internet stvari će biti objašnjeni u sljedećim potpoglavljima, a proširena stvarnost će kasnije imati vlastito poglavlje jer je ona najvažnija za temu ovog rada.

3.4.1. Umjetna inteligencija i iERP

Umjetna inteligencija je naziv sa simulaciju ljudske inteligencije u strojevima kako bi izvršavala zadatke za koje je potreban ljudski razum. Strojno učenje je podskup umjetne inteligencije koji služi za učenje računala iz prikupljenih podataka i njihovo unaprjeđivanje bez direktnog programiranja. Ukratko objašnjeno - računala uče sama. (Tomasz, T., Damian, D. 2018.)

Jedna od najdominantnijih tehnologija u modernim ERP sustavima koja kreira brojne mogućnosti poslovnim procesima optimizacije je umjetna inteligencija. Umjetnu inteligenciju je

moguće svrstati u četiri kategorije: onu koja misli kao čovjek, onu koja se ponaša kao čovjek, onu koja razmišlja racionalno i onu koja se ponaša racionalno. Korištenje umjetne inteligencije u ERP sustavima ovisi o razini „inteligencije“ koja je u direktnoj korelaciji sa važnim resursima čiji troškovi mogu postati veći od troškova ERP sustava. Budući da postoji više dijelova ERP sustava gdje je moguće stvoriti dodanu vrijednost potrebno ih je gledati zasebno. Neki dijelovi potencijalnog korištenja gdje je moguće kreirati znatnu dodanu vrijednosti su optimizacija poslovnih procesa, donošenje optimalnih i polu optimalnih odluka (savjetovanje), dobivanje uvida u ljudske želje i vrijednosti, raspoznavanje slika i prepoznavanje anomalija. Taj potencijal je u rukama uprave koja treba donijeti odluku da se počinju primjenjivati u svakodnevnim aktivnostima. Dva su pristupa u implementaciji umjetne inteligencije u ERP sustave. U prvom umjetna inteligencija donosi odluke i zatim predlaže korake za implementaciju, a u drugom umjetna inteligencija predlaže različita rješenja problema, ali konačnu odluku donosi čovjek. U kontekstu implementacije umjetne inteligencije u ERP sustav drugi pristup je preferiran jer automatizira posao menadžmenta i u isto vrijeme mu omogućuje lakše donošenje odluka. Druga velika primjene umjetne inteligencije u ERP sustavima je za pronalazak „uskih grla“, što u softverskom smislu, što u organizacijskom. U softverskom smislu omogućava brži rad računala, može prepoznati na prvi pogledat neznatne razloge usporavanja, ali u globalu to može značiti puno. S druge strane u organizacijskom smislu može služiti poboljšanju produktivnosti i raspodjeli organizacijskih resursa. (Akturk, C. 2021; Picek, R., & Androcec, D. 2022)

Još neke primjene umjetne inteligencije u sklopu inteligentnih ERP sustava osim donošenja odluka su u: prepoznavanju logičkih grešaka, npr. u ispunjavanju raznih formi unutar ERP sustava i predviđanju potražnje analizom prikupljenih podataka. Umjetna inteligencija i strojno učenje iz perspektive industrije 4.0 i inteligentnih ERP sustava pomažu u svim poslovnim procesima organizacije, a ponajviše u procesima proizvodnje, kod predviđanja servisiranja strojeva, koristeći velik broj raznih podataka prikupljenih od drugih strojeva, i od stroja kojeg se prati pomoću senzora. Ti senzori su dio druge nove tehnologije, interneta stvari, o kojoj će biti više govora u sljedećem poglavlju. (Tomasz, T., Damian, D. , 2018. ; Yathiraju, N. 2022.)

3.4.2. Internet stvari i iERP

Internet stvari je najjednostavnije definirati kao mrežu (eng. *net*) međusobno povezanih uređaja/stvari (eng. *things*) unutar zatvorenog koncepta (eng. *inter*). Na to se primarno odnose senzori koji imaju ugrađene čipove za prikupljanje podataka i slanje istih drugima, a to su

uglavnom jednostavni uređaji koji imaju po jedan zadatak, npr. mjeriti temperaturu ili pritisak u nekom pogonu za vrijeme proizvodnje. Sami po sebi nisu kompleksni, ali njihova povezanost i kombinacija omogućuju automatizaciju i poboljšanje raznih zadataka. (techtarget.com, 2023)

Internet stvari je tehnologija sa velikim potencijalom za stvaranje dodane vrijednosti u ERP sustavima. Internet stvari „otvara vrata“ operativnoj izvrsnosti, automatizaciji u procesima gdje je moguća. Kao što je ranije navedeno glavna karakteristika interneta stvari je prikupljanje podatak u stvarnom vremenu od različitih izvora internetom u cijeloj organizaciji. Iz tih podataka se stvaraju analize i planovi koji mogu služiti kao baza za operativne i/ili strateške odluke. Potencijal interneta stvari je prepoznat od strane proizvođača ERP sustava i postaje standardna funkcionalnost ERP sustava. Najdominantniji razlog integracije interneta stvari unutar ERP sustava su mogućnosti prikupljanja značajnih podataka unutar cijele organizacije kako bi se mogli vizualizirati i kako bi se uređaji interneta stvari mogli upravljati preko ERP sustava. Poduzeća su svjesna o poboljšanjima koje internet stvari nudi poput: pametne analize, bolje sigurnosti, povećane produktivnosti, pametnog inventara, sigurnijeg putovanja i pregleda potražnje u stvarnom vremenu. Integracija interneta stvari i ERP sustava postavlja pitanja vezana uz kvalitetu signala uređaja, preciznosti podataka, kompleksnost i sigurnost. Važan pojam uz Internet stvari su veliki podatci jer kao što je ranije navedeno internet stvari prikuplja iznimno veliku količinu podataka koja se zapravo može klasificirati kao veliki podatci. Najjednostavnija definicija velikih podataka ogromna količina setova podataka kako bi se mogla analizirati i pokazati trendove. (Jayender, P., & Kundu, G. K. 2022 ;Picek, R., & Androcec, D. 2022)

Uređaji interneta stvari najveću ulogu imaju u automatizaciji. Nema potrebe za ručnim kontrolama strojeva jer imaju razne senzore koji konstantno prate rad sustava i javljaju u slučaju da se nekakav problem dogodi. Što se tiče ERP sustava internet stvari omogućava prikupljanje što većeg broja podataka koji se onda u ERP sustavu obrađuju. Ti podatci u kombinaciji sa ranije objašnjenom umjetnom inteligencijom omogućuju detaljne analize i predviđanja. (techtarget.com, 2023; Paththinige, P., Thilakarathne, K., Rathnasekara, T., Wickramaarachchi, R., & Withanaarachchi, A. 2022.)

Sve u svemu, Internet stvari kao dio industrije 4.0 i inteligentnih ERP sustava omogućuje neusporedivu povezanost i razmjenu podataka između uređaja i sustava u stvarnom vremenu, a kad se koristi u kombinaciji sa drugim novim tehnologijama, poput umjetne inteligencije, stvarno pokazuje svoje potencijale.

3.5. Stvaranje dodane vrijednosti ERP sustavu

Tržište ERP sustava je vrlo dinamično i kompetitivno, i potražnja je velika. Uz nju velika je i ponuda, a većinu tržišta zauzimaju velike tvrtke poput *SAP*-a, *Oracle*-a i *Microsoft-a*. Točne najnovije brojke udjela tržišta teško je odrediti i „zaključane“ su iza analiza poput *Gartner report*-a, ali je moguće sa velikom sigurnošću reći da zauzimaju više od polovice sveukupnog tržišta. Ostatak je podijeljen na manje proizvođače i tu svaka vrsta specijalizacije i diverzifikacije postaje ključna u njihovom opstojanju. Potrebno je stvoriti dodanu vrijednost koji ti ERP sustavi mogu pružiti korisniku kako bi ih odabrao umjesto konkurencije.

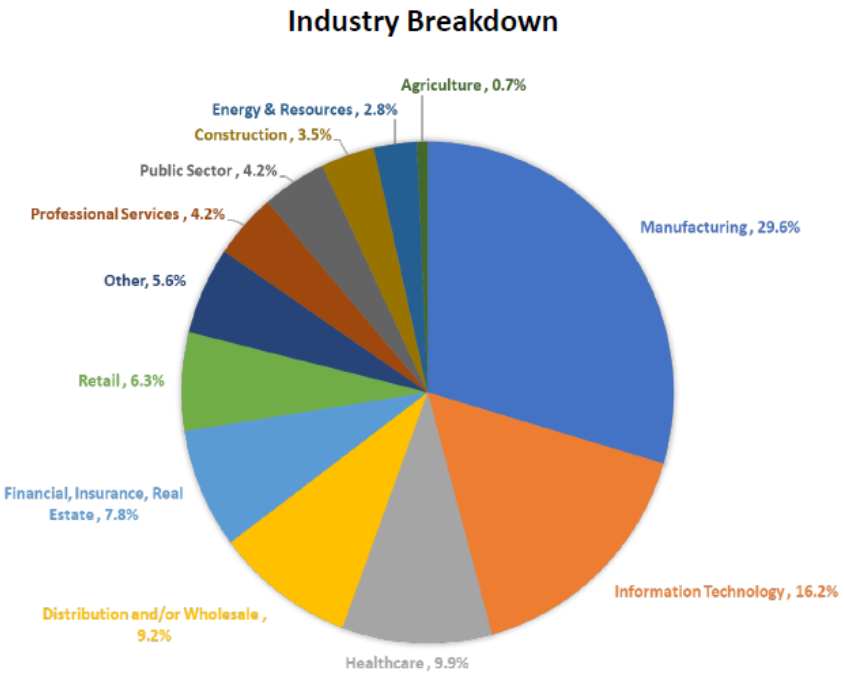
3.5.1. Analiza tržišta

Kako bi se manji proizvođač ERP sustava mogao diversificirati i stvoriti dodanu vrijednost kupcu potrebno je analizirati tržište, ali i najveće igrače, također zvane titanima. Dobra analiza titana i tržišta je „*2023 Clash of the Titans SAP vs. Oracle vs. Microsoft vs. Infor*“ *Panorama Consulting* grupe.

Izvještaj uspoređuje najveće proizvođače ERP sustava, ali i pruža informacije o tržištu. Važno je istaknuti da su najveći proizvođači ERP sustava fokusirani na velike, globalne organizacije kao klijente. Naravno, imaju i verzije ERP sustava za manja i srednja poduzeća, ali im je glavni fokus na tim većim poduzećima. Tu se može izdvojiti prvi način diversifikacije - fokus na srednja i manja poduzeća. Većina velikih poduzeća vjerojatno već koristi neki od gore navedenih titana kao svoj ERP sustav, te je stoga najbolje manjem proizvođaču ERP sustava usredotočiti se na mala i srednja poduzeća. (Panorama Consulting Group 2023.)

Analizom izvještaja mogu se izdvojiti zajedničke osobine titana, kao što su rad u oblaku i hibridna okruženja, upravljanje odnosa sa klijentima i korisničko iskustvo, upravljanje financijama, upravljanje opskrbnim lancem, upravljanje logistikom nabave, te prodaja i marketing. Iz toga je važno zaključiti da bi, barem do neke razine, svaki ERP sustav trebao imati te module u sebi. (Panorama Consulting Group 2023.)

Sljedeća stavka je tržište. Na slici 1 je prikazana podjela tržišta raznih industrija ovisno o njihovoj veličini.



Slika 1: Podjela tržišta raznih industrija ovisno o njihovoj veličini
(Izvor: 2023 Clash of the Titans SAP vs. Oracle vs. Microsoft vs. Infor)

Najveće tržište je tržište prerađivačke industrije. Prerađivačka industrija je skup mnogo različitih specijaliziranih industrija, ali dobar dio proizvodnje ima dosta sličnosti koje se mogu univerzalno primijeniti pa je fokus na neku nišu prerađivačke industrije dobra strategija diversifikacije. (Panorama Consulting Group 2023.) U kontekstu ovog rada i ERP sustava koji će biti korišten kao primjer, ta niša biti će drvoprerađivačka industrija.

3.5.2. Dodana vrijednost kroz tehnologije industrije 4.0 i digitalnu transformaciju

ERP sustav sa fokusom na mala i srednja poduzeća, te specijaliziranim tržišnim nišama drvoprerađivačke industrije već se poprilično izdvojio od titana i atraktivan je poduzećima drvoprerađivačke industrije. Možda je to u nekim slučajevima dovoljno, ali što ako postoji konkurencija koja radi isto to? Kako uvjeriti kupca da odabere vas, a kako zadržati kupca? Odgovor na sva navedena pitanja je: stvaranjem dodane vrijednosti sustavu.

Dodanu vrijednost se može definirati kao povećanje vrijednosti, korisnosti i kvalitete proizvoda, pojedinosti izvan osnovnog paketa koje proizvod diversificiraju od konkurencije i „primamljivost“ krajnjim korisnicima. (countingup.com 2023.)

Budući da govorimo o ERP sustavu koji se specijalizira za drvoprerađivačku industriju, odličan način stvaranja dodane vrijednosti je implementacija tehnologija industrije 4.0 u ERP sustav, specifično onih koje su objašnjene ranije u radu. To su tehnologije koje se sve više i više počinju primjenjivati, pa ih titani uvode i primjenjuju u svojim ERP sustavima. Međutim, manjim proizvođačima ERP sustava ipak treba više vremena da počnu primjenjivati te tehnologije. Tu se javlja prilika za stvaranje dodane vrijednosti prije nego ju konkurencija razvije i prilika za postavljanje svojeg ERP sustava kao lidera u primjeni ovih tehnologija.

Sljedeće poglavlje će prikazivati kako jednu od tih tehnologija, proširenu stvarnost. Biti će detaljno objašnjena sama po sebi, objašnjavat će se njezina primjena i načini na koji se može integrirati u ERP sustave. Nakon toga će u se u praktičnom primjeru izraditi mobilna aplikacija proširene stvarnosti koja se može primijeniti na ERP sustavu koji je specijaliziran za drvoprerađivačku industriju, te mu stvoriti dodanu vrijednost.

4. Proširena stvarnost

Proširenu stvarnost je moguće definirati kao tehnologiju koja prikazuje digitalne informacije poput 2D ili 3D virtualnih objekata u stvarnom svijetu u stvarnom vremenu, i na taj način proširuje korisnikovu stvarnost. U nekim aspektima je slična virtualnoj stvarnosti (eng. *VR – virtual reality*), ali za razliku od nje koja je u potpunosti virtualna, stvara mješavinu digitalnog i stvarnog svijeta. (Carmigniani J, Furht B, Anisetti M, Ceravolo P, Damiani E, Ivkovic M, 2011.)

4.1. Pojedinosti tehnologije

Ovo potpoglavlje će objasniti kako proširena stvarnost radi, prvobitno u obliku procesa u dva koraka, a kasnije će svaki od tih koraka biti detaljnije opisan, što ti koraci uključuju i koje tehnologije se mogu koristiti za te korake. (Craig, A. B. 2013)

Dva glavna koraka koja se trebaju dogoditi u svakom vremenskom intervalu aplikacije proširene stvarnosti su:

1. Aplikacija treba odrediti trenutno stanje fizičkog svijeta i treba odrediti trenutno stanje virtualnog svijeta

2. Aplikacija treba prikazati virtualni svijet u odnosu na fizički tako da ljudima stvore dojam da su virtualni elementi dio njihovog fizičkog svijeta i zatim se ponovno treba vratiti u prvi korak u sljedećem vremenskom intervalu

Naravno, postoji niz podkoraka u svakom od tih koraka, ali u srži se vrte ta dva ključna koraka. Kada se uzmu u obzir ta dva koraka moguće je vidjeti niz različitih metoda i tehnologija koje se mogu koristiti za njihovo postizanje i implementaciju. (Craig, A. B. 2013)

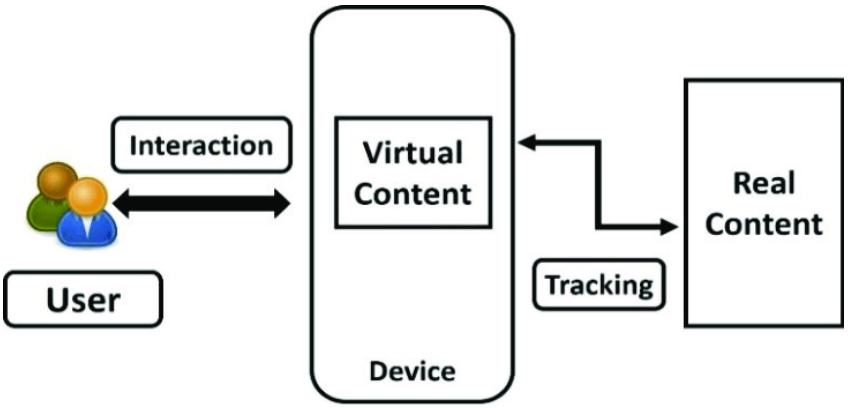
Postoje tri ključne komponente sustava proširene stvarnosti koje podržavaju ranije navedene korake:

1. Senzor/i za određivanje stanja fizičkog svijeta u kojem je aplikacija korištena.

2. Procesor za procjenu i obradu senzorom prikupljenih podataka, kako bi aplikacija mogla primijeniti „zakone prirode“ i druga pravila u virtualni svijet te generirati signale potrebne za prikaz.

3. Prikaz prikladan za stvaranje dojma o koegzistenciji virtualnog i fizičkog svijeta te da utječe na korisnikova osjetila kako bi osjetio kombinaciju fizičkog i virtualnog svijeta

Slika 2 vizualno prikazuje arhitekturu generičkog sustava proširene stvarnosti sa svim ključnim komponentama.



Slika 2: Arhitektura proširene stvarnosti

(Izvor: https://www.researchgate.net/figure/Augmented-reality-architecture-comprised-by-six-elements_fig1_283962218)

Moguće je definirati i četvrtu komponentu, ulazne uređaje – gumbe, tipkovnice i druge pokretače. Ti uređaji su zapravo senzori u čijem postavljanju vrijednosti korisnik ima aktivnu ulogu. Drugi senzori su pasivniji po prirodi, iz perspektive direktnog utjecaja korisnika na njihovu vrijednost. (Craig, A. B. 2013)

Uzimanjem u obzir ranije navedene korake i komponentne moguće je odrediti nekoliko ključnih „sastojaka“ iskustva proširene stvarnosti. Svaki sustav koji koristi proširenu stvarnost bi trebao barem njih imati u sebi, a oni su:

1. Aplikacija proširene stvarnosti – programsko rješenje koje orkestrira i upravlja raznim aspektima iskustva proširene stvarnosti.

2. Sadržaj – odnosi se na sve objekte, ideje, priče, stimulacije i „zakone prirode“ ključne za iskustvo. Može uključivati i simulacije, pravila igri i bilo koju drugu vrstu sadržaja koja je pod kontrolom aplikacije.

3. Interakcija – aplikacija koja koristi proširenu stvarnost mora biti interaktivna na neki način. Najčešća vrsta interakcije je prikaz miksa stvarnog i digitalnog svijeta. Osim te osnovne interakcije moguća je interakcija sa gumbima na zaslonu, gestama ili dodirom i druge.

4. Tehnologija – svako iskustvo proširenom stvarnosti koristi neku količinu tehnologije, neka iziskuju sofisticiraniju tehnologiju od drugih, ali je osnovna razina ista. Prikupljanje podataka o stvarnom svijetu putem senzora, vrstu obrade tih podataka i zaslon na kojem se mogu prikazati.

5. Fizički svijet – svako iskustvo proširenom stvarnosti se odvija u stvarnom svijetu. Fizički svijet je ključan dio iskustva proširenom stvarnosti. Može biti vezan uz neku specifičnu geografsku lokaciju poput Eiffelovog tornja, ali može biti i u bilo kojem fizičkom prostoru koji zadovoljava neke uvjete, npr. ravna ploha.

6. Korisnik – sva „čarolija“ proširene stvarnosti se odvija u umu korisnika proširene stvarnosti. Uloga proširene stvarnosti je kreirati umjetnu stimulaciju korisniku kako bi on vjerovao da je stvarno nešto što nije. (Craig, A. B. 2013)

Interakcija sa proširenom stvarnosti može biti postignuta na različite načine i izuzetno je važno objasniti osnovne vrste interakcije iz perspektive načina prikaza sadržaja. Sve vrste

proširene stvarnosti koriste informacije iz stvarnog svijeta za prikaz sadržaja ali ne pružaju istu razinu uranjanja. Tri su najčešće vrste proširene stvarnosti:

1. Proširena stvarnost bazirana na lokaciji – ova vrsta proširene stvarnosti je najbolje prikazana na primjeru aplikacija poput video igre *Pokemon GO* gdje se podatci prikazuju ovisno o lokaciji i položaju uređaja. Najčešće su korišteni za aplikacije korištene na otvorenom. Često su korištene za navigaciju u prometu, pronalaženje automobila na parkiralištu.

2. Proširena stvarnost na temelju markera – ova vrsta proširene stvarnosti koristi fizički marker koji aplikacija prepoznaje i pomoću njega određuje lokaciju za prikaz virtualnih objekata. Često su korišteni za prikaz objekata na malim udaljenostima i sadržaj je vezan uz fizički marker, ako se miče marker micati će se i objekt na njemu.

3. Proširena stvarnost bez markera – ova vrsta proširene stvarnosti precizno prepoznaje i stvara trodimenzionalni prikaz stvarnog svijeta oko korisnika. Zidovi, stolovi i drugi objekti su mapirani i omogućuju virtualnim objektima interakciju sa njima. Aplikacije koje pružaju ovu vrstu proširene stvarnosti pružaju najveću razinu uranjanja. (Maheshwari, A. 2019)

Ovo potpoglavlje se posvetilo načinima na koje sustavi proširene stvarnosti kreiraju iskustvo sudionicima. Predstavlja ideje različitih senzora za prikupljanje inputa iz fizičkog svijeta, osnovama procesiranja informacijama i načinima kako prikazati te podatke korisnicima u stvarnom svijetu. Sljedeće će poglavlje detaljnije obraditi uređaje pomoću kojih je to ostvarivo.

4.1.1. Uređaji za prikaz proširene stvarnosti

Kako bi korisnik mogao vidjeti digitalno proširenu svijet i imati interakciju s istim, potrebna mu je neka vrsta uređaja koja mu to može omogućiti. Svaki uređaj koji se koristi za proširenu stvarnost treba imati barem kameru i neku vrstu zaslona na kojem će korisnik moći vidjeti spoj stvarnog i digitalnog. Postoje još i neki drugi senzori koji mogu poboljšati uvjerljivost tehnologije, poput senzora dubine. Najčešći uređaji korišteni za prikaz proširene stvarnosti su pametni telefoni i tableti. Za vrijeme pisanja ovog rada razvija se velik broj pametnih naočala i

drugih uređaja koje korisnik ima na glavi i direktno ispred njegovih očiju prikazuju proširenu stvarnost (najbolji primjer *HoloLens*). (Carmigniani J, Furht B, Anisetti M, Ceravolo P, Damiani E, Ivkovic M, 2011.)

Zaslon platforme koja koristi proširenu stvarnosti kreira prozor u virtualni svijet i najprepoznatiji je dio ekosustava proširene stvarnosti. Ti uređaji uključuju niz senzora, tehnologija zaslona i jedinstvenih sučelja koji zajedno stvaraju željenu iluziju. (Maheshwari, A. 2019)

Različite razine uranjanja, društvene interakcije i kontrole od strane korisnika su omogućene od strane raznih faktora. Veličini, konfiguraciji i procesorskoj snazi uređaja. Također imaju drugačije ciljane skupine, tržišta i cijene. Developeri aplikacija proširene stvarnosti trebaju oprezno uzeti u obzir za koliko uranjanja je ciljani korisnik voljan platiti. Uređaji u pitanju su sljedeći:

1. Pametni telefoni
2. Pametni telefon postavljen na glavu
3. Zaslon postavljen na glavu

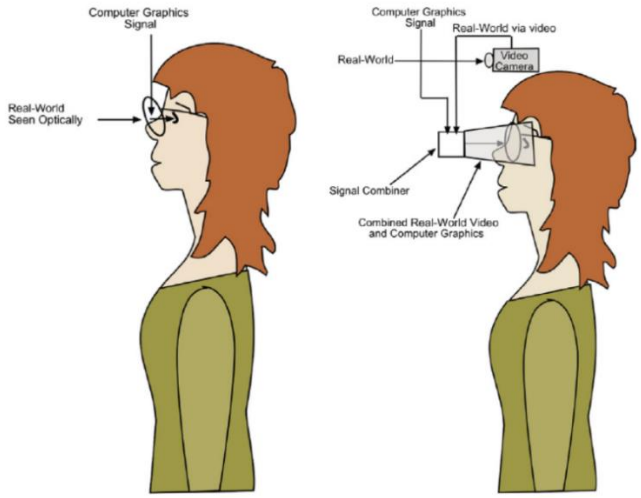
Pametni telefoni su idealni za većinu korisnika i svakodnevnu upotrebu jer ne zahtijevaju nikakvu dodatnu kupovinu hardvera. Korisnik može koristit ekran na dodir za interakciju a velika rasprostranjenost pametnih telefona omogućuje veću interakciju korisnika aplikacija. (Maheshwari, A. 2019)

Pametni telefon postavljen na glavi (eng. *head-mounted phone*) je kombinacija pametnog telefona sa okvirom koji izolira svijet izvan zaslona. Najbolje primjer je Google Cardboard, okvir je izrađen od kartona, financijski je isplativ. Potreban je dodatan hardver, ali hardver u pitanju je karton ili plastika. Uranjanje je bolje od onog postignutog samo pametnim telefonom. Korisnik ne može „zavirit“ iza mobitela i vidjeti da se digitalni objekt zapravo ne nalazi tamo gdje ga zaslon prikazuje. (Maheshwari, A. 2019)

Zaslon postavljen na glavu (eng. *head mounted display*) poput *HoloLens*-a ozbiljnom korisniku tehnologije pruža najviše udobnosti i procesorske snage. Također, budući da je uređaj na korisnikovoj glavi oslobađaju ruke i omogućuju interakciju sa aplikacijama pomoću gesti. (Maheshwari, A. 2019)

Pametni telefoni koriste hardver dostupan u uređajima i većina optimizacije dolazi iz softvera. Zaslon postavljen na glavu sa druge strane ima hardver specifično izgrađen za

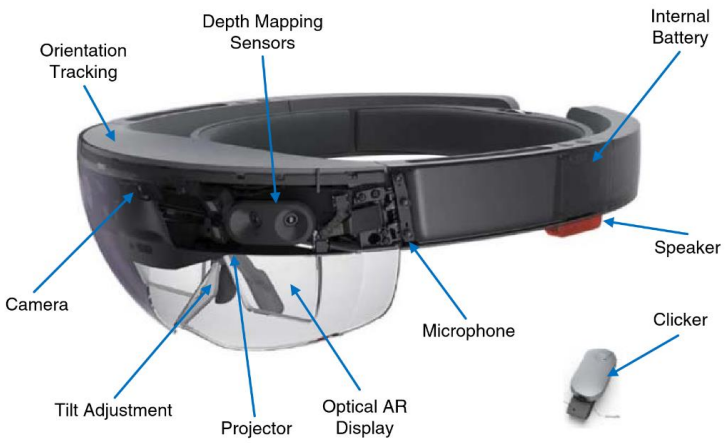
proširenu stvarnost. Koriste dva zaslona, po jedan za svako oko, i udaljeni su sveg nekoliko centimetara od očiju i prikazuju dva namještena pogleda sadržaja kako bi stvorili iluziju dubine. Zasloni korišteni za proširenu stvarnost su prozirni. Slika 3 prikazuje koncept zaslona postavljenog na glavu.



Slika 3: Koncept zaslona postavljenog na glavu

(Izvor: Understanding Augmented Reality : Concepts and Applications)

Zasloni postavljeni na glavi slojevito nižu sadržaj na polu prozirnu leću koristeći tehnologiju zvanu *planar waveguides*. Svjetlost je detektirana kroz površinu leće i preusmjerena u oči koristeći istim principima kao u optičkim kablovima. Omogućuje korisniku pregled digitalnog sadržaja unutar konteksta stvarnog svijeta. Nažalost tehnologija pati od ograničenog vidnog polja i fluorescentne palete boja, budući da display može samo povećavati osvjetljenje, a ne može se zacrniti. Dobar primjer takvog uređaja je *HoloLens*. Slika 4 prikazuje izgled *HoloLens*-a sa označenim dijelovima.



Slika 4: HoloLens

(Izvor: Digital Transformation : Building Intelligent Enterprises)

HoloLens je eksperimentalni zaslon postavljen na glavi za prikaz proširene stvarnosti sa fokusom na primjenu u poslovanju. Cijeli sustav je samodostatan, sve se nalazi unutar uređaja: računalna snaga, pohrana i napajanje. Digitalni sadržaj je prikazan pomoću dvije polu prozirne *planar waveguide* leće.

HoloLens koristi „*inside out*“ praćenje pozicije da mapira svoje okruženje bez eksternih senzora. Dizajn uređaja korisnicima daje slobodu kretanja, ali time i ograničava korištenje kontrolera, budući da nema vanjskih sustava. HoloLens koristi kombinaciju pogleda, gesti i glasovnih naredbi i uključeni „kliker“ – uređaj koji pomaže u korištenju.

4.1.2. Razvoj aplikacija proširene stvarnosti

Razvoj proširene stvarnosti je puno pristupačniji, za razliku od razvoja drugih tehnologija industrije 4.0, kao na primjer umjetne inteligencije koja se može vrlo jednostavno koristiti, ali teško razviti samostalno, kao pojedinac, jer je za kvalitetan razvoj potrebna infrastruktura za treniranje modela koju trenutno posjeduju samo najveće organizacije poput *Google*-a i *Microsoft*-a. Internet stvari je u nešto boljoj situaciji, ali također iziskuje znanje o mikroprocesorima i elektrotehnici, te je fizički potrebno nabaviti razne senzore i čipove. Proširena stvarnost je drugačija po tome što je većina potrebne infrastrukture već u pametnim telefonima, samo je potrebno znanje razvoja mobilnih aplikacija i računalne grafike, a postoji velik broj javno dostupnih i besplatnih SDK-eva (eng. *Software development kit*) poput *ARCore*-a i *Vuforije*. Uzevši to u obzir i promatrajući trenutne trendove, proširena stvarnost nije toliko uzvikana kao umjetna inteligencija i internet stvari, ali je zato puno pristupačnija za razvoj, i njezini potencijali će biti maksimalno iskorišteni tek kad hardverska rješenja poput pametnih naočala budu usavršena i masovno proširena, a do tad su najbolji načini i platforme razvoja pametni telefoni.

Za razvoj aplikacija proširene stvarnosti postoje razni načini, najbolji je preko nekog od dostupnih razvojnih okruženja. Najbolja razvojna okruženja su takozvana *game engine* razvojna okruženja od kojih su najpoznatiji *Unity* i *Unreal Engine*. Ta razvojna okruženja su primarno korištena za razvoj videoigara, ali su zbog svojeg fokusa na 3D vizualizaciju odlični za razvoj aplikacija proširene stvarnosti. (Maheshwari, A. 2019)

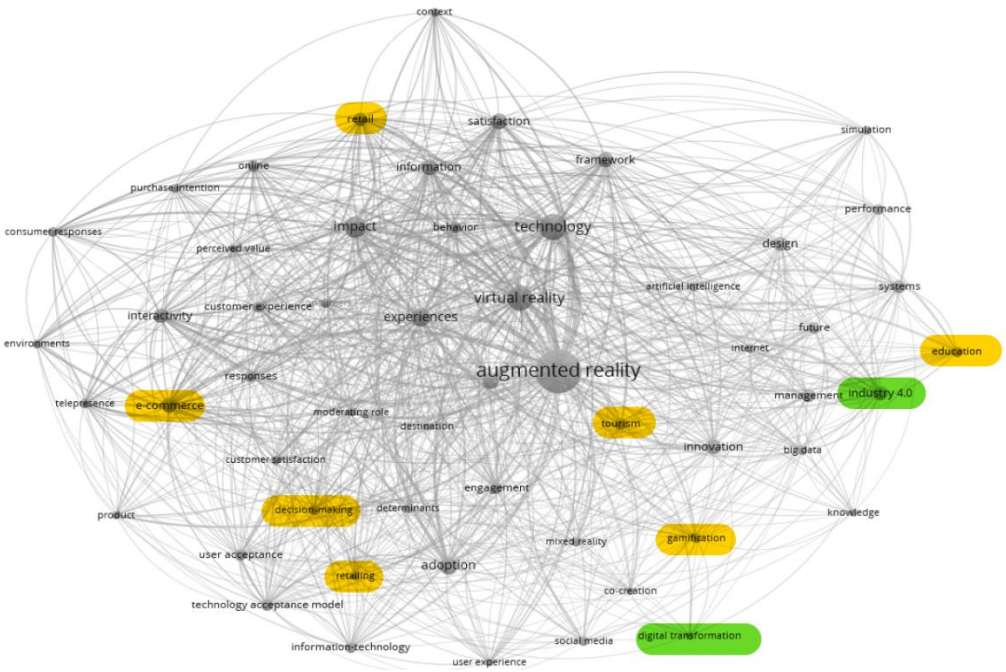
Unreal Engine je jedan od prvih razvojnih okruženja za razvoj videoigara koji je vodio na tržištu i bio često korišten za razvoj tehnološki naprednih videoigara za igrače konzole. Najpoznatiji je po svojem naprednom renderiranju grafike, koja je ključna za proširenu stvarnost. (Maheshwari, A. 2019)

Unity će biti korišten za razvoj aplikacije u sklopu ovog rade pa će o njemu biti više govora u kasnijim poglavljima.

4.2. Proširena stvarnost u poslovanju

Većina dostupne literature o proširenoj stvarnosti se fokusira na njezinu generičku upotrebu ili specifične teme, poput zdravstva ili turizma. Međutim, proširena stvarnost je postala jedna od glavnih tema u poslovanju i ekonomiji budući da je jedan od glavnih pokretača rasta disruptivnih tvrtki. (Jajic, I., Khawaja, S., Hussain Qureshi, F., & Pejić Bach, M. 2022)

Zbog svojih velikih potencijala za utjecaj u znanosti, poslovanju i obrazovanju dostupna stručna literatura obuhvaća niz tema, od općenitijih tema do onih koje se fokusiraju na specifične tehnologije unutar proširene stvarnosti. Ne postoji puno bibliometrijskih analiza proširene stvarnosti u poslovanju i ekonomiji. Neke postoje ali uglavnom proučavaju neki segment poslovanja, poput proizvodnje ili turizma. (Jajic, I., Khawaja, S., Hussain Qureshi, F., & Pejić Bach, M. 2022) Istraživanje korišteno u ovom poglavlju je „*Augmented Reality In Business And Economics: Bibliometric And Topics Analysis*“. Istraživanje se bavilo analizom stručne literature i tražilo u kojim se vrstama poslovanja najviše pojavljivala proširena stvarnost. Slika 5 prikazuje klaster analizu.



Slika 5: Klaster analiza

(Izvor: Augmented Reality in Business and Economics: Bibliometric and Topics Analysis)

Zelenom bojom je označena povezanost proširene stvarnosti sa digitalnom transformacijom i industrijom 4.0, što je zanimljivo zbog sadržaja prijašnjih poglavlja. Žutom bojom su označeni tipovi poslovanja najčešće povezani sa proširenom stvarnosti. Oni uključuju: turizam, obrazovanje, prodaju (online trgovinu). z njih su još označeni igrifikacija

(eng. *gamification*) i donošenje odluka. Igrifikacija je usko povezan uz ranije navedene tipove poslovanja jer se preko igrifikacije može direktno implementirati proširena stvarnost.

Sljedeća potpoglavlja će se posvetiti proširenoj stvarnosti u obrazovanju i turizmu. Više govora o proširenoj stvarnosti u prodaji biti će u ostatku rada jer je glavna primjena aplikacije čiji je razvoj ovog rada namijenjena primjeni u sklopu prodaje.

4.2.1. Primjer proširene stvarnosti u školstvu i edukaciji u poslovanju

Proširena stvarnost u obrazovanju se može gledati iz više aspekata. Obrazovanje u smislu školskih i sveučilišnih sustava i smislu obrazovanja i edukacije radnika na nekom zanimanju.

Proširena stvarnost u edukaciji će biti prikazana na primjeru „*The Effects of Virtual Reality, Augmented Reality, and Mixed Reality as Training Enhancement Methods: A Meta-Analysis*“
Istraživanje je imalo za cilj analizirati uspješnost edukacije zaposlenika u virtualnoj, proširenoj i miješanoj stvarnosti i usporedbu sa edukacijom u živo.

Edukacija je potrebna kako bi ljudi razvili potrebne vještine. Protokoli učenja mogu biti skupi i uzimati puno vremena. Zbog toga su svi napredci u tehnologiji ili metodologiji koji bi mogli smanjiti trošak ili vrijeme potrebno za edukaciju dobrodošli. Iz tih razloga edukacija pomoću simulacija postaje sve češća. Istraživanju su provedena za razne industrije, poput zrakoplovne i proizvodne. Rezultati istraživanja su pokazali da je edukacija koristeći proširenu stvarnost dala jednake rezultate kao i tradicionalna edukacija i pokazala svoju korist. (Kaplan, A. D., Cruit, J., Endsley, M., Beers, S. M., Sawyer, B. D., & Hancock, P. A. 2021)

Osim aspekata uštede novca i vremena smanjuje i rizik, simulatori leta smanjuju rizik koji dolazi sa stvarnim letenjem. S druge strane neočekivane situacije poput COVID-19 pandemije mogu spriječiti obrazovanje u područjima gdje je fizička praksa potrebna. Dobar primjer je medicina, proširena stvarnost je omogućila praćenje operacijskih zahtjeva puno većem broju studenata bez fizičkih ograničenja. (Kaplan, A. D., Cruit, J., Endsley, M., Beers, S. M., Sawyer, B. D., & Hancock, P. A. 2021)

Slika 6: Proširena stvarnost u zdravstvu

(Izvor: https://medium.com/swevens/augmented-reality-ar-in-healthcare-3c12bdf86a8e)

4.2.2. Primjer proširene stvarnosti u turizmu

Proširena stvarnost se sve više i više primjenjuje u raznovrsnim turističkim i uslužnim područjima poput zabavnih parkova, kruzera, muzeja i povijesnih znamenitosti. Turistička industrija je počela primjenjivati virtualni svijet kao inovativnog posrednika marketinga za stvaranje brenda turističkim destinacijama. Također ga je počela primjenjivati za razvoj interaktivnog oglašavanja i unaprjeđenje mišljenje turista o destinaciji. (Wei, W. 2019)

Velik broj aplikacija koje koriste proširenu stvarnost u turizmu korisniku pružaju inovativne načine revitalizacije povijesnih znamenitosti kako bi privukle više posjetitelja i poboljšale kvalitetu usluge. Proširena stvarnost transformira način na koji posjetitelj iskusi posjetu nekoj destinaciji ili atrakciji pružajući dublje, interaktivnije i diversificiranije iskustvo. (Wei, W. 2019)

Iz ranije navedenih stavki moguće je zaključiti da je značajni naglasak na marketingu korisnikovog iskustva. Proširena stvarnost je najprisutnija u turističkim destinacijama, kulturnim znamenitostima, muzejima, palačama, šoping centrima, događajima i drugim turističkim atrakcijama poput izložbi umjetnina. (Wei, W. 2019)

Odličan primjer proširene stvarnosti u turizmu je partnerstvo singapuranske turističke zajednice i *Google*-a. *Google*-ov *ARCore* i singapuranska turistička zajednica su kreirali dva nova AR iskustva u *Merlion Park*-u i *Victoria Theather & Concert Hall*-u unutar svoje *Visit Singapore* aplikacije. Pomoću tih dvaju iskustava posjetitelji će moći vidjeti vođenu turu po tim znamenitostima. Aplikacija je kreirana uz pomoć *ARCore*-a i alata za razvoj *Unity*. *ARCore*

2022. godine lansirao svoj *Geospatial API* koji pomaže vezati objekte proširene stvarnosti za stvarne lokacije na Zemlji. (traveltomorrow.com 2023)

Slika 7: Marky Polo in Singapore

(Izvor: https://traveltomorrow.com/singapore-tourism-board-partners-with-google-to-create-augmented-reality-tours/)

4.3. Proširena stvarnost u ERP sustavima

Prošlo potpoglavlje je gledalo na proširenu stvarnost iz aspekta korištenja direktno u poslovanju, konkretnije na primjerima u turizmu i edukaciji. Ovo potpoglavlje će proučavati kako se proširena stvarnost primjenjuje u ERP sustavima. ERP sustavi sve više i više primjenjuju proširenu stvarnosti u sebi, a najbolje dolazi do izražaja kada je komplementarno korištena sa drugim tehnologijama industrije 4.0. Što se tiče poslovnih funkcija koje ERP sustavi obuhvaćaju najveća primjena proširene stvarnosti je u proizvodnji, ali nije i jedina, sljedeće će potpoglavlje detaljnije obuhvatiti najčešća područja primjene proširene stvarnosti unutar ERP sustava, a potpoglavlje nakon njega izazove i probleme koje tehnologija suočava u kontekstu ERP sustava.

4.3.1. Područja primjene

Područja primjene proširene stvarnosti u ERP sustavima široka, najveća primjena proširene stvarnosti unutar ERP sustava je u sklopu modula koji obuhvaćaju sljedeće poslovne

funkcije: proizvodnju, skladište i marketing. Poredane su po redoslijedu njihovog utjecaja u tim funkcijama.

Prva od navedenih funkcija je proizvodnja. U sklopu proizvodnje proširena stvarnost ima najveću primjenu u sastavljanju proizvoda, npr. automobila ili drugih visoko prilagodljivih proizvoda i u održavanju strojeva.

Usprkos visokoj automatizaciji koja karakterizira modernu proizvodnju, ljudi i dalje imaju ključnu ulogu u proizvodnji, koja bi trebala biti održana u tranziciji industrije 4.0 i inteligentnih ERP sustava. Čovjek u proizvodnji može podržati povećanu fleksibilnost i prilagodljivost zadataka. Ljudi imaju prednost u odnosu na strojeve u izradi visoko prilagodljivih proizvoda gdje se konfiguracija proizvoda i zadatci vezani uz njih stalno mijenjaju. Strojeve se uglavnom i dalje ne može tako lako prilagoditi za takve promjene pa tu uskače uloga ljudi, koja je isto obogaćena primjenom tehnologije industrije 4.0, proširene stvarnosti. Ovisno o proizvodu u pitanju čovjek dobije upute putem okruženja u oblaku ERP sustava u svoje pametne naočale za svaki prilagodljivi proizvod i bez problema je spreman za njegovo slaganje. Taj pristup ja najčešće primijenjen u industriji proizvodnje automobila. (Mourtzis, D., Zogopoulos, V., & Xanthi, F. 2019)

Druga funkcija je funkcija skladišta. Razne tehnologije industrije 4.0 imaju veliki utjecaj na tradicionalnu logistiku skladišta. Zbog stalnog rasta automatizacije u proizvodnji i logistici porasla je i potražnja za integracijom ljudi opremljenih sa opremom koja koristi tehnologije industrije 4.0 u poslovni proces. Uzmimo kao primjer potpuno automatizirani proces transporta unutar skladišta, koji je maksimalno optimiziran. I dalje postoji prostor za unaprjeđenje, prostor gdje automatizacija nije na istoj razini i postoji ovisnost o ljudima. Dobar primjer je u prostoru za preuzimanje narudžbi, zaposlenici trebaju pronaći pravi proizvod, zatim ga skenirati i zatim uzeti točnu količinu proizvoda. Na prvi pogled možda trivijalan zadatak, ali on iziskuje koncentraciju i greške se često događaju i najiskusnijim radnicima. Proširena stvarnost ovdje rješava problem tako da pomoću prepoznavanja prostora pomaže u lociranju, automatski skenira i unosi potrebne podatke u ERP sustav i smanjuje greške, čak i broji je li čovjek uzeo točnu količinu robe prepoznavanjem oblika. (Wang, W., Wang, F., Song, W., & Su, S. 2020)

Treća funkcija je funkcija marketinga, ona je direktno vezana uz prodaju, praktični primjer koji će biti kasnije opisan kasnije u radu se također može klasificirati u prodaju/marketing. Primjena u marketingu je široka, najbolje svrstana u dvije kategorije. Prva je obogaćivanje onoga što već postoji koristeći proširenu stvarnost, a drugo je pružanju privida nečeg fizičkog kako bi korisniku olakšali npr. proces odluke u kupnji. Primjer obogaćivanja više je fokusiran na stvari poput turizma i on je opisan u ranijem poglavlju. Primjer pružanja privida

nečeg fizičkog kako bi korisniku olakšali proces odluke u kupnji je više vezan uz ERP sustave. Odličan primjer je u prodaji odjevnih predmeta. Kod kupnje odjevnih predmeta preko interneta ljudi su neodlučniji jer ne mogu isprobati odjeću, proširena stvarnost nudi prikaz 3D modela željenog odjevnog predmeta na vlastitom tijelu. Preko ERP sustava se može pratiti koliko je taj prikaz pomoću proširene stvarnosti povećao prodaju ili posjete web prodajnom mjestu. (Bulearca, M., & Tamarjan, D. 2010)

To su glavni primjeri primjene proširene stvarnosti u modulima poslovnih funkcija ERP sustava. Primjena koja će se koristiti u sklopu ovog rada je vizualizacija 3D modela namještaja poput stolova, stolica i ormara u kontekstu BORM Business ERP sustava koji je specijaliziran za drvoprerađivačku industriju, u obliku mobilne aplikacije. Ona bi stvorila dodanu vrijednost jednom takvom specijaliziranom ERP sustavu. Međutim, stvaranje vrijednosti nije nužno vezano samo uz specijalizirane ERP sustave nego se odnosi na njih u globalu, pa će se iduće poglavlje detaljno posvetiti kreiranju dodane vrijednost ERP sustavima kroz primjere slučajeva korištenja proširene stvarnosti.

4.3.2. Izazovi i problemi

Uz pozitivne prednosti navedene u prošlom potpoglavlju postoje izazovi i problemi koje je potrebno istaknuti.

Usprkos rastućem zanimanju za proširenu stvarnost i velikom broju istraživanja i primjene i dalje postoje neki problemi i izazovi koje treba dotaknuti. Moguće ih je klasificirati u sljedeće tri kategorije: tehnološki problemi, problemi društvene prihvatljivosti i problemi praktične primjene. Od kojih su oni tehnološki najbitniji i njihovo rješavanje direktno utječe na ostala dva. Kad je već u pitanju tehnologija tu postoji kvaliteta 3D modela, njihova uvjerljivost i ponašanje, reagiranje sa stvarnim izvorima svjetlosti itd. Postoje i hardverski problemi, oprema poput pametnih naočala je postala dosta manja, ali da bi tehnologija imala svakodnevnu upotrebu, npr. da svi zaposlenici imaju neku vrstu pametnih naočala i putem njih komuniciraju treba biti neprimjetna. Tu je i pitanje trajanja baterija uređaja, pogotovo u proizvodnji i pristupačnost cijene uređaja iz perspektive poduzeća i ERP sustava, trenutno je to jako veliki trošak. (Mekni, M., & Lemieux, A. 2014)

Usprkos svojoj visokoj naprednosti, proširena stvarnosti i dalje treba obraditi i razumjeti stvarnosti, uz sve mogućnosti inteligentnih ERP sustava i obrade podataka i dalje fizički svijet ima previše varijabli koje trenutno nije moguće prikupiti. U proteklih godinu dana od pisanja ovog rada detektori dubine su se jako razvili. Proširenoj stvarnosti je to jako bitno jer je jedan od najvećih problema kako prikazati virtualni objekt iza nekog fizičkog objekta bez korištenja

trikova na uvjerljiv način. Senzori dubine su na tom putu, ali tehnologija nije još dovoljno proširena da bi njezina primjena bila praktična.

5. Integracija proširene stvarnosti u ERP sustav

Ovo poglavlje se detaljno posvećuje integraciji proširene stvarnosti u ERP sustave. Postoji integracija u ERP sustav u obliku vanjskog modula, kako će to biti prikazano kasnije u radu na primjeru mobilne aplikacije. Ovo poglavlje će obuhvatiti direktnu integraciju proširene stvarnosti u poslovnim funkcijama proizvodnje i skladišta ERP sustava na konkretnim primjerima iz prakse.

5.1. Integracija u proizvodnji

Integracija ERP sustava u proizvodnji biti će prikazana na primjeru SAP-ovog ERP sustava. SAP ima preko 200 milijuna korisnika oblaka diljem svijeta i od 2016. godine ulaže u tehnologije poput proširene stvarnosti, virtualne stvarnosti i miješane stvarnosti kako bi pomogla svojim kupcima ubrzati vrijeme uvida i obogatiti vizualizaciju podataka. (immersivelearning.news 2023)

SAP ERP sustav je godinama bio korišten isključivo na računalima i mobilnim uređajima. To se promijenilo uvođenjem *SAP Extended Reality Cloud* rješenjem, koje koristi *Unity* za razvoj aplikacija koje koriste proširenu stvarnost. Čak je i kreiran *Unity Integration Toolkit* kako bi olakšao razvoj aplikacija koje se mogu integrirati u SAP sustave. (immersivelearning.news 2023)

XR Cloud servis i njegovi *Unity plugin*-ovi olakšavaju *Unity* developerima integraciju SAP podataka u svoje aplikacije koje koriste proširenu stvarnost. Kako te aplikacije stvaraju dodanu vrijednost za kupca:

1. Modeliranjem poslovnih procesa i treniranjem zaposlenika za implementacije timove SAP-a i krajnje korisnike
2. Proširenom stvarnosti za usluge, održavanje i kvalitetu inspekcije strojeva za vrijeme proizvodnje

U praksi je proširena stvarnost naglašena integracijom sa drugim tehnologijama industrije 4.0 i izgleda ovako u proizvodnji:

Nakon što klijenti instaliraju pametni uređaj, poput SAP-ovog ventila sa dvostrukim sjedištem, jednostavno ga mogu spojiti na Internet stvari pomoću desktop aplikacije. Stroj čiji su dijelovi povezani na internet stvari ima svog digitalnog blizanca preko kojeg se cijela proizvodnja može pratiti i pomoću umjetne inteligencije i strojnog učenja predviđa kada je potrebno servisirati stroj. Kada treba provesti popravak sustav šalje upit serviseru I on je pomoću proširene stvarnosti vođen do uređaja i proširena stvarnost mu pomaže zamijeniti potreban dio. (immersivelearning.news 2023)

Slika 8: SAP ERP i AR

(Izvor: https://www.immersivelearning.news/2021/02/26/case-study-sap-integrates-ar-for-erp/)

Iz primjera možemo zaključiti da je proširena stvarnost ključna u proizvodnji kod rješavanja problema i servisiranja strojeva i SAP ju aktivno primjenjuje u najnovijim inačicama svojeg ERP sustava.

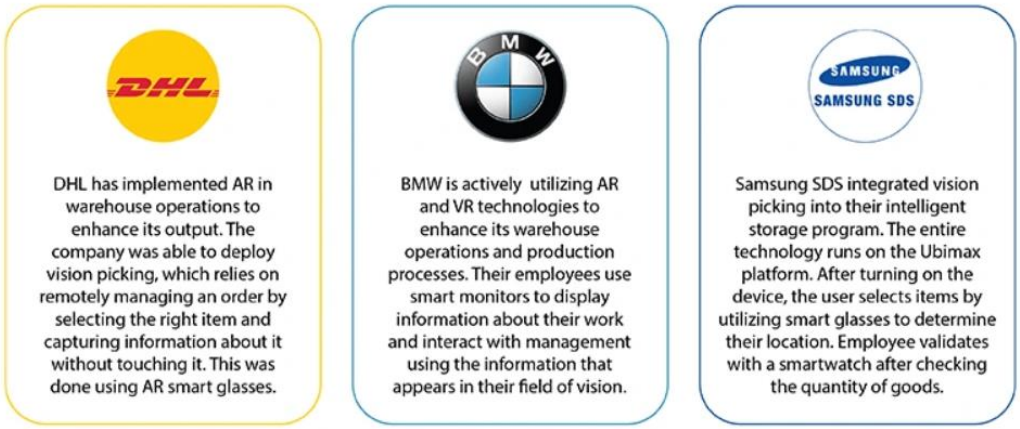
5.2. Integracija u skladištu

Druga funkcija integracije proširene stvarnosti u ERP sustav je u sklopu skladišta. Upravljanje skladištem uključuje upravljanje kompleksnim procesima koji obuhvaćaju raznolike dijelove koji su stalno u pokretu: ljudi, oprema, narudžbe i inventar. Većina poduzeća još koristi arhaične pristupe za upravljanje svojim skladištima. Također zbog trenutnih procedura u upravljanju nadležni stalno moraju rješavati probleme poput:

1. Kašnjenja zbog ručnih procesa
2. Krivo sastavljenih narudžbi
3. Velikih troškova održavanja zastarjelog sustava
4. Teško organiziranje čestih povrata robe
5. Krivo označavanje poremećuje i usporava rad

Implementacija proširene stvarnosti u skladište je najefektivniji način rješavanja takvih problema. Implementacija proširene stvarnosti u skladišta omogućuje zaposlenicima pregled tlocrta skladišta na jednostavan način, koji je dodatno obogaćen korištenjem zaslona na glavi za prikaz proširene stvarnosti.

Proširena stvarnost može poboljšati upravljanje skladištem svojom integracijom u ERP sustav. Najbolje prikazano kroz konkretne primjere velikih organizacije koje primjenjuju proširenu stvarnost u upravljanju skladištem. Slika 9 prikazuje neke od tih organizacija. (softwebsolutions.com 2023)



Slika 9: Primjeri proširene stvarnosti u skladištu
(Izvor: https://www.softwebsolutions.com/resources/ar-warehouse-transforming-supply-chain-with-ar-logistics.html)

Prvi primjer je DHL, poduzeće koje se bavi dostavom. Oni imaju implementiran *vision picking* u sklopu svojeg ERP sustava, ta tehnologija pomoću pametnih naočala prikuplja podatke o paketu bez fizičkog dodira paketa. Drugi primjer je BMW, proizvođač automobila koji koristi proširenu stvarnost kako bi obogatili radnicima svojeg skladišta vidno polje korisnim informacijama. Treći primjer je Samsung SDS, koji se bavi proizvodnjom pametnih uređaja i također u svojim skladištima koristi pametne naočale za pronalaženje stavki u skladištu. (softwebsolutions.com 2023)

Ovakav pristup upravljanju skladištem koristeći proširenu stvarnost u sklopu ERP sustava počinje standard u većim poduzećima jer klasični načini upravljanja skladištem jednostavno nisu dovoljno skalabilni za potrebe skladišta i svaka greška usporava rad cijelog poduzeća.

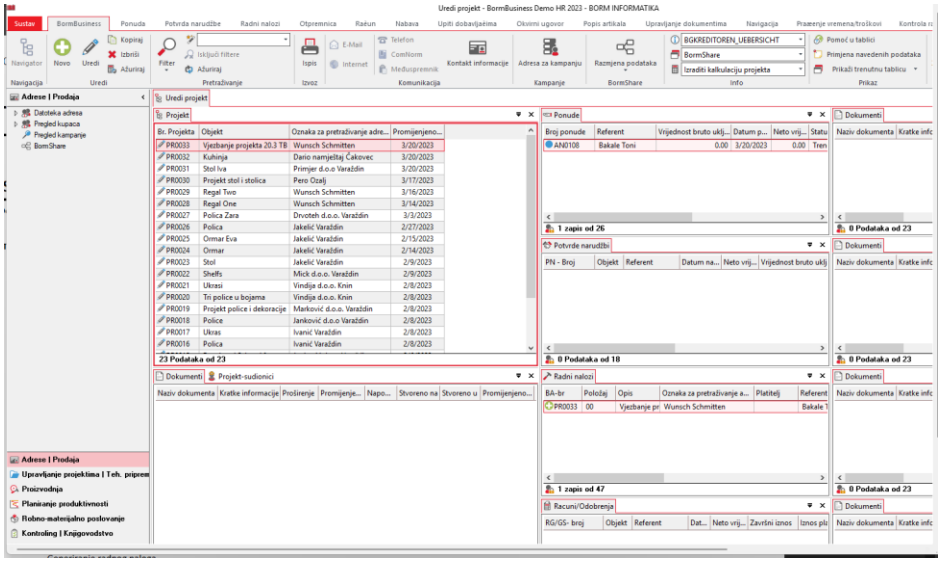
Ovim poglavljem je završen teoretski dio rada i prelazi se na praktični dio u kojem će biti prikazan razvoj mobilne aplikacije proširene stvarnosti za ERP sustav.

6. Praktičan primjer razvoja mobilne aplikacije proširene stvarnosti za ERP sustav

Ovo poglavlje predstavlja kraj teoretskog dijela rada i prelazak na opis praktičnog dijela. Praktični dio rada prikazuje razvoj mobilne aplikacije ARoom koja koristi proširenu stvarnost kako bi kreirala dodanu vrijednost BORM Business ERP sustavu. Prije opisa aplikacije potrebno je ukratko opisati BORM Business ERP sustav i PointLineCAD sustav koji su usko vezani jedno uz drugo.

6.1. BORM Business ERP

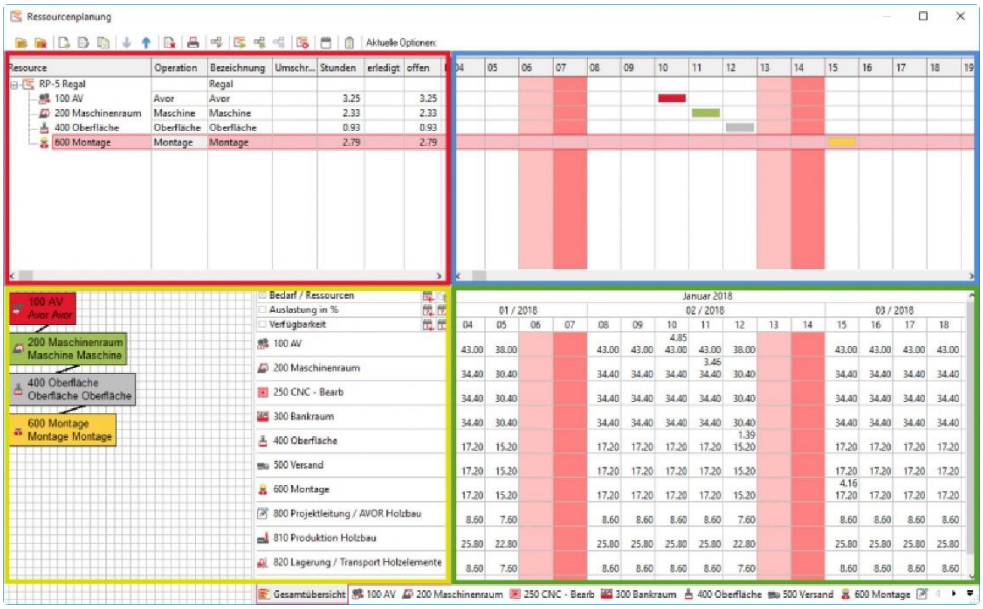
Business ERP je prilagodljivo softversko rješenje koje obuhvaća procese prodaje, pripreme proizvodnje, proizvodnje i fakturiranja, ali i individualne procese po potrebi poduzeća. Tijek rada u Business ERP sustavu je projektno orijentiran, najbolje prikazano pomoću slike 10.



Slika 10: BORM Business ERP

(Izvor: vlastita izrada)

Slika 10 prikazuje izgled uobičajenog zaslona Business ERP sustava. Sučelje je moguće uređivati a prikazani raspored je uobičajen. Odabirom nekog projekta iz popisa kartice „Projekt“ popunjavaju se ostale kartice sa podacima vezanim uz taj projekt. Najbolji način prikaza tijeka rada je kroz konkretni primjer.



Slika 11: Planiranje resursa

(Izvor: vlastita izrada)

Svaki projekt počinje sa upitom kupca. Kupac šalje upit, npr. mailom, i iz toga se kreira projekt. Kupac je nedavno izgradio zgradu u Opatiji i želi opremiti kuhinje svih apartmana. Opisao je

što bi sve htio u tim kuhinjama, ovisno o tome kakva je kuhinja u pitanju. Ako elementi nisu po mjeri, moguće je izvući sve potrebne podatke za ponudu iz baze podataka, inače je potrebno kuhinju nacrtati. Tu se vidi jedna od velikih prednosti Business ERP sustava - integracija sa Pointline CAD sustavom. Čest problem u drvoprerađivačkoj industriji je vrijeme
potrebno za unošenje raznih dijelova iz nacrta u ERP sustav. BORM taj problem rješava povezivanjem ERP sustava i CAD sustava tako da je moguće povući podatke potrebne za narudžbu direktno iz CAD nacrta i kreirati sastavnicu. Nakon što je ponuda kreirana i kupac ju odluči prihvatiti kreira se narudžba za potrebne dijelove i iz ponude se kreira radni nalog, te kreće proizvodnja. Proizvodnja u Business ERP sustavu je jako detaljno napravljena jer je ona najkompleksniji dio drvoprerađivačke industrije. Pruža detaljno planiranje resursa (prikazano na slici 11) i upravljanje materijalima i skladištem.

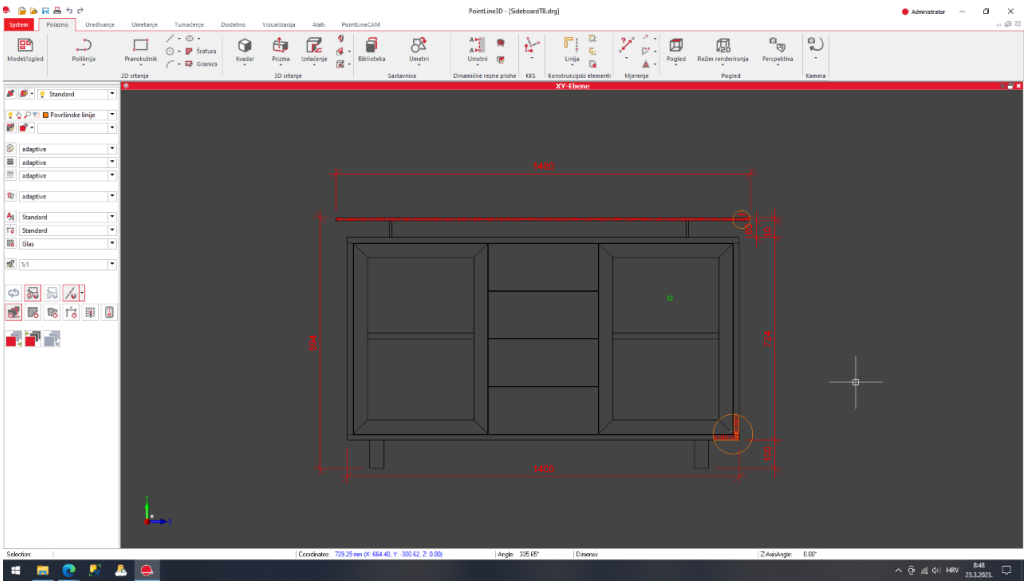
Nakon kreiranja plana proizvodnje proizvodnja može započeti. Preostali koraci u ERP sustavu su kreiranje/slanje računa i na kraju otprema/logistika.

Svaka faza je prikazana u obliku svojeg modula, a svaka kartica prikazuje zaseban modul, te postoji još niz drugih modula koji su mogući unutar Business ERP sustava, ovisno o paketu koji korisnik ERP sustava ima kupljen. Zbog tog modularnog pristupa prostor za nadogradnju je osiguran i otvara mjesto raznim načinima stvaranja dodane vrijednosti ERP sustavu, poput AR aplikacije čija je izrada tema ovog rada.

6.2. Pointline CAD

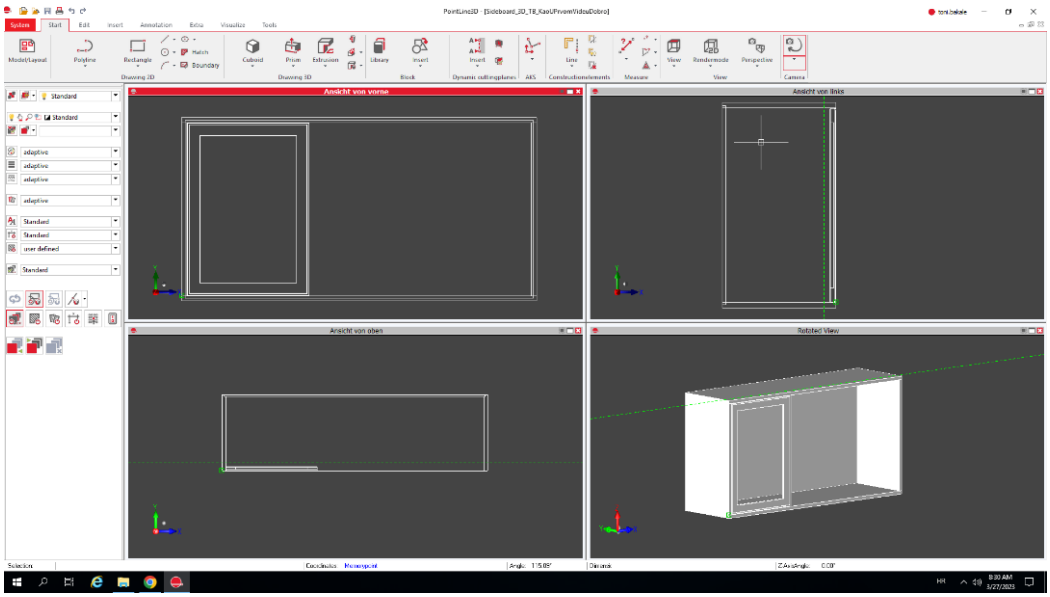
Makar je fokus rada na Business ERP sustavu potrebno je objasniti i Pointline CAD. BORM-ov klijent može kupiti licencu samo za ERP sustav ili samo za CAD sustav, ali njihovo zajedničko korištenje donosi najviše vrijednosti klijentu pa postoji i opcija kupnje paketa oba softvera.

CAD (eng. *Computer Aided Desing*) sustavi su aplikacije koje omogućuju crtanje, modificiranje i analizu 2D (nacrta) i 3D modela. Pointline nudi 2D i 3D crtanje na intuitivan i jednostavan način bez da narušava kvalitetu finalnog proizvoda i u osnovi je CAD sustav sa svim osnovnim funkcionalnostima. Ono što ga razlikuje su pojedinosti koje puno znače u drvoprerađivačkoj industriji, a drugi alati ih nemaju direktno ugrađene, primjerice automatskog kreiranja zazora, odnosno prostora između dijelova nekog elementa, npr. zazor u okviru ladice kako bi se lakše mogla otvarati. Slike 12 i 13 prikazuju izgled sučelja crtanja 2D i 3D modela.



Slika 12: Pointline 2D crtanje

(Izvor: vlastita izrada)



Slika 13: Pointline 3D crtanje

(Izvor: vlastita izrada)

Najveća prednost Pointline CAD sustava, osim integracije sa Business ERP sustavom, je *ParamLib*, biblioteka parametarskih modela. To su 3D modeli sa programiranom logikom u *Pythonu*. Npr. postoji parametarski model kuhinjskog ormarića, nakon uvoženja u CAD sustav moguće je manipulirati njegovim dimenzijama preko izbornika, te nije potrebno dodatno crtanje i uređivanje nacrta direktno, a logika automatski povećava broj polica, vijaka i svih ostalih varijabilnih dijelova. Ako korisnik ima sve elemente kuhinje u parametarskom obliku, u sustavu

ju je potrebno samo „složiti“ i izvesti (eng. *export*) sastavnicu u ERP sustav. Time se uvelike smanjuje vrijeme izrade nacrta kuhinje, a i smanjuje mogućnost pogreške.

Pointline CAD je važan za AR aplikaciju jer će se u njemu kreirati modeli korišteni za 3D prikaz.

6.3. Mobilna aplikacija – *ARoom*

Ovim potpoglavljem počinje opis razvoja aplikacije. Prošla poglavlja su služila upoznavanju čitatelja sa trendovima modernih i inteligentnih ERP sustava, industrije 4.0 i njezinih tehnologija, posebno proširenom stvarnosti i njezinim primjenama u ERP sustavima. Kombinacija raznih ideja i koncepata iz navedenih tema rezultirala je idejom razvoja aplikacije. Ostatak rada bavi se opisom izrade AR aplikacije, počevši sa metodikom korištenom u razvoju aplikacije, razvoj aplikacije prikazan kroz metodološki okvir osobnog ekstremnog programiranja i završava konceptom integracije aplikacije u BORM Business ERP.

6.3.1. Metodika razvoja – Osobno ekstremno programiranje

Metodika korištena u razvoju *ARoom* aplikacije je ekstremno programiranje (eng. *XP - extreme programming*) i to specifično – osobno ekstremno programiranje (eng. *PXP - Personal extreme programming*) jer je aplikaciju razvijala jedna osoba. Razlog odabira osobnog ekstremnog programiranja umjesto npr. SCRUM metodika je jednostavan, SCRUM je namijenjen radu u većim timovima i zahtijeva puno dodatnog dokumentiranja i planiranja.

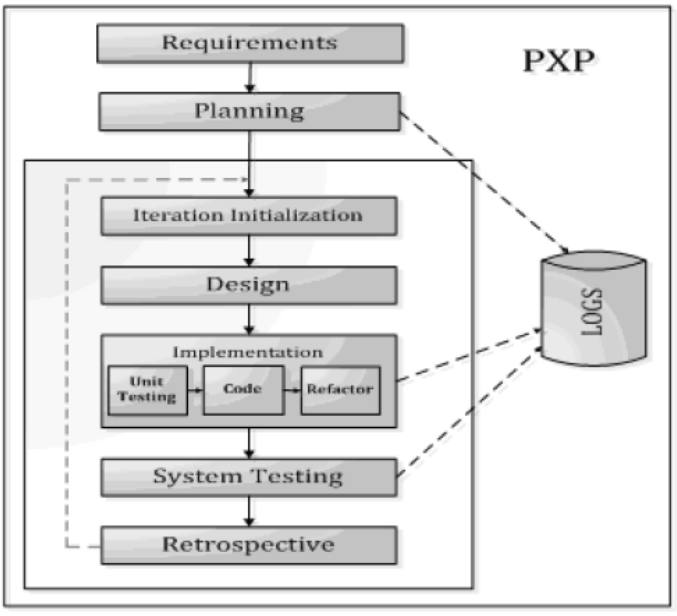
Osobno ekstremno programiranje je agilna metoda razvoja softvera osmišljen za razvoj aplikacija kada ih razvija samo jedan developer. Osobno ekstremno programiranje je bazirano na sljedećim principima:

1. PXP zahtijeva disciplinu – developeri su odgovorni za praćenje procesa i primjene PXP praksi
2. Developer bi trebao mjeriti, pratiti i analizirati svoj posao na dnevnoj bazi
3. Developer bi trebao učiti iz prikupljenih podataka i ciljati na poboljšanje procesa
4. PXP uključuje kontinuirano testiranje
5. Popravljanje defekata bi se trebalo raditi već u ranom razvoju aplikacije
6. Developeri bi trebali automatizirati što više posla

Ono što dijeli sa ekstremnim programiranjem su sljedeće dobre prakse:

1. Kontinuirana integracija – PXP uključuje verzioniranje koda, automatizaciju izgradnje aplikacije i automatizaciju testiranja
2. Jednostavan dizajn
3. Mala izdanja
4. Refaktoriranje
5. Testno vođen razvoj
6. *Spike solutions* – jednostavni programi za testiranje potencijalnih rješenja

Slika 13 prikazuje faze razvoja u PXP metodike. (Dzhurov, Y., Krasteva, I., & Ilieva, S. 2009)



poble

Slika 14: Faze razvoja u PXP metodici

(Izvor: Personal Extreme Programming – An Agile Process for Autonomous Developers)

Prva faza je prikupljanje korisničkih zahtjeva, korisnik developeru daje funkcionalne i ne-funkcionalne zahtjeve. Sljedeća faza je faza planiranja, u toj fazi developer slaže plan razvoja aplikacije, određuje razvojno okruženje i zatim slijedi treća faza – faza razvoja. Faza razvoja je iterativna, što znači da se njezine faze konstanto ponavljaju dok finalni proizvod ne bude gotov. Podfaze razvoja su početak iteracije, dizajn, implementacija i retrospektiva. Početak iteracije je početak novog kruga a retrospektiva je osvrt na taj krug. Glavne faze su faze dizajna implementacije i testiranja. (Dzhurov, Y., Krasteva, I., & Ilieva, S. 2009)

Ostatak ovog poglavlja će biti sistematiziran na temelju faza i podfaza razvoja PXP metodike. Iako je proces razvoja iterativan, zbog načina pisanja rada, sve faze će biti prikazane slijedno makar se tokom razvoja iterativno dizajniralo, implementiralo i testiralo.

6.3.2. Planiranje

Razna znanja stečena istraživanjem tema prijašnjih poglavlja su spojena u ideju aplikacije čija je svrha kreirati BORM Business ERP sustavu dodanu vrijednost. BORM Business ERP sustav je usredotočen na proizvodna poduzeća, specifično na drvoprerađivačku industriju, dijelovima ERP sustava sa opširnijim modulima za proizvodnju, povezanosti sa CAD sustavom, te je već time diversificiran. Kako bi ostao vodeći u svojoj niši i zadržao trenutne klijente potrebno je stalno unaprjeđivati sustav, dodavati nove module i time stvarati dodanu vrijednost sustavu.

Jedan od načina postizanja tog cilja je kreiranje novog modula, aplikacije koja ne obogaćuje samo ERP sustav, nego i cijelo poduzeće koje ga koristi. Dobar dio poduzeća koji se bavi proizvodnjom namještaja ga često i prodaje pa je ideja ove aplikacije obogatiti prodaju. Ideja aplikacije je sljedeća: mobilna aplikacija koja krajnjem korisniku (kupcu nekog namještaja) donosi pomoć u odlučivanju koristeći proširenu stvarnost za prikaz željenog namještaja u stvarnom prostoru. Uzmimo sljedeću personu kao primjer: Ivana želi kupiti novi stol za svoj dnevni boravak. Pronašla je stol koji joj se sviđa, ali nije sigurna hoće li joj se dobro uklopiti u estetiku dnevnog boravka. Aplikacija bi mogla pomoći Ivani tako da pomoću svojeg pametnog telefona u nekoliko klikova može vidjeti taj stol u svojoj sobi i vidjeti kako bi se uklopio.

Nakon ideje bilo je potrebno odraditi istraživanje tržišta o sličnim aplikacijama, vidjeti njihove prednosti i mane, uzeti dobro za uzor i imati loše na umu prilikom razvoja. Slična rješenja postoje, a najpoznatije je *IKEA Place*. IKEA je jedan od najvećih svjetskih prodavača namještaja i očekivano je da će biti u trendu sa novim tehnologijama. Međutim, njihova aplikacija ima jednu znatnu propuštenu priliku, dostupna je samo na iPhone uređajima, a oni prema posljednjim istraživanjima zauzimaju samo oko 18.55% ukupnog broja korištenih mobilnih uređaja prema bankmycell.com, tako da razvoj aplikacije treba biti za *Android* uređaje, ali najbolje oboje, *Android* i *iOS* uređaje. Drugi trend na tržištu je: slične aplikacije nisu dio nekog ERP sustava, već su zasebne aplikacije, čiji developeri nude opcije prilagodbe aplikacije poduzeću. Također, u ovakvim aplikacijama su intuitivnost, preciznost i dobro korisničko sučelje ključni pa je potrebno posvetiti puno pažnje na njih.

Sljedeći korak je određivanje razvojnog okruženja. Razvoj aplikacija za proširenu stvarnost je moguć u sklopu različitih okruženja, a okruženje koje je korišteno za izradu aplikacije u ovom radu je *Unity*, zbog dobre integracije za razvoj proširene stvarnosti, mogućnosti razvoja za više platformi i odličnom korisničkom sučelju za rad sa 3D objektima unutar aplikacije.

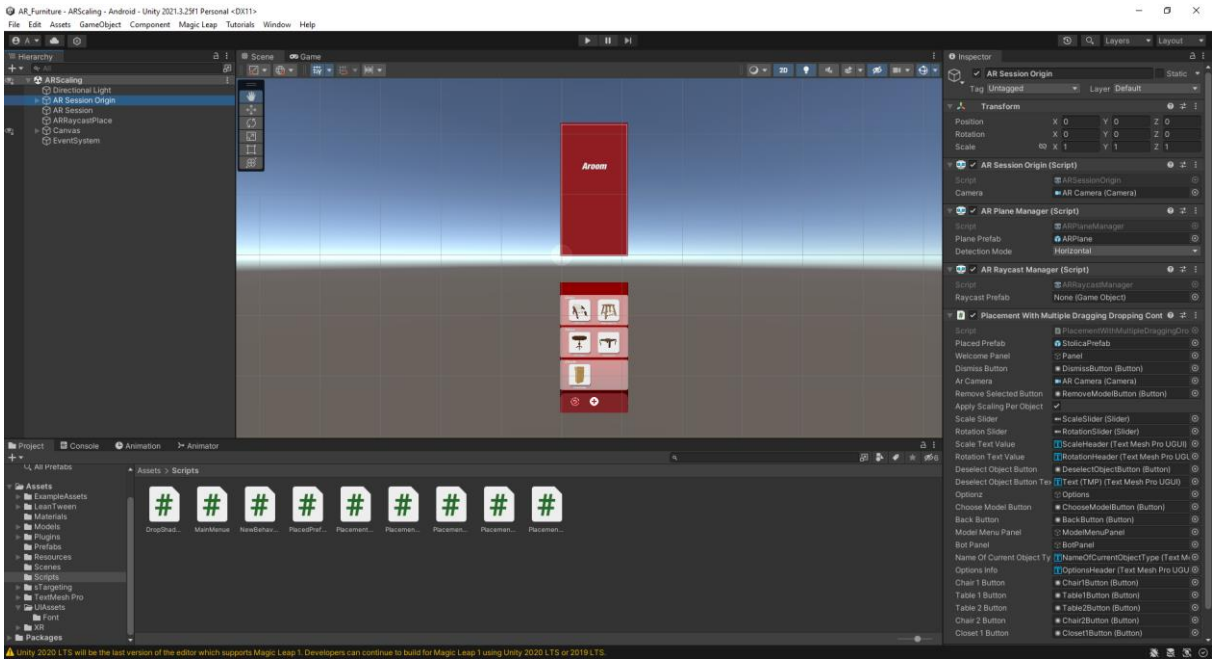
6.3.3. Razvojno okruženje – *Unity*

Unity je razvojno okruženje primarno korišteno za razvoj video igara, ali se može koristi i za razvoj drugih vrsta aplikacija. U njemu je moguće razvijati mobilne aplikacije (*Android* i *iOS*), web aplikacije, desktop aplikacije i aplikacije za konzole (*PlayStation*, *Xbox*, *Nintendo Switch*). Odličan je za razvoj aplikacija koje koriste 3D vizualizaciju pa tako i za aplikacije koje koriste proširenu stvarnost. Jako je intuitivan i jednostavan za korištenje početnicima, a veliki broj stvari u alatu se može postaviti i manipulirati sa minimalnom količinom direktnog kodiranja pomoću raznih izbornika. S druge strane pruža dubinu i slobodu vještijim developerima proširivanjem postojećih skripti i kreiranjem svojih. (unity.com 2023.)

U *Unity*-ju se sve vrti oko objekata koji su dio neke scene. Svaki objekt ima programiranu vlastitu logiku u obliku skripti, napisanih u programskom jeziku C#. Na konkretnom primjeru ovog projekta objekti su: kamera, elementi korisničkog sučelja i razni 3D objekti. Njima dodijeljene skripte prikazuju logiku, npr. kada se klikne neki gumb korisničkog sučelja ili se promijeni pozicija 3D modela na ekranu. (unity.com 2023.)

Unity također nudi veliki broj besplatnih i javno dostupnih API-a (eng. *application programming interface*), SDK-eva (eng. *software developement kit*) i *plugin*-ova (skupinama SDK-eva i API-a), koji pomažu u razvoju aplikacija. Najvažniji *plugin* korišten u ovom radu je *AR Foundation*, *Unity*-ev *plugin* koji koristi *Google*-ov *ARCore*, *Apple*-ov *ARKit* i omogućuje paralelni razvoj za više platformi (eng. *cross-platform*). (unity.com 2023.)

Slika 14 prikazuje izgled korisničkog sučelja *Unity* editora.



Slika 15: Unity razvojno okruženje

(Izvor: vlastita izrada)

6.3.4. Dizajn

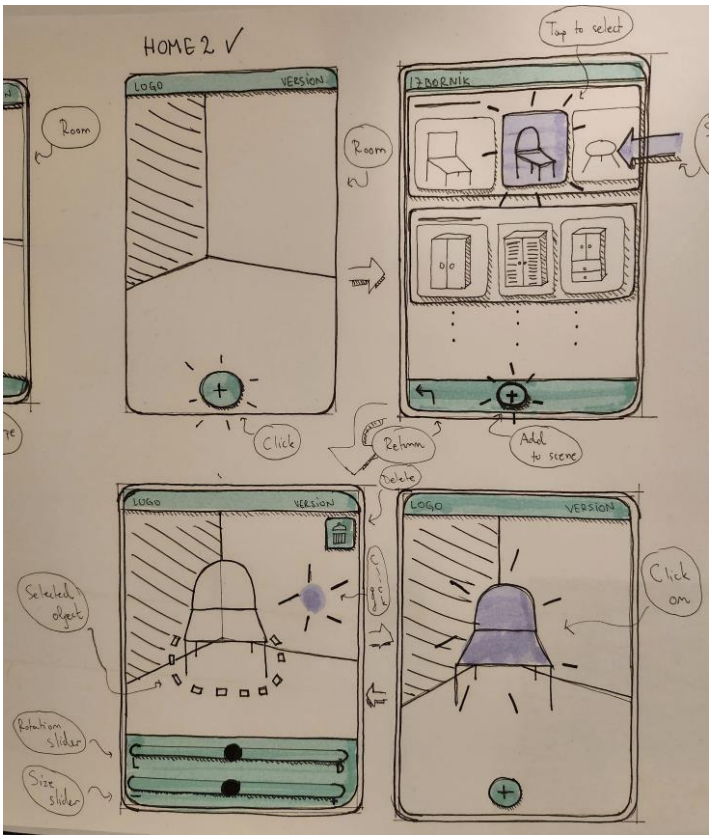
Dizajn počinje sa detaljnijom razradom ideje: mobilna aplikacija koja krajnjem korisniku (kupcu nekog namještaja) donosi pomoć u odlučivanju koristeći proširenu stvarnost za prikaz željenog namještaja u stvarnom prostoru.

Ime aplikacije je *ARoom* - kombinacija riječi AR i Room jer aplikacija koristi AR za prikaz namještaja u prostoru tj. sobi (eng. room). Opis razvoja će početi sa konceptualnim, koji se nadovezuje na ranije opisanu ideju.

Mobilna aplikacija za odabrano razvojno okruženje koristi *Unity*, platformu *Android*, mogući razvoj za više platformi, ali trenutno sa fokusom na *Android* jer je puno lakše testirati bez bespotrebnih komplikacija koje *iOS* okruženje stvara. Ova aplikacija će biti korištena samo ponekad pa je bitno da je aplikacija što jednostavnija i da nije potrebno duže vrijeme učenja za korištenje, te treba ukratko prikazati korisniku upute, u par rečenica i slika, i to je sve što mu je potrebno da ju počne koristiti.

Imajući na umu ranije navedeno, sljedeći korak u izradi aplikacije je skiciranje zaslona. U ovakvoj aplikaciji je naglasak na vizualnom i stoga je potrebno uložiti puno truda u izgled aplikacije.

Slika 15 prikazuje rukom nacrtane skice zaslona aplikacije.



Slika 16: Ručna skica aplikacije

(Izvor: vlastita izrada)

Prva skica, sa lijeva prikazuje početni zaslon aplikacije. Nakon što korisnik otvori aplikaciju prikazat će se neka vrsta uputa na zaslonu. Nakon toga će biti vidljiv zaslon poput ovog sa prve slike. U AR aplikacijama je bitno što više zaslona učiniti slobodnim kameri pa je ovakvo korisničko sučelje minimalno. Na vrhu zaslona nalazi se traka sa nazivom aplikacije i nekim drugim informacijama poput verzije aplikacije. U donjem dijelu zaslona nalazi se tipka sa plusom. Na neki način, animacijom, će trebati privući korisnikov pogled na tipku. U početku je to jedina interaktivna stvar na zaslonu.

Druga skica prikazuje izbornik koji se otvara klikom na gumb u donjem dijelu zaslona. Klik otvara izbornik za odabir željenog namještaja. U izborniku se nalazi lista dostupnog namještaja, grupiranog po tipu: stolovi, stolice, ormari... Korisnik može odabrati željeni namještaj i njegov odabir može potvrditi tipkom u donjem dijelu zaslona, na istom mjestu gdje je tipka bila i na prijašnjem zaslonu. Lijevo od tipke za potvrdu odabira je tipka za povratak u nazad, koja poništava odabir, kao da izbornik nikad nije bio otvoren.

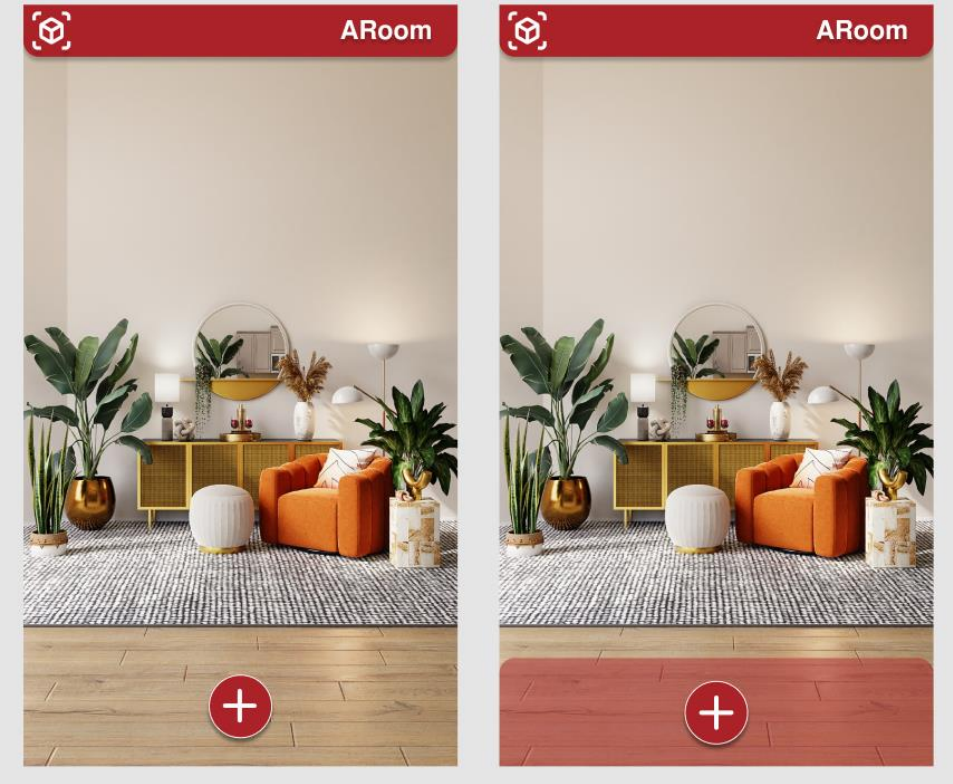
Treća skica prikazuje ponovno zaslon sa kamerom, ali ovaj put je moguće dodati 3D model dodirom na detektiranu površinu. Dodavanjem novog 3D modela otvara se izbornik za uređivanje modela. Aplikacija će omogućavati sljedeće manipulacije nad modelima: skaliranje, rotiranje, pomicanje po detektiranoj površini, odznačavanje i trajno brisanje.

Četvrtka skica prikazuje zaslon sličan prvom, ali sada sa novo dodanom stolicom. Do njega se dolazi odznačivanjem označenog objekta i time se zatvara krug korištenja. Korisnik može ponovno dodati novi komad namještaja, isti takav, ili novi, i uređivati scenu po želji.

Nakon izrade ručnih skica na papiru kreće izrada detaljnih digitalnih skica u programu za prototipiranje *Figma*. Ručne skice će biti detaljnije rekreirane u *Figmi*. Razlog za odabir *Figme* su mogućnost postavljanja preciznih dimenzija ekrana, boja i ikonica, koje je vrlo jednostavno prebaciti u *Unity*.

Potrebno je i paralelno početi razvoj aplikacije, testiranjem proširene stvarnosti u *Unity*-ju na testnoj aplikaciji. Nakon razvoja testne aplikacije slijedi dodavanje modela, micanje i rotiranje, i kada aplikacija funkcionalno radi zamjena primitivnog korisničkog sučelja novim, napravljenim po uzoru na skice izrađene u *Figmi*. Ostatak ovog poglavlja će opisati skice rađene u *Figmi*, a sljedeće će se fokusirati na tehničke dijelove aplikacije, AR logiku i analizu koda.

Slika 16 prikazuje drugu verziju dizajna početne stranice korisničkog sučelja.

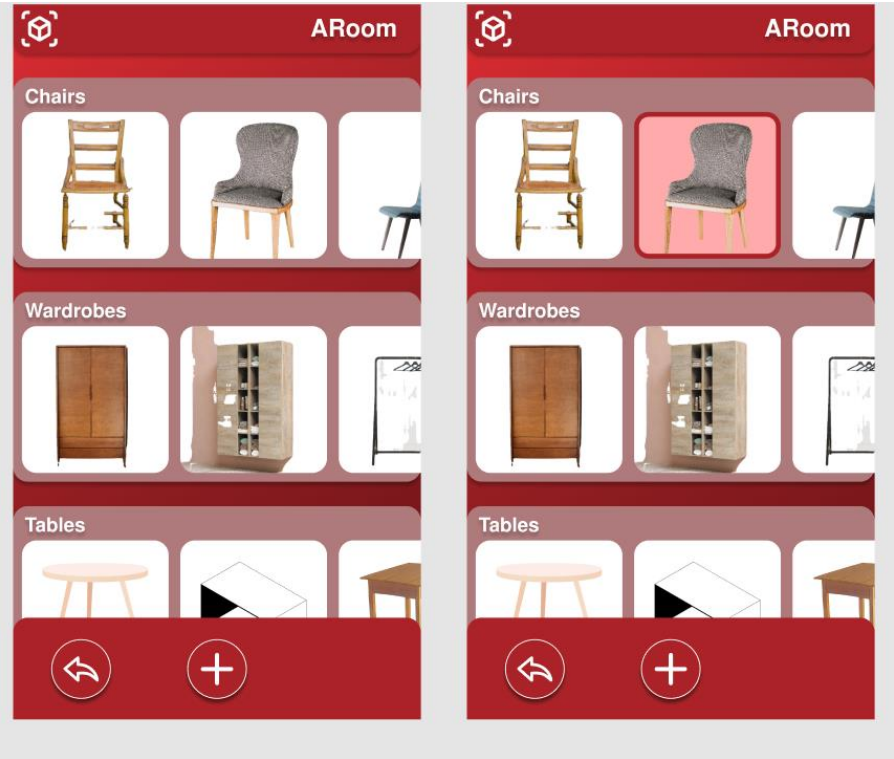


Slika 17: Skica početnog zaslona u alatu *Figma*

(Izvor: vlastita izrada)

Dizajn je gotovo identičan ručnoj skici, ali je bitan dodatak boje i fonta. Budući da je aplikacija zamišljena kao dodatak BORM Business ERP sustavu, boja i font su odabrani po uzoru na BORM. Oboje su prisutni na njihovim web stranicama. Dodani su i detalji poput sjena i prikazane su dvije varijante početne stranice, jedna sa okvirom oko donje trake i jedna bez.

Slika 17 prikazuje izbornik za odabir namještaja.



Slika 18: Skica odabira namještaja u alatu *Figma*
(Izvor: vlastita izrada)

Lijeva skica zaslona prikazuje popis bez označenog elementa, a desna sa označenim elementom. Skica je također jako slična ručnoj skici, sa dodanim bojama, fontom, sjenama i detaljima poput dodavanja okvira označenom objektu, promjeni boje pozadine i slično.

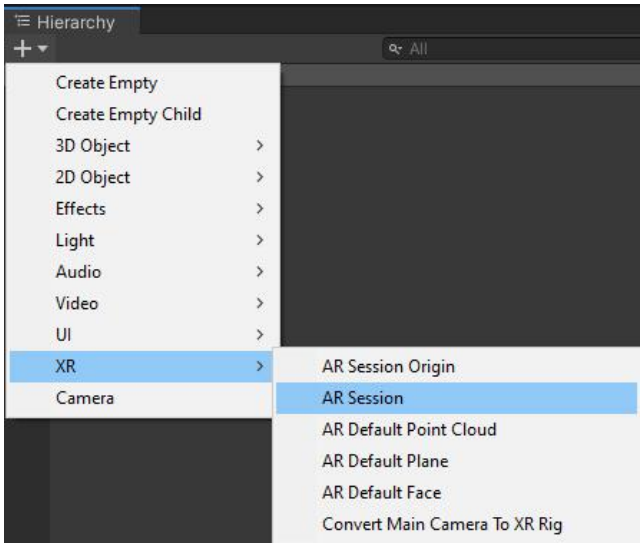
Jako je bitno da ova vrsta aplikacije ima jednostavno i oku ugodno sučelje. Crvena boja je žarka boja, što može biti dobro i loše. Dobro je kad nešto treba istaknuti, ali ako je izrazito žarka postaje previše uočljiva i oku neugodna, tako da je jako bitno uzeti dobru nijansu. Sučelje je što jednostavnije i minimalističko, kako bi se što veći naglasak stavio na kameru i na 3D modele koji se prikazuju.

6.3.5. Implementacija

Ovo poglavlje opisuje specifičnosti programskog koda i drugih tehničkih dijelova razvoja aplikacije u razvojnom okruženju *Unity*. Također, opisuje kronološki razvoj aplikacije, greške koje su se javljale u razvoju i načine na koji su riješene.

Prvi korak razvoja bio je kreiranje testne aplikacije. Svrha testne aplikacije je testiranje rada u *Unity* okruženju i razvoj jednostavne AR aplikacije koja postavlja sferu na detektiranu plohu. Prva stvar koju treba kreirati u razvoju AR aplikacije je detekcija ploha, jer ako taj dio

ne radi, ostatak razvoja nije moguć. Svaka AR aplikacija treba imati sljedeće objekte koji su dio *ARFoundation plugin*-a i vidljivi su na slici 18.

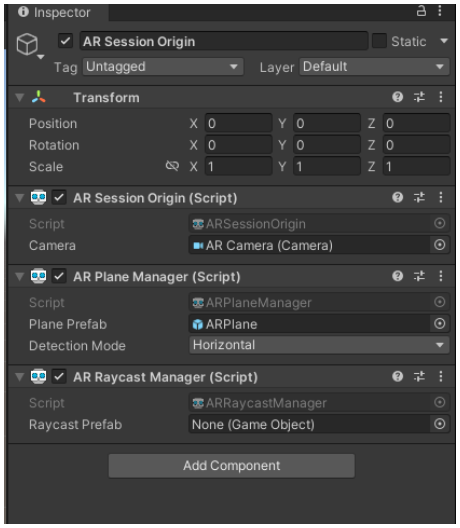


Slika 19: *Unity* hijerarhija
(Izvor: vlastita izrada)

Njihovo dodavanje je trivijalno. Kada je *plugin* dodan u projekt dodavanje je dostupno preko plusa za dodavanje objekata u hijerarhiju. Potrebno je dodati *AR Session*, *AR Session Origin* i obrisati *Main Camera* objekt, jer će njega zamijeniti *AR Camera*, objekt koji je dio *AR Session Origin*-a. (docs.unity3d.com 2023.)

ARSession - objekt koji kontrolira životni ciklus i postavke AR sesije.
ARSessionOrigin – sadrži kameru (*AR Camera*) i sve objekte koji su kreirani iz detektiranih ploha.

Detektiranje plohe je dio *AR Session Origin*-a. Preko inspektora je moguće promijeniti neke pojedinosti poput vrste ploha za detekciju, odnosno treba li aplikacija detektirati sve plohe ili samo vertikalne ili samo horizontalne. Zbog prirode ove aplikacije dovoljno je ostaviti samo horizontalne plohe jer se stolovi, stolice i ormari stavljaju na pod pa nema potrebe trošiti resurse na detekciju drugih ploha. Druga bitna stvar koju je potrebno namjestiti je tekstura detektirane plohe. Korisnik treba znati da postoji ploha i da na nju može postaviti objekt. Ta tekstura treba biti jednostavna i gotovo neprimjetna ako korisnik ne obraća pozornost direktno na nju. Slika 19 prikazuje inspektor, a slika 20 izgled teksture za detekciju horizontalnih ploha.



Slika 20: *Unity* inspektor

(Izvor: vlastita izrada)

Slika 21: Tekstura detekcije ploha

(Izvor: vlastita izrada)

Nakon što aplikacija detektira plohu potrebno je dodati neki objekt na mjesto dodira prsta sa plohom. Ako dodavanje radi sa kuglom, raditi će i sa bilo kojim drugim 3D modelom. Postoji više načina na koji se to može postići, jednostavniji način je korišten u testnoj aplikaciji, prikazan kodom dolje. Na temelju njega će kasnije biti napravljen kod za dodavanje objekata u finalnoj aplikaciji, ali u svrhu objašnjavanja osnovnih koncepata ova skripta je dovoljna.

```
using System.Collections;
using System.Collections.Generic;
using UnityEngine;
using UnityEngine.XR.ARFoundation;
```

```
using UnityEngine.XR.ARSubsystems;

public class ARRaycastPlace : MonoBehaviour
{
    public ARRaycastManager raycastManager;
    public GameObject objectToPlace;
    public Camera arCamera;

    private List<ARRaycastHit> hits = new List<ARRaycastHit>();

    void Update()
    {
        Ray ray = arCamera.ScreenPointToRay(Input.mousePosition);

        if (Input.GetMouseButton(0))
        {
            if (raycastManager.Raycast(ray, hits,TrackableType.PlaneWithinPolygon))
            {
                Pose hitPose = hits[0].pose;
                Instantiate(objectToPlace, hitPose.position, hitPose.rotation);
            }
        }
    }
}
```

Update() metoda u *Unity*-ju je metoda koja se poziva u svakoj sličici (eng. *frame*). *GetMouseButton(0)* predstavlja dodir na zaslon, ako se dodir na zaslonu podudari sa plohom onda se na toj točki dodira instancira objekt.

Nakon testiranja osnovnih funkcionalnosti *ARFoundation*-a počeo je rad na *ARoom* aplikaciji kreiranjem *git* repozitorija na *GitHub*-u. Početak razvoja je bio isti kao i u testnoj aplikaciji, dodavanje *AR Session* i *AR Session Origin* objekata.

Sljedeći korak je bio kreiranje *Canvas* objekta u kojem će se nalaziti svi elementi korisničkog sučelja. U njemu su se prvo nalazili primitivni elementi korisničkog sučelja za testiranje. U testnoj aplikaciji korištena je kugla, a ovdje je prvo korišten 3D model testne stolice koji će se kasnije mijenjati drugim modelima namještaja.

Prva funkcionalnost koju je bilo potrebno kreirati je postavljanje više stolica na prazne dijelove plohe, one dijelove gdje nema stolica. Zatim je potrebno dodati da korisnik odabranu stolicu može micati po plohi i da se ne stvori nova na mjestu dodira. Važno je postaviti micanje samo po plohi kako se ne bi stolica pomaknula ispod ili iznad plohe jer je ploha pod i nepoželjno je da stolica lebdi ili da je dio poda.

Nakon što je taj dio funkcionalan potrebno je dodati mogućnost promjene modela koji se dodaje. Na primjer: promjena iz dodavanja stolica u dodavanje stola. Također se na ekranu

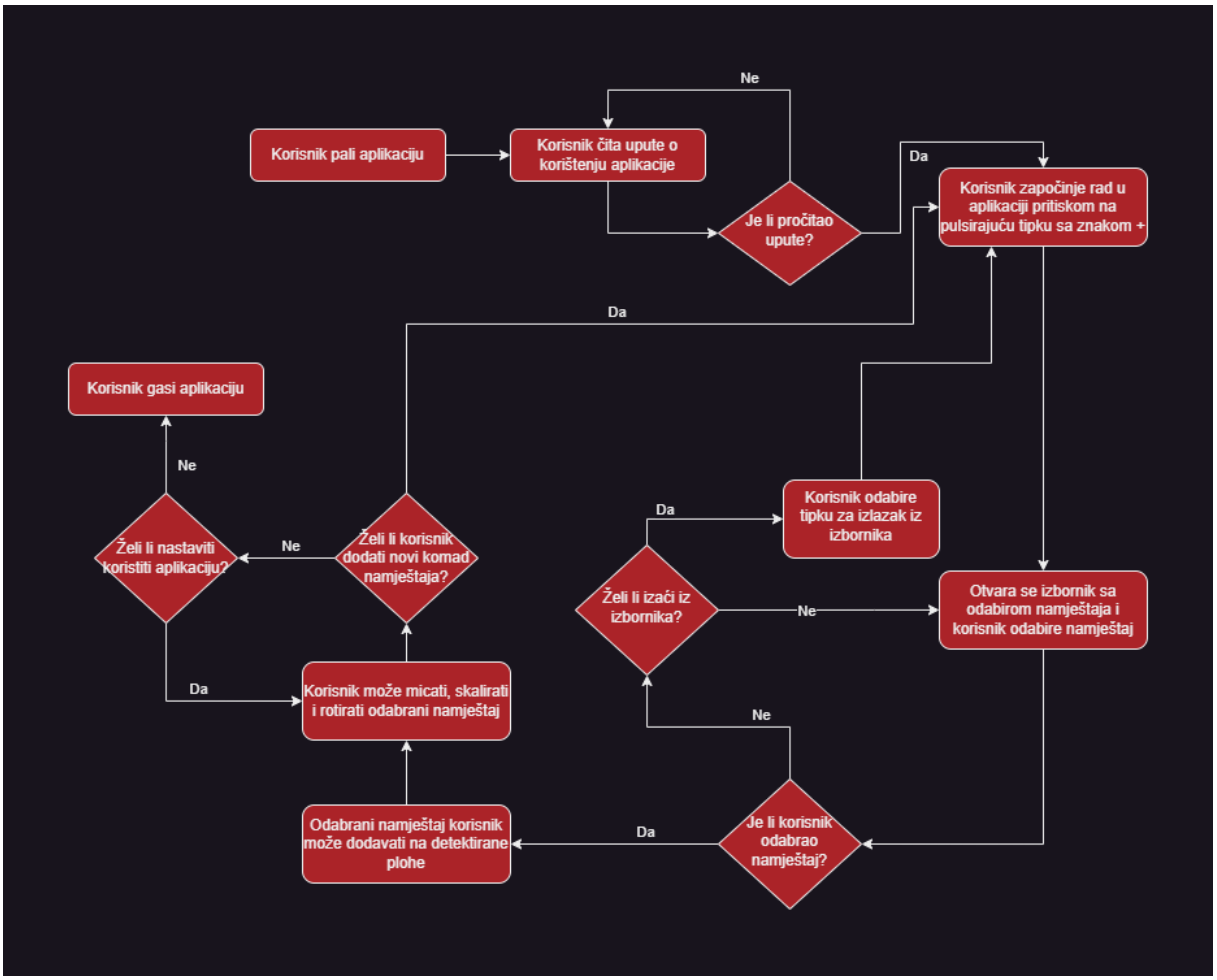
počelo pojavljivati previše modela za vrijeme testiranja pa ih je trebalo na neki način brisat. Dodana je tipka za brisanje odabranog modela. Do tog trenutka model je bio označen samo dok je prst bio na njemu. To je trebalo promijeniti kako bi se objekt mogao brisati, skalirati i rotirati jer je jako nepraktično držati jedan prst kako bi objekt bio selektiran i zatim drugim prstom upravljati *slider*-ima. Trebalo je i nekako istaknuti odabrani model pa je dodan indikator ispod modela. Skaliranje i rotiranje je implementirano pomoću *slider*-a jer je preciznije od korištenja gesta dodira.

Slika 22 prikazuje primitivno korisničko sučelje sa ranije navedenim funkcionalnostima osposobljenima.

Slika 22: Primitivno korisničko sučelje
(Izvor: vlastita izrada)

Nakon što su ranije navedene funkcionalnosti ispravne i testirane razvijeno je finalno korisničko sučelje na temelju skica iz *figme.* Funkcionalnosti su unaprijeđene i prilagođene korisničkom sučelju.

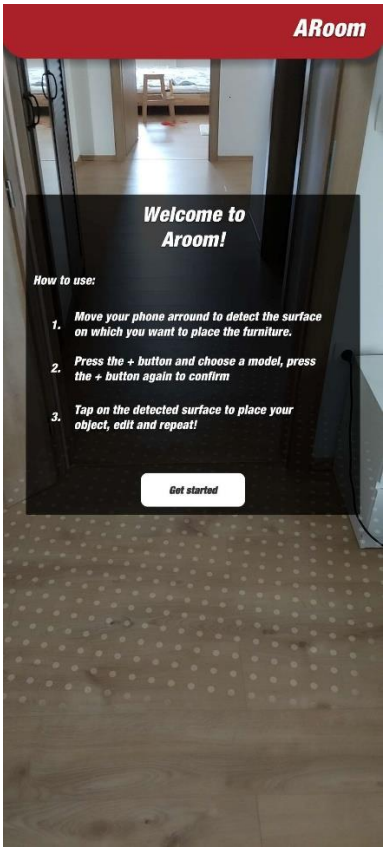
Ostatak poglavlja će detaljno opisivati rad aplikacije kroz korake dijagrama tijeka sa slike 23. Svaki korak će biti detaljno objašnjen, kako su implementirani i opisani dijelovi koda koji se na njega odnose.



Slika 23: Dijagram tijeka aplikacije

(Izvor: vlastita izrada)

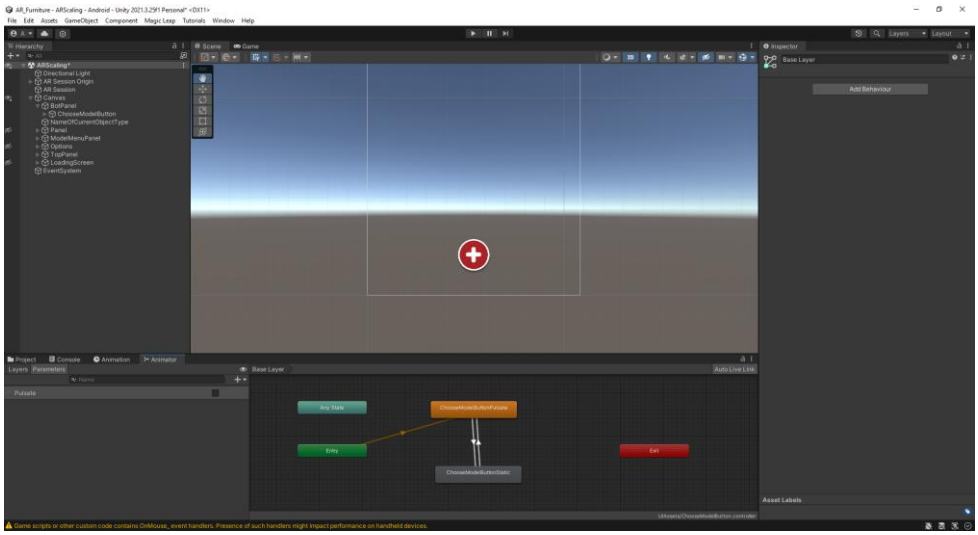
Nakon paljenja aplikacije korisniku se izbacuje prozor sa uputama o korištenju aplikacije koje bi mu trebale biti dovoljne da u potpunosti može koristiti aplikaciju. Slika 24 prikazuje zaslon sa uputama.



Slika 24: Upute u aplikaciji
(Izvor: vlastita izrada)

Prozor koji sadrži upute blokira bilo kakvo registriranje klikova izvan sebe kako bi se izbjeglo slučajno dodavanje 3D modela na plohu. Većina logike se nalazi u skripti *PlacementScript*. Dijelovi skripte će biti prikazivani ovisno o dijelu koda koji je korišten u određenoj funkcionalnosti, blokiranje registriranja klikova izvan okvira je postignuto *if* selekcijom unutar *Update()* metode tako da se kod za provjeru klikova vrti samo ako je prozor sa uputama zatvoren.

Nakon što korisnik zatvori izbornik u donjem dijelu zaslona se pojavljuje crvena tipka sa plusom u sredini i budući da nije odabran nikakav namještaj jer je aplikacija tek upaljena tipka pulsira, signalizirajući korisniku da ju klikne i time otvori izbornik za odabir namještaja. Animacija tipke je kreirana *Unitiy*-jevim ugrađenim animatorom, kao i sve ostale animacije korištene u aplikaciji. Njihovo postavljanje i kreiranje je dijelom rađeno preko skripti, a dijelom preko *Unity*-jevog grafičkog sučelja, prikazano na slici 25.



Slika 25: Editor animacija u *Unitiy*-ju

(Izvor: vlastita izrada)

Klik tipke za otvaranje izbornika registriran je u *Awake()* metodi, to je metoda koja se poziva jednom i to kod prvog pozivanja skripte, postavljanjem *listener*-a, funkcije koja čeka, tj. „sluša“, u ovom slučaju čeka klik. Kod za postavljanje svih *listener*-a i kod za paljenje i gašenje animacije se nalazi ispod.

```
    void Awake()
    {
        objectPrefab = null;

        arRaycastManager = GetComponent<ARRaycastManager>();
        dismissButton.onClick.AddListener(Dismiss);
        aRSessionOrigin = GetComponent<ARSessionOrigin>();
        scaleSlider.onValueChanged.AddListener(ScaleChanged);
        rotationSlider.onValueChanged.AddListener(RotationChanged);
        deselectObjectButton.onClick.AddListener(ToggleOptions);
        removeSelectedButton.onClick.AddListener(RemoveSelected);
        chooseModelButton.onClick.AddListener(OpenModelMenu);


        backButton.onClick.AddListener(removeModel);
        Table1Button.onClick.AddListener(ChangeTable1);
        Table2Button.onClick.AddListener(ChangeTable2);
        Chair1Button.onClick.AddListener(ChangeChair1);
        Chair2Button.onClick.AddListener(ChangeChair2);
        Closet1Button.onClick.AddListener(ChangeCloset1);

        Optionz.SetActive(false);

    }

    private void Dismiss() => welcomePanel.SetActive(false);
```

```
    void Update()
    {
        if (welcomePanel.activeSelf == false)
        {
            botPanel.SetActive(true);
            if (objectPrefab == null)
            {
                chooseModelButton.animator.SetBool("Pulsate", true);
            }
            else
            {
                chooseModelButton.animator.SetBool("Pulsate", false);
            }

            if (EventSystem.current.IsPointerOverGameObject())
                return;
…
```

U ovom isječku se mogu vidjeti i dijelovi koda za gašenje uputa - *welcomePanel.SetActive(false)* i onemogućavanje klikanja iza drugih objekata - *EventSystem.current.IsPointerOverGameObject()*. Animacija gumba se pali ako je *objectPrefab == null* , gdje je *objectPrefab* odabran tip modela za dodavanje.

Listener je postavljen na *onClick()* metodu, što znači da će se metoda (*OpenModelMenu()*) pozvati kada se gumb klikne i otvoriti će se izbornik za odabir namještaja. Slika 26 prikazuje izbornik za odabir namještaja.

Slika 26: Izbornik namještaja

(Izvor: vlastita izrada)

U ovom izborniku korisnik može kliknuti tipku za nazad koja ga vraća na prijašnji zaslon. Može odabrati neki od ponuđenih komada namještaja i kliknuti bijelu tipku sa znakom plus i time selektirati taj namještaj, tada će se izbornik zatvoriti, crvena tipka sa plusom više neće pulsirati i korisnik će dodirom na selektiranu površinu moći dodati odabrani namještaj u prostor. Odabir namještaja instancira *objectPrefab* ako je *null* i mijenja mu kostur (eng. *mesh*) i teksturu. Metode prikazane kodom ispod su korištene za te promjene.

```
    private void ChangeModel(string path)
    {
        ChangeMesh(path);
        ChangeMaterial(path);
        nameOfCurrentObject = path;
        nameOfCurrentObjectType.text = "Currently adding:" + path;
    }
    private void ChangeMesh(string path)
    {
        if (objectPrefab==null)
        {
            objectPrefab = placedPrefab;
```

```
        }
        GameObject newModel = Resources.Load<GameObject>("Meshes/" + path);

        MeshFilter newMeshFilter = newModel.GetComponentInChildren<MeshFilter>();
        Mesh newMesh = newMeshFilter.sharedMesh;

        MeshFilter selectedMeshFilter = objectPrefab.GetComponent<MeshFilter>();
        if (selectedMeshFilter != null)
        {
            selectedMeshFilter.mesh = newMesh;
        }
        else
        {
            Debug.LogWarning("MeshFilter not found on the selected object.");
        }
    }

    private void ChangeMaterial(string path)
    {
        if (objectPrefab == null)
        {
            objectPrefab = placedPrefab;
        }
        GameObject newModel = Resources.Load<GameObject>("Meshes/" + path);

        Renderer newRenderer = newModel.GetComponentInChildren<Renderer>();
        Material newMaterial = newRenderer.material;

        Renderer selectedRenderer = objectPrefab.GetComponent<Renderer>();
        if (selectedRenderer != null)
        {
            selectedRenderer.material = newMaterial;
        }
        else
        {
            Debug.LogWarning("Renderer not found on the selected object.");
        }
    }
```

Datoteke kostura i tekstura se nalaze u *Resources* mapi i preko koda se učitavaju u prethodno postavljeni objekt - *prefab*. Razlog zašto se cijelo vrijeme promjene kostura i tekstura provode na istom *prefab*-u je to što se *prefab* ne može generirati kad je aplikacija pokrenuta i ne mogu mu se dodavati nove skripte. Iz tog razloga je kreiran *prefab* sa dodijeljenim svim bitnim skriptama, poput *BoxColider*-a koja dodaje fiziku objektu i omogućuje njegovu interakciju sa dodirom. Dodijeljeni su i *PlacementObject* te *PlacementBoundingArea* čiji je kod prikazan dolje.

```
using TMPro;
using UnityEngine;

public class PlacementObject : MonoBehaviour
{
    [SerializeField]
    private bool IsSelected;
    [SerializeField]
```

```
    public int uniqueID;
    public float scaleValue;
    public float rotationValue;
    public string nameOfModel;


    public bool Selected
    {
        get
        {
            return this.IsSelected;
        }
        set
        {
            IsSelected = value;
        }
    }

    [SerializeField]
    private TextMeshPro OverlayText;

    [SerializeField]
    private Canvas canvasComponent;

    [SerializeField]
    private string OverlayDisplayText;

    public void SetOverlayText(string text)
    {
        if (OverlayText != null)
        {
            OverlayText.gameObject.SetActive(true);
            OverlayText.text = text;
        }
    }

    void Awake()
    {
        OverlayText = GetComponentInChildren<TextMeshPro>();
        if (OverlayText != null)
        {
            OverlayText.gameObject.SetActive(false);
        }
    }



    public void ToggleOverlay()
    {
        OverlayText.gameObject.SetActive(IsSelected);
        OverlayText.text = OverlayDisplayText;
    }

    public void ToggleCanvas()
    {
        canvasComponent?.gameObject.SetActive(IsSelected);
    }
}
```

PlacementObject skripta pohranjuje identifikator i podatke o skali i rotaciji kako bi se kasnije objekt mogao identificirati, to je potrebno zbog problema koji će u sljedećem poglavlju biti detaljnije opisani.

```
using System.Collections;
using System.Collections.Generic;
using UnityEngine;

[RequireComponent(typeof(PlacementObject))]
public class PlacementBoundingArea : MonoBehaviour
{
    private PlacementObject placementObject;

    private bool initialized = false;

    private GameObject boundingArea;

    [SerializeField]
    private float boundingRadius = 1.0f;

    [SerializeField]
    private Vector3 boundingAreaPosition = Vector3.zero;

    [SerializeField]
    private Material boundingAreaMaterial;

    void Awake()
    {
        SetupBounds();
    }

    void SetupBounds()
    {
        placementObject = GetComponent<PlacementObject>();
        initialized = true;
    }

    void Update()
    {
        if (initialized)
        {
            DrawBoundingArea(placementObject);
            DrawBoundingArea(placementObject.Selected);
        }
    }

    void DrawBoundingArea(bool isActive)
    {
        if (boundingArea == null)
        {
            boundingArea = GameObject.CreatePrimitive(PrimitiveType.Plane);
            boundingArea.name = "BoundingArea";
            boundingArea.transform.parent = placementObject.transform;
            boundingArea.GetComponent<MeshRenderer>().material = boundingAreaMaterial;
        }

        boundingArea.transform.localScale = new Vector3(boundingRadius,
        boundingRadius, boundingRadius);
```

```
        boundingArea.transform.position = placementObject.transform.position +
        boundingAreaPosition;

        boundingArea.SetActive(isActive);
    }
}
```

PlacementBoundingArea se koristi za prikazivanje indikatora o selektiranom objektu, dodjeljuje ga odabranom objektu i briše ga kada je drugi objekt označen. Također je bitno da se indikator briše kod brisanja objekta.

Kada je *prefab* ispravno postavljen izbornik se zatvara i klikom je moguće dodati odabrani namještaj. Logika iza dodavanja objekta klikom je objašnjena još ranije na testnom primjeru. Sljedeći kod prikazuje dodavanje i micanje selektiranog objekta.

```
Touch touch = Input.GetTouch(0);

            touchPosition = touch.position;


            if (touch.phase == TouchPhase.Began && (lastSelectedObject.Selected ==
                                        false || lastSelectedObject == null) )
            {
                lastSelectedObject.Selected = false;
                Ray ray = arCamera.ScreenPointToRay(touch.position);
                RaycastHit hitObject;
                if (Physics.Raycast(ray, out hitObject))
                {
                    lastSelectedObject =
                    hitObject.transform.GetComponent<PlacementObject>();
                    if (lastSelectedObject != null)
                    {

                        PlacementObject[] allOtherObjects =
                          FindObjectsOfType<PlacementObject>();
                        foreach (PlacementObject placementObject in allOtherObjects)
                        {
                            if (placementObject != lastSelectedObject)
                            {
                                placementObject.Selected = false;
                            }


                            else
                            {
                                placementObject.Selected = true;
                                Optionz.SetActive(true);
                                UpdateSliders();
                                UpdateInfo();
                            }
                        }
                    }
                }
            }
        }
```

```
if (arRaycastManager.Raycast(touchPosition, hits,
UnityEngine.XR.ARSubsystems.TrackableType.PlaneWithinPolygon))
{
    Pose hitPose = hits[0].pose;

    if (lastSelectedObject == null)
    {
        lastSelectedObject = Instantiate(objectPrefab, hitPose.position,
        hitPose.rotation).GetComponent<PlacementObject>();
              lastSelectedObject.uniqueID = Random.Range(int.MinValue,
                            int.MaxValue);
        lastSelectedObject.scaleValue = 1;
        lastSelectedObject.Selected = true;
        lastSelectedObject.nameOfModel = nameOfCurrentObject;

        Optionz.SetActive(true);
        UpdateSliders();
        UpdateInfo();
    }

    else
    {
        if (lastSelectedObject.Selected)
        {
               lastSelectedObject.transform.position = hitPose.position;
        }
    }
}
```

Dodavanje funkcionira tako da se instancira novi objekt na temelju *objectPrefab*-a ako je detektiran dodir na plohi i nema označenog objekta. Otvara se izbornik za taj objekt i on postaje označen. Dok je objekt označen dodiri na plohu ne kreiraju nove objekte već mijenjaju lokaciju označenog objekta na lokaciju dodira.

Označavanje objekta počinje ako je detektiran dodir i ako niti jedan objekt nije prethodno označen. Dohvaća se lista svih *placementObject* objekata i pogođeni objekt postaje označen. Otvara se izbornik i *slider*-i se popunjavaju na temelju spremljenih podataka u *placementObject* objektu. Slika 27 prikazuje izbornik odabranog namještaja nakon što je dodan.

Slika 27: Izbornik selektiranog namještaja

(Izvor: vlastita izrada)

Izbornik odabranog modela omogućuje četiri funkcije: rotiranje, skaliranje, odznačavanje i brisanje objekta. Sljedeće metode prikazuju te funkcije.

```
    private void ScaleChanged(float newValue)
    {

        if (applyScalingPerObject)
        {

            lastSelectedObject.transform.localScale = Vector3.one * newValue;
            lastSelectedObject.scaleValue = newValue;
        }
    }

    private void RotationChanged(float newValue)
    {
        lastSelectedObject.transform.rotation = Quaternion.Euler(0, newValue, 0);
        lastSelectedObject.rotationValue = newValue;
    }
    private void RemoveSelected()
    {
        PlacementObject[] allOtherObjects = FindObjectsOfType<PlacementObject>();
        foreach (PlacementObject placementObject in allOtherObjects)
        {
            if (placementObject.uniqueID==lastSelectedObject.uniqueID)
```

```
            {
                Destroy(placementObject.gameObject);
                Optionz.SetActive(false);
            }

        }
    }
    private void ToggleOptions()
    {
        Optionz.SetActive(false);
        lastSelectedObject.Selected = false;

    }
```

ScaleChanged, *RotationChanged* i *ToggleOptions* su jednostavne i iz samih linija koda se može zaključiti što rade bez dodatnog objašnjavanja, *RemoveSelected* dohvaća listu svih objekata tipa *placementObject* i uništava selektirani objekt iz nje. Važno je napomenuti da prije spomenuti indikatori selekcije trebaju biti djeca *placementObject*-a kako bi se uništili skupa sa njim i ne bi ostali siročad.

Time je dijagram tijeka završen. Sada korisnik može ponovno dodavati drugačiji namještaj i njime manipulirati. Video potpunog rada aplikacije moguće je pogledati klikom na link ovjde.

6.3.6. Testiranje funkcionalnosti aplikacije, problemi i izazovi tokom razvoja

Prošlo poglavlje se posvetilo opisom implementacije aplikacije, a u ovom će biti opisani načini testiranja. Također će biti opisani i izazovi tokom razvoja, dijelovi čija implementacija nije bila na prvi pogled jasna. Prijašnje poglavlje, poglavlje implementacije je slijedno opisivalo proces, bez oslanjanja na poteškoće koje su se pojavljivale. Razlog tome je lakše praćenje slijeda razvoja, ali zato postoji ovo poglavlje koje se u potpunosti posvećuje tim dijelovima. Problemima, ozbiljnijim greškama koje su se tokom razvoja aplikacije pojavile i načina na koji su riješeni. Također, važan razlog zašto su problemi i izazovi u istom poglavlju kao i testiranje je taj da se većina problema i izazova uoči tokom testiranja.

Ključne stvari koje je trebalo testirati: dobro detektiranje ploha i stabilnost modela, pravilno ponašanje modela kod njihove manipulacije i pravilno ponašanje korisničkog sučelja. Sljedeća potpoglavlja će se detaljnije posvetiti svakom od tih testiranja i problemima/izazovima koji su se pojavili tokom razvoja i testiranja.

6.3.6.1. Testiranje detekcije ploha i stabilnosti modela

Testiranje detekcije ploha i stabilnosti modela je provedeno na način da se instalirala aplikacija na testni uređaj i testirale su se granične situacije, to su situacije u kojima bi se mogli dogoditi problemi.

Najbitnije testiranje je detekcija ploha, ona je testirana još za vrijeme razvoja testne aplikacije, jer ako ona ne radi daljnji razvoj nije moguć. Također, detekcija je u potpunosti implementirana kroz *Google*-ov *ARCore* i direktno utjecanje na nju nije moguće pa je potrebno rješavati probleme na kreativne načine i zaobilaženje problema gdje je moguće.

Prvi problem kod detekcije ploha je bilo detektiranje „nepoželjnih ploha“. Nepoželjne plohe u sklopu *ARoom*-a su zidovi, ormari, drugi namještaj. U stvari jedina ploha koja se treba detektirati je pod. Iz tog razloga je upaljena samo detekcija horizontalnih ploha. To i dalje ostavlja namještaj poput stola, ali treba uzeti primjenu aplikacije u obzir. Ako korisnik želi vidjeti kako će mu novi stol izgledati pored kauča, pomaknuti će stol izvan vidokruga i onda detekcija plohe stola kao poda neće biti problem.

Drugi problem detekcije ploha je boja plohe, poprilično je veliki problem sa bijelom bojom, taj problem je primijećen kod testiranja detekcije dok je bila upaljena detekcija vertikalnih ploha, bijele zidove aplikacije nije htjela prepoznati, ili je imala dosta problema. Taj problem, na sreću nije toliko strašan u okviru *ARoom*-a, većina podova nije bijele boje, ali određeni uzorci tepiha ili PVC podova znaju stvarati probleme. Taj problem nije trenutno moguće riješiti, ali su maksimalno izbjegnuti slučajevi gdje god je to moguće.

Stabilnost modela s druge strane je cijelo vrijeme bila zadovoljavajuća. Testirana je na prvi pogled „primitivne“, ali efektivne načine. Kretanje oko objekta i to brzo i nepredvidljivo kretanje. Micanjem kamere, okretanjem kamere da objekt više nije u vidokrugu i ponovnim vraćanjem u vidokrug te raznim ekstremnim vrćenjem kamere kako bi se testirale granice.

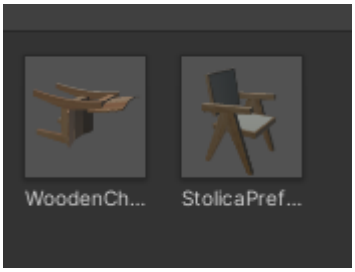
6.3.6.2. Testiranje manipulacije modelima

Pravilno ponašanje modela kod manipulacije je imalo pojedinih problema. Prvobitno je dodano rotiranje i skaliranje gestama prstiju. Taj pristup je bio previše neprecizan pa je implementirano preko *slider*-a, pomoću njih je puno lakše raditi manje izmjene u skali i rotaciji, bez da se korisniku npr. slučajno izmakne prst.

Sljedeći veliki problem se javio kod skaliranja, način skaliranja koji je često korišten sa *AR Foundation*-om je skaliranje cijele scene, to je korisno kada aplikacija zaključi da je detektirana ploha udaljena od kamere i temeljem toga skalira cijelu scenu. Međutim, ako bi se to primijenilo na *slider*-u svi objekti bi se povećali. To je veliki problem ako želimo imati različite veličine objekata. Problem je riješen dodavanjem posebnih identifikatora svakom objektu i implementiranjem zasebnih metoda za skaliranje. Slična stvar se događala i sa rotacijom, gdje bi se svi modeli paralelno rotirali. Identifikatori i zasebne metode rotiranja su riješile i taj problem. Ovaj dio je bio jedan od većih izazova u razvoju aplikacije i puno vremena je uloženo u otkrivanje problema. Ključ u otkrivanju je bila tekstura za detekciju, naime skalirali su se svi objekti i skaliranje je bilo postavljeno na njima pa je sav fokus traženja bio na objektima, a problem je bio širi, vezan uz scenu. Kada bi se skalirali objekti maksimalno, nestali bi a s njima i ploha za detekciju. Trenutak prije nestajanja bi krugovi unutar teksture postali jako veliki i na taj način je fokus prebačen na samu scenu i njezino skaliranje.

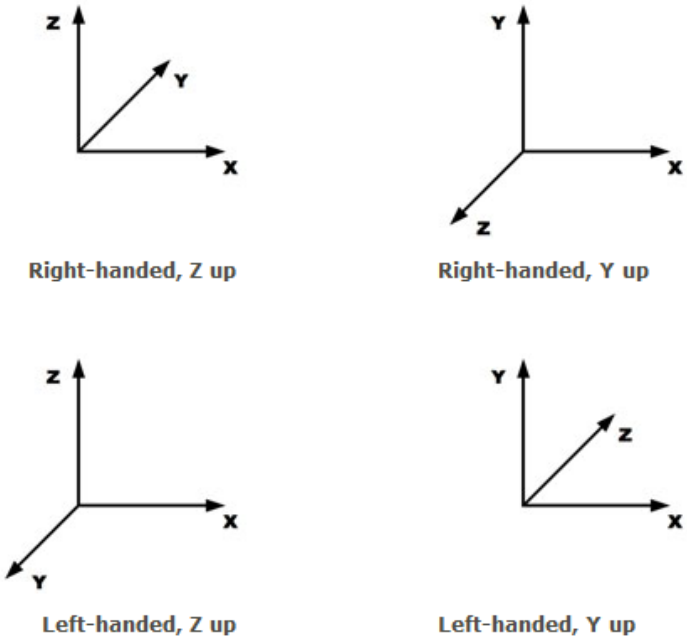
Brisanje modela je radilo dobro dok nisu bili dodani indikatori selekcije. Kada bi se model obrisao indikator bi ostao na zadnjem mjestu gdje se nalazio prije brisanja modela. Taj problem je bio dosta naporan. Budući da se u *Unity*-ju stvari događaju paralelno unutar različitih skripti, ponekad je teško odrediti slijed kako se stvari uistinu izvrše, pogotovo kada je nešto postavljeno preko dviju skripti i treba se dogoditi u isto vrijeme. Takva se situacija dogodila sa objektom i indikatorom, koji je sam po sebi objekt. Prvobitno je logika bila sljedeća: „na pritisak gumba za brisanje se briše objekt i briše se indikator selekcija“. Naime indikator bi stalno ostajao, makar bi se trebao obrisati. Postoji mogućnost brisanja svih objekata u sceni, ali to ne rješava problem, jer je vjerojatno više objekata u sceni koje korisnik želi tamo imati. Postavljanje indikatora kao djeteta modela je riješilo problem jer je onda brisanje roditelja povuklo i brisanje djeteta.

Pravilna promjena modela je također imala svoje probleme. Ti problemi nisu direktno vezani uz programski kod već uz 3D modele, teksture i način na koji se 3D modeli crtaju u *Unity*-ju. Tip podataka korišten za 3D modele u ovom projektu je *.fbx* jer uz kostur nudi i neke metapodatke poput teksture koju je potrebno modelu dodijeliti, korišteni modeli su preuzeti sa web stranice *sketchfab.com* koju i BORM koristi u svojem CAD sustavu. Problemi su se javljali kod početne rotacije modela, stolice bi bile „prevrnute“. Usporedba prikazana na slici 28.



Slika 28: Greška rotacije
(Izvor: vlastita izrada)

Model lijevo je krivo rotiran nakon uvoženja u *Unity*, a model desno nije. Razlog je u alatu u kojem su modeli rađeni, klasični alati za 3D modeliranja poput *blender*-a i *AutoCAD*-a koriste 3D koordinatni sustav po pravilu desne ruke (*Y up*) dok *Unity* koristi po pravilu lijeve ruke i to „*left-handed Y up*“. Slika 29 vizualno prikazuje različite koordinatne sustave.

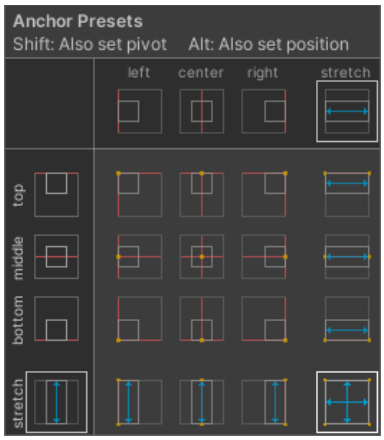


Slika 29: Različiti koordinatni sustavi
(Izvor: https://docs.speedtree.com/doku.php?id=coordinate_systems)

Zbog te razlike modeli ponekad imaju krivu rotaciju. postoji više rješenja, komplicirano rješenje je rotirati modele unutar aplikacije, ali kako znati je li model krivo rotiran? Drugi način je određivanje vlastitog standarda za modele koji se unose u aplikaciju. Određeni standard je da su modeli translatirani u „*left-handed, Y up*“ sustav. *Blender*, nudi opciju izvoza 3D modela u toj verziji koordinatnog sustava i time rješava problem.

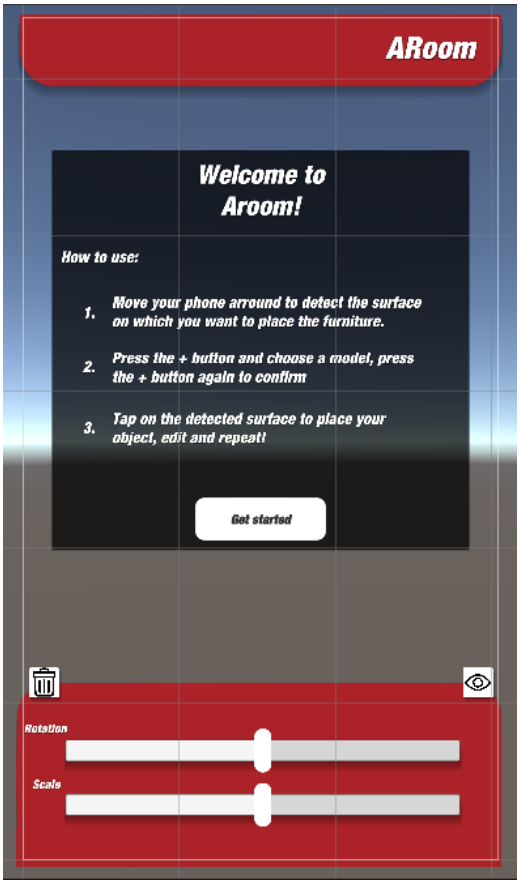
6.3.6.3. Testiranje korisničkog sučelja

Posljednji problem je čest problem kod razvoja mobilnih aplikacija, a to su dimenzije elemenata korisničkog sučelja na različitim ekranima. *Unity* ima sustav sidrenja elemenata korisničkog sučelja za rubove. Prikazano na slici 30.



Slika 30: Izbornik sidrenja u *Unity*-ju
(Izvor: vlastita izrada)

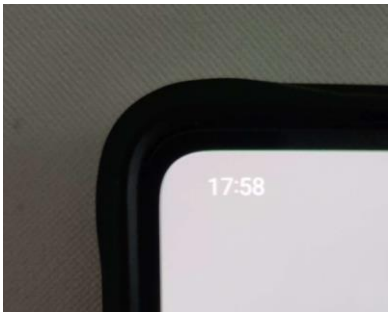
Dobar je, ali je ponekad potrebno element proširit malo izvan rubova zaslona kako bi bolje „sjeo“, što možda na nekim rezolucijama ekrana i dalje neće dobro izgledati. Nemoguće je testirati na svim mogućim rezolucijama pa je problem riješen za rezoluciju mobitela na kojem je testiran, A52s 5G rezolucije 1080 x 2400 piksela. I dalje rješenje nije savršeno, makar su točne brojke rezolucije postavljene korisničko sučelje nije išlo do ruba već je postojala polu-milimetarska crta, koja je vidljiva jer je u pozadini kamera i kad god se kamera micala ta crta bi bila vidljiva. Najlakše rješenje tog problem je bilo povećati dimenzije korisničkog sučelja da budu preko okvira, na taj način postoji „prostor za grešku“ kod drugih zaslona. Slika 31 prikazuje korisničko sučelje u aplikaciji i u razvojnom okruženju.



Slika 31: Korisničko sučelje u *Unity*-ju

(Izvor: vlastita izrada)

Bijeli okvir prikazuje dimenzije zaslona rezolucije 1080 x 2400 piksela. Ta rezolucija nije ručno kodirana, i dalje se koriste sidra i vezana su za okvir kako bi prividno prikazala kako bi sučelje izgledalo na uređaju takvih dimenzija zaslona. Tu se javlja problem zakrivljenosti rubova ekrana mobitela. Prikazano na slici 32. I njega je potrebno uračunati kod povećanja veličine zaslona izvan okvira u razvojnom okruženju jer je u *Unity*-ju taj okvir pravokutan.



Slika 32: Zakrivljenost ruba ekrana

(Izvor: vlastita izrada)

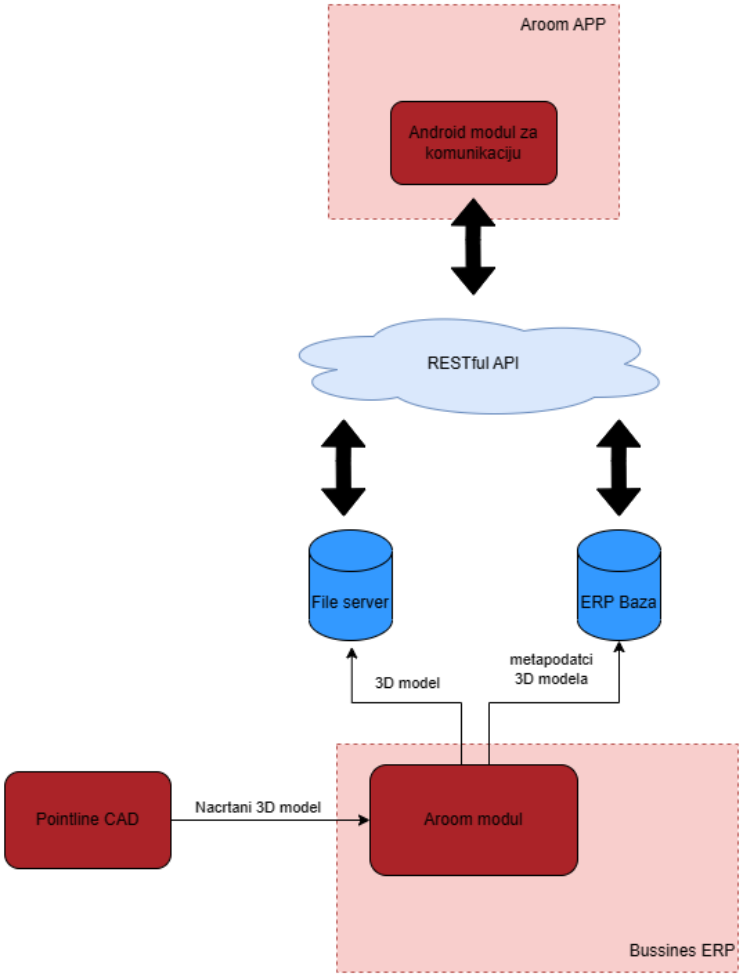
7. Koncept integracije mobilne aplikacije u ERP sustav

Aplikacija je razvijana sa različitim načinima integracije u ERP sustav na umu. Ovo poglavlje će se baviti opisivanjem potencijalnih načina integracije. Integracija je u radu obrađena samo teoretski, budući da se svo nadograđivanje BORM Business ERP sustava radi „*in-house*“ kanali za komunikaciju bi trebali biti napravljeni od strane njihovih developera. Za vrijeme pisanja rada BORM-ove aplikacije prolaze kroz veliki vizualni redizajn pa je odlučeno da će integracija biti teoretska i potencijalno u budućnosti implementirana u sustav.

7.1. Arhitektura sustava

BORM Business implementacije ERP sustava lokalno i u oblaku, a *ARoom* može biti implementiran u oba slučaja. Način na koji bi se *ARoom* povezao sa Business ERP sustavom je sljedeći. Aplikacija treba nekako dobiti 3D modele namještaja koje poduzeće koje koristi ERP sustav proizvodi/prodaje. ERP sustavi aplikacija postoje. Potrebno je kreirati server gdje će se pohranjivati modeli, tablicu u bazi za pohranjivanje metapodataka o 3D modelima, komunikacijski kanal (npr. *RESTful API*) i osigurati sigurnosnu razinu (npr. *OAuth*).

Sljedeći dijagram prikazuje arhitekturu sustava.



Slika 33: Arhitekture sustava
(Izvor: vlastita izrada)

Nakon što korisnik nacrta 3D model namještaja unosi ga u ERP sustav, gdje se preko *ARoom* modula model pohranjuje na server, a u bazu ERP sustava se pohranjuju metapodatci o modelu, koji uključuju i URL pohranjenog modela na serveru. Nakon toga pomoću RESTful API-a komuniciraju aplikacija i baza/server, aplikacija šalje zahtjev preko REST-a i dobiva model koji kasnije prikazuje u ERP sustavu.

Arhitektura izgleda isto u slučaju da je ERP sustav lokalan ili u oblaku. Ako je u pitanju lokalni ERP sustav, server za pohranu podataka i baza ERP sustava se nalaze na *BormBox*-u; mini serveru koji BORM nudi u svojim paketima i omogućuje instalaciju svih svojih usluga na njega. Ako se korisnik odluči za varijantu u oblaku usluge su pohranjene na BORM-ovom okruženju u oblaku.

7.2. Problemi integracije

Budući da praktična implementacija trenutno nije moguća ovo potpoglavlje će opisati probleme integracije, što je potrebno promijeniti i/ili kreirati kako bi se ARoom mogao implementirati u BORM Business ERP sustav. Arhitekturom je opisano što je potrebno za integraciju u sustav.

Najbolje je početi od PointLine CAD sustava. On ima mogućnost izvoza u različitim tipovima podataka, ali najnovija produkcijska verzija nema sve potrebne tipove podataka potrebne za aplikaciju, poput *.obj* i *.fbx*. Izvoz takvih tipova podataka je prisutan u testnim verzijama PointLine CAD sustava, potrebno ga je usavršiti i staviti na produkcijsku verziju. Druga opcija je prilagoditi aplikaciju za veći broj tipova podataka.

Sljedeći sustav je Business ERP. Modul za proširenu stvarnost ne postoji pa ga je potrebno kreirati. Budući da je rad u Business ERP sustavu projektno orijentiran on treba moći iz trenutnog projekta izvući 3D modele i prenijeti ih u modul. Modul zatim nudi mogućnost unosa projektnih modela u bazu. Ako se modulu pristupa bez projekta trebao bi omogućiti uređivanja, brisanja ili dodavanja modela ručno u bazu.

Direktno na Business ERP se vežu baza podataka i server sa podatcima koji postoje, u bazi je potrebno kreirati tablicu sa metapodatcima o modelu. Server sa podatcima također postoji samo je potrebno ograničiti prostor da se ne prepuni 3D modelima, čija veličina može biti poprilično velika.

Komunikacijski kanal je potrebno kreirati pomoću RESTful API-a kojeg BORM koristi za komunikaciju drugih aplikacija sa ERP sustavom tako da im kreiranje takvog kanal nije problem i može se računati kao spremno.

Posljednji dio je kreiranje modula za spajanje na BORM-ov komunikacijski kanal unutar mobilne aplikacije. Od svih gore navedenih problem integracija ovaj dio je najzahtjevniji jer ostali su već na slične načine implementirani unutar raznih BORM-ovih aplikacija. Aplikacija je rađena modularno tako da neće biti potrebne veće izmjene unutar aplikacije. Najveći izazove bi stvaralo kreiranje modula za komunikaciju.

7.3. Monetizacija

Postoji više načina na koji se modul može monetizirati. Najpoželjniji je prodaja u sklopu ERP sustava. Na taj način BORM prodaje većinu svojih modula. Postoje različite razine usluge u BORM Business ERP sustavu i ova aplikacija bi proširila njegov portfelj modulom koji drugi ERP sustavi ne nude.

Drugi način, koji se može kombinirati sa prvim je prodaja usluge držanja modela unutar aplikacije. Ova verzija bi zahtijevala neke promjene unutar aplikacije. Aplikacija bi postala tržište 3D modelima gdje bi poduzeća koja žele imati svoje proizvode vidljive u aplikaciji platila pretplatu i njihovi modeli bi postali dio dostupnih modela unutar aplikacije. Kada bi korisnik isprobao namještaj u željenom prostoru i odlučio se za kupnju aplikacija bi ga prespojila na stranicu poduzeća gdje je moguće kupiti željeni namještaj.

Treći način je integracije aplikacije kao modula unutar postojeće aplikacije klijenta. Poduzeća koja već imaju razvijenu aplikaciju često žele da sve funkcionalnosti budu unutar nje, umjesto da krajnji korisnik, kupac namještaja, mora skidati još jednu aplikaciju kako bi mogao vidjeti namještaj. Ovaj način integracije bi zahtijevao najviše promjena i mogao bi stvarati probleme kod ažuriranja pa ga je najbolje izbjeći, ako je to moguće.

7.4. Mjerenje uspješnosti

Ovakva vrsta aplikacije obogaćuje uslugu koja već postoji i stoga je jako bitno odrediti dobar način mjerenja. Aplikacija obogaćuje prodaju na način da krajnjem korisniku nudi pomoću u odlučivanju.

Jedan od načina mjerenja uspješnosti je kroz povećanje prodaje, ako je integracija aplikacije u ERP sustav jedina promjena koja se dogodila i prodaja se poveća moguće je pretpostaviti da je razlog povećanja integracija aplikacije. To je dobar pokazatelj, ali nije bez svojih mana. Naime prodaja je mogla porasti iz drugih, eksternih razloga koji su izvan kontrole poduzeća i ERP sustava i time bi mogla dati krivu informaciju.

Bolji način mjerenja uspješnosti zahtijeva preusmjeravanje iz aplikacije u neku vrstu internetske trgovine ili druge platforma za kupovinu željenog namještaja, na taj način se može zabilježiti koje su prodaje pristigle iz *ARoom* aplikacije, a koje direktno iz internetske trgovine. Ni taj način nije savršen jer korisnik može ući u internetsku trgovinu preko aplikacije, izići i ne

kupiti, vratiti se kasnije direktno preko internetske trgovine i kupiti. Je li *ARoom* pomogao u toj situaciji ili nije? Vjerojatno je, ali sustav to neće registrirati.

To su samo neki od jednostavnih načina mjerenja uspješnosti i uvijek je moguće kreirati bolja i sveobuhvatnija rješenja. Kakva god ona bila u sebi će do neke mjere ukomponirati gore navedena dva prijedloga.

7.5. Moguće nadogradnje aplikacije

Aplikacija trenutno prikazuje gotove 3D modele. BORM je poznat po tome što nudi izradu namještaja po mjeri i njegovi sustavi su prilagođeni takvim poduzećima. Najznačajnija nadogradnja koja bi se mogla napraviti u aplikaciji je mogućnost uređivanja namještaja; dimenzija, boja, materijala. Uređivanje boja i materijala je moguće. Logika se zapravo već nalazi unutar aplikacije, primijenili bi se isti principi kao i kod zamjene teksture.

Uređivanje dimenzija na razini koja bi bila dovoljno precizna trenutno nije moguće. Uređivanje dimenzija je djelomično implementirano u aplikaciji putem skaliranja, ali pravo uređivanje dimenzija, poput onog u PointLine CAD sustavu pomoću parametarskih datoteka, nije dostupno. Za tu vrstu uređivanja nije dovoljno samo rastegnuti model, potrebno je urediti *mesh*, a to *Unity* sam po sebi ne omogućuje i bilo bi potrebno napraviti vlastiti *engine* koji bi služio uređivanju dimenzija 3D modela.

Drugi problem koji se javlja je problem preciznosti, najvažnija stvar kod namještaja po mjeri je preciznost. Proširena stvarnost još uvijek nije dovoljno precizna da bi se namještaj mogao 99% precizno uređivati pomoću kamere. Mjerenje pomoću proširene stvarnosti je moguće i testirano za vrijeme razvoja ARoom-a, ali je preciznost uvelike varirala. Postoje aplikacije na tržištu koje jako precizno mjere udaljenosti i dimenzije objekata pomoću proširene stvarnosti, ali za to koriste dodatne *computer vision* algoritme i strojno učenje kako bi nadoknadili nedostatke preciznosti proširene stvarnosti. Pa bi i njih trebalo ukomponirati u nadograđivanju aplikacije.

7.6. Mogućnost integracije u druge ERP sustave

Makar je *ARoom* aplikacija zamišljena kao dio BORM Business ERP paketa, ne znači da njezina integracija nije moguća u druge ERP sustave.

Ranije u radu bio je prikazan primjer SAP-ovog *XR Cloud-a* i *Unity Integration Toolkit*-a u proizvodnji. Te tehnologije bi se mogle povezati i sa *ARoom* mobilnom aplikacijom. *ARoom* je također napravljen u *Unity* programskom okruženju i umjesto kreiranja vlastitog modula za komunikaciju spajanje sa *SAP XR Cloud*-om bi se moglo izvesti preko *Unity Integration Toolkit*-a. Mobilna aplikacije je prvobitno namijenjena namještaju, ali funkcionira sa bilo kojom vrstom 3D modela tako da bi prilagodba za druge industrije bila moguća.

Drugi ERP sustavi koji nemaju posebna okruženja za razvoj komunikacijskih kanala poput SAP-a nisu isključena za povezivanje sa *ARoom*-om. Njihovo spajanje je moguće napraviti na isti način na koji bi spajanje sa BORM Business ERP sustavom bilo realizirano. Kreiranjem modula u ERP sustavu, neke vrste komunikacijskog kanala i pohrane modela/podataka te modul unutar *ARoom* mobilne aplikacije prilagođen za spajanje tom sustavu.

Primjer univerzalne implementacije *ARoom* mobilne aplikacije je sljedeći. Bilo koji ERP sustav koji se želi spojiti na *ARoom* mobilnu aplikaciju treba imati kreiran komunikacijski kanal pomoću kojeg bi se mobilna aplikacija povezala sa ERP sustavom. Unutar ERP sustava može postojati zaseban modul posvećen *ARoom*-u, ali i ne treba, ovisno o načinu na koji bi aplikacija bila implementirana. Standardni način, i ujedno način s kojim je aplikacija zamišljena je taj da se preko aplikacije pošalju podatci u ERP sustav i generira prodajni nalog. U aplikaciji trebaju biti postavljeni modeli i metapodatci, način na koji će doći do aplikacije nije bitan, bitno je što *ARoom* dalje radi sa tim modelima i podatcima. Korisnik aplikacije odabere željene modele namještaja i kada je zadovoljan *ARoom* ga može preusmjeriti u internet trgovinu gdje je forma za kupovinu automatski popunjena ovisno o korisnikovom odabiru i dalje se prosljeđuje ERP sustavu koji kreira prodajni nalog. Komunikacija između ERP sustava i internet trgovine je specifična ERP sustavu, bitno je da podatci stignu iz *Aroom*-a do ERP sustava.Ostatak procesa isti kao da je krajnji korisnik naručio namještaj bez *Aroom*-a. To bi bio primjer najopćenitije primjene aplikacije, kostur. Daljnje modificiranje prilagodbe su moguće, ovisno o potrebama ERP sustava u koji bi aplikacija bila implementirana.

8. Zaključak

Konstantno kreiranje dodane vrijednosti ERP sustavu je ključno za njegovu diverzifikaciju na tržištu, zadržavanje klijenata i privlačenje novih. Ovaj rad bavio se razvojem aplikacije koja bi obogatila ERP sustav i stvorila mu dodanu vrijednost.

Kako bi se čitatelju bolje približio proces odluke o vrsti aplikacije teoretski dio rada opisuje pojmove poput modernog ERP sustava, inteligentnog ERP sustava, industrije 4.0 i njezinih tehnologija sa naglaskom na proširenu stvarnosti i načina kako stvoriti dodanu vrijednost ERP sustavu tim tehnologijama.

Aplikacija je osmišljena kao sustav koji kupcima nekog namještaja nudi pomoć u odlučivanju kroz prikaz namještaja u stvarnom prostoru koristeći proširenu stvarnost. Zbog prirode proširene stvarnosti aplikacija je mobilna, primarno namijenjena mobilnim telefonima operacijskog sustava *Android*. Praktični dio rada prikazuje detaljan opis razvoja aplikacije kroz okvir osobnog ekstremnog programiranja, od planiranja, dizajna, razvoja i testiranja do problema koji su se javljali tokom razvoja i načina na koji su oni riješeni.

Na kraju rada su objedinjena sva prijašnja poglavlja u opisu koncepta integracije aplikacije u BORM Business ERP sustavu. Opisom arhitekture, mogućih varijanti aplikacije, monetizacije, mogućim nadogradnjama unutar aplikacije i potencijalnim proširivanjem na druge ERP sustave. Navedene su funkcionalnosti i unaprjeđenja za budući nastavak razvoja aplikacije. Ovaj rad prikazuje jedan potencijalan način stvaranja dodane vrijednosti ERP sustavu, ali predstavlja i predložak pomoću kojeg se mogu osmisliti slični načini kreiranja dodane vrijednosti kroz druge vrste aplikacija.

Popis literature

[1] Picek, R. (2023) Abeceda ERP sustava Preuzeto 15.8.2023. s https://elfarchive2223.foi.hr/mod/folder/view.php?id=96269

[2] Rawat, C. (2023). Role of ERP modernization in digital transformation: PeopleSoft insight. Preuzeto 7.9.2023. s https://doi.org/10.48550/arxiv.2303.03224

[3] Tomasz, T., Damian (2018)., D.:The ERP process system as a direction of the evolution of integrated management information systems. Informatyka Ekonomiczna Bus. Inf. 3(49), 118–131 (2018). The New Generation of ERP in the Era of Artificial Intelligence and Industry 4.0 Preuzeto 15.8.2023. s https://link.springer.com/chapter/10.1007/978-3-030-90633-7_96

[4] Picek, R., & Androcec, D. (2022). IMPACT OF DIGITAL TRANSFORMATION TECHNOLOGIES ON ERP SYSTEMS. Economic and Social Development: Book of Proceedings, 50-57. Preuzeto 19.8.2023. s https://www.esd-conference.com/upload/book_of_proceedings/Book_of_Proceedings_esdLisbon2022_Online.pdf

[5] Jayender, P., & Kundu, G. K. (2021). Intelligent ERP for SCM agility and graph theory technique for adaptation in automotive industry in india. International Journal of System Assurance Engineering and Management, https://doi.org/10.1007/s13198-021-01361-y

[6] Jenab, K., Staub, S., Moslehpour, S., & Wu, C. (2019). Company performance improvement by quality based intelligent-ERP. Preuzeto 11.9.2023. s https://scholarworks.moreheadstate.edu/msu_faculty_research/902/

[7] Bouchemal, N., & Bouchemal, N. (2019). Intelligent ERP based multi agent systems and cloud computing. In Machine Learning for Networking: First International Conference, MLN 2018, Paris, France, November 27–29, 2018, Revised Selected Papers 1 (pp. 378-386). Springer International Publishing. Preuzeto 11.9.2023. s https://link.springer.com/chapter/10.1007/978-3-030-19945-6_27

[8] Sathiyamoorthi, V., & Bhaskaran, V. M. (2009). Data mining for intelligent enterprise resource planning system. International Journal of Recent Trends in Engineering, 2(3), 1. Preuzeto 11.9.2023. s https://citeseerx.ist.psu.edu/document?repid=rep1&type=pdf&doi=35a101070bbf131cc93de2b5cb91c862f09beab6

[9] Amini, M., & Abukari, A. M. (2020). ERP systems architecture for the modern age: A review of the state of the art technologies. Journal of Applied Intelligent Systems and

Information Sciences, 1(2), 70-90. Preuzeto 11.9.2023. s https://journal.research.fanap.com/article_111141.html

[10] Špatenka, J., & Koch, M. (2021). Sustainable development of companies using the ERP system as a fundamental tool of digital transformation. Trendy Ekonomiky a Managementu, 15(38), 61-70. https://doi.org/10.13164/trends.2021.38.61

[11] Jayender, P., & Kundu, G. K. (2022). Big data, IOT, ERP interoperability - an intelligent SCM decision system. Paper presented at the 549-555. Preuzeto 11.9.2023. s https://doi.org/10.1109/ICAIS53314.2022.9742745

[12] Akturk, C. (2021). Artificial intelligence in enterprise resource planning systems: A bibliometric study. Journal of International Logistics and Trade, 19(2), 69-82. https://doi.org/10.24006/JILT.2021.19.2.069

[13] sap.com (2023) What is industry 4.0 Preuzeto 20.8.2023. s https://www.sap.com/

[14] Yathiraju, N. (2022). Investigating the use of an Artificial Intelligence Model in an ERP Cloud-Based System. International Journal of Electrical, Electronics and Computers, Preuzeto 22.8.2023. s https://aipublications.com/uploads/issue_files/IJEEC-APR20222-Investigating.pdf

[15] techtarget.com (2023) What is the internet of things (IoT)? Preuzeto 22.8.2023. s https://www.techtarget.com/iotagenda/definition/Internet-of-Things-IoT

[16] Paththinige, P., Thilakarathne, K., Rathnasekara, T., Wickramaarachchi, R., & Withanaarachchi, A. (2022, September). Examine the Impact of IoT for Supply Chain-Based Operations in ERP Systems: Systematic Literature Review. In 2022 International Research Conference on Smart Computing and Systems Engineering (SCSE) (Vol. 5, pp. 344-350). IEEE. Preuzeto 25.8.2023. s https://ieeexplore.ieee.org/abstract/document/9905098

[17] Carmigniani J, Furht B, Anisetti M, Ceravolo P, Damiani E, Ivkovic M (2011) Augmented reality technologies, systems and applications. Preuzeto 25.8.2023. s https://www.csd.uoc.gr / hy469/files/panels/Augmented_reality_technologies_systems_and_applications.pdf

[18] Craig, A. B. (2013). Understanding augmented reality: Concepts and applications. Elsevier Science. Preuzeto 8.9.2023. s https://doi.org/10.1016/C2011-0-07249-6

[19] Maheshwari, A. (2019). Digital transformation : Building intelligent enterprises. John Wiley & Sons, Incorporated. Preuzeto 8.9.2023. s https://ebookcentral.proquest.com/lib/uzuccs/reader.action?docID=5891700#

[20] Panorama Consulting Group (2023) 2023 Clash of the Titans SAP vs. Oracle vs. Microsoft vs. Infor, Preuzeto 28.8.2023. s https://www.panorama-consulting.com/resource-center/clash-of-the-titans-sap-vs-oracle-vs-microsoft-vs-infor/

[21] countingup.com (2023) What is added value in business? Preuzeto 29.8.2023. s https://countingup.com/resources/what-is-added-value-in-business/

[22] Jajic, I., Khawaja, S., Hussain Qureshi, F., & Pejić Bach, M. (2022). Augmented reality in business and economics: Bibliometric and topics analysis. Interdisciplinary Description of Complex Systems, 20(6), 723-744. Preuzeto 8.9. s https://doi.org/10.7906/indecs.20.6.5

[23] Kaplan, A. D., Cruit, J., Endsley, M., Beers, S. M., Sawyer, B. D., & Hancock, P. A. (2021). The effects of virtual reality, augmented reality, and mixed reality as training enhancement methods: A meta-analysis. Human Factors, 63(4), 706-726. Preuzeto 8.9.2023. s https://doi.org/10.1177/0018720820904229

[24] Wei, W. (2019). Research progress on virtual reality (VR) and augmented reality (AR) in tourism and hospitality. Journal of Hospitality and Tourism Technology, 10(4), 539-570. Preuzeto 8.9.2023. s https://doi.org/10.1108/JHTT-04-2018-0030

[25] traveltomorrow.com (2023) Singapore Tourism Bord partners with Google to create Augmented Reaulty tours Preuzeto 8.9.2023. s https://traveltomorrow.com/singapore-tourism-board-partners-with-google-to-create-augmented-reality-tours/

[26] Mourtzis, D., Zogopoulos, V., & Xanthi, F. (2019). Augmented reality application to support the assembly of highly customized products and to adapt to production re-scheduling. The International Journal of Advanced Manufacturing Technology, 105, 3899-3910. Preuzeto 11.9.2023. s https://link.springer.com/article/10.1007/s00170-019-03941-6

[27] Wang, W., Wang, F., Song, W., & Su, S. (2020). Application of augmented reality (AR) technologies in inhouse logistics. In E3S Web of Conferences (Vol. 145, p. 02018). EDP Sciences. Preuzeto 11.9.2023. s https://www.e3s-conferences.org/articles/e3sconf/pdf/2020/05/e3sconf_iaecst2020_02018.pdf

[28] Bulearca, M., & Tamarjan, D. (2010). Augmented reality: A sustainable marketing tool. Global business and management research: An international journal, 2(2), 237-252. Preuzeto 11.9.2023. s https://books.google.hr/books?hl=en&lr=&id=PkvjLMfrtrUC&oi=fnd&pg=PA237&dq=augmented+reality+applications+in+marketing&ots=QRfr02xzYV&sig=MQmorjppIAQoH8dOwFpPn2_RWIc&redir_esc=y#v=onepage&q=augmented%20reality%20applications%20in%20marketing&f=true

[29] Mekni, M., & Lemieux, A. (2014). Augmented reality: Applications, challenges and future trends. Applied computational science, 20, 205-214. Preuzeto 11.9.2023. s https://www.researchgate.net/profile/Nirmala-Svsg/post/What_are_the_possibilities_of_Augmented_reality_to_enhance_construction_automation/attachment/59d6328c79197b8077990458/AS%3A370944288804866%401465451418864/download/ACACOS-29.pdf

[30] immersivelearning.news (2023) Case study: SAP Integrates AR for ERP Preuzeto 8.9.2023 s https://www.immersivelearning.news/2021/02/26/case-study-sap-integrates-ar-for-erp/

[31] softwebsolutions.com (2023) Here's why augmented reality in warehouse management is creating a breakthrough Preuzeto 8.9.2023. s https://www.softwebsolutions.com/resources/ar-warehouse-transforming-supply-chain-with-ar-logistics.html

[32] borm-software.com (2023) Borm : ERP-Systeme und CAD-Software für Unternehmen Preuzeto 29.8.2023. s https://www.borm-software.com/

[33] Dzhurov, Y., Krasteva, I., & Ilieva, S. (2009). Personal Extreme Programming–An Agile Process for Autonomous Developers. Preuzeto 9.9.2023. s https://core.ac.uk/display/213561553?utm_source=pdf&utm_medium=banner&utm_campaign=pdf-decoration-v1

[34] bankmycell.com (2023) How Many People Have iPhones in 2023? Preuzeto 1.9.2023. s https://www.bankmycell.com/blog/number-of-iphone-users

[35] unity.com (2023) Unity Real-Time Development Platform. Preuzeto 2.9.2023. s

https://unity.com/our-company

[36] docs.unity3d.com (2023) UnityEngine.XR.ARFoundation Preuzeto 4.9.2023. s https://docs.unity3d.com/Packages/com.unity.xr.arfoundation@1.0-preview.8/api/UnityEngine.XR.ARFoundation.ARSessionOrigin.html

Popis slika

Slika 1: Podjela tržišta raznih industrija ovisno o njihovoj veličini 12
Slika 2: Arhitektura proširene stvarnosti 14
Slika 3: Koncept zaslona postavljenog na glavu 18
Slika 4: HoloLens 18
Slika 5: Klaster analiza 20
Slika 6: Proširena stvarnost u zdravstvu 22
Slika 8: SAP ERP i AR 27
Slika 10: BORM Business ERP 30
Slika 11: Planiranje resursa 30
Slika 12: Pointline 2D crtanje 32
Slika 13: Pointline 3D crtanje 32
Slika 14: Faze razvoja u PXP metodici 34
Slika 15: Unity razvojno okruženje 37
Slika 16: Ručna skica aplikacije 38
Slika 17: Skica početnog zaslona u alatu *Figma* 40
Slika 18: Skica odabira namještaja u alatu *Figma* 41
Slika 19: *Unity* hijerarhija 42
Slika 20: *Unity* inspektor 43
Slika 21: Tekstura detekcije ploha 43
Slika 22: Primitivno korisničko sučelje 45
Slika 23: Dijagram tijeka aplikacije 46
Slika 24: Upute u aplikaciji 47
Slika 25: Editor animacija u *Unitiy*-ju 48
Slika 26: Izbornik namještaja 50
Slika 27: Izbornik selektiranog namještaja 56
Slika 28: Greška rotacije 60
Slika 29: Različiti koordinatni sustavi 60

Slika 30: Izbornik sidrenja u *Unity*-ju 61
Slika 31: Korisničko sučelje u *Unity*-ju 62
Slika 32: Zakrivljenost ruba ekrana 62
Slika 33: Arhitekture sustava 64