



Satakunnan ammattikorkeakoulu
Satakunta University of Applied Sciences

TONI SEESSALO

Visual Components -simulointi- ympäristön mahdollisuudet hyvinvointiteknologian sovellusten simuloinnissa

SÄHKÖ- JA AUTOMAATIOTEKNIIKAN
TUTKINTO-OHJELMA
2023

| | | |
|---|-------------------------------------|----------------------------|
| Tekijä(t) Seessalo, Toni | Julkaisun laji Opinnäytetyö, AMK | Päivämäärä Kesäkuu 2023 |
| | 26 | Julkaisun kieli Suomi |
| Julkaisun nimi Visual Components -simulointiympäristön mahdollisuudet hyvinvointiteknologian sovellusten simuloinnissa | | |
| Tutkinto-ohjelma Sähkö- ja automaatiotekniikan tutkinto-ohjelma | | |
| <p>Tässä opinnäytetyössä selvitettiin Visual Components -simulointiympäristön mahdollisuuksia hyvinvointiteknologian sovellusten simuloinnissa. Selvitys toteutettiin tekemällä saman skenaarion simulointi kahdella eri tavalla. Ensin käytettiin Visual Componentsin omia toimintoja ja seuraavaksi sama sovellus tehtiin käyttämällä itse luotuja toimintoja. Toiminnot tehtiin Python3-ohjelmointikielellä Visual Componentsin koodieditorissa. Näin Visual Componentsin mahdollisuudet ovat selkeästi arvioitavissa.</p> <p>Tämä opinnäytetyö toteutettiin Satakunnan ammattikorkeakoulun ja Tampereen yliopiston toteuttamalle, Suomen Akatemian rahoittamalle, Kriittisten terveydenhuoltoympäristöjen toiminnan varmistaminen modernin teknologian avulla - RoboRFID-hankkeelle. RoboRFID-hankkeen yhtenä osana oli luoda skenaario, jossa mobiilirobotti auttaa esimerkiksi yöaikaan palvelukodin yleisiin tiloihin eksynyttä muistisairasta asukasta löytämään takaisin omaan huoneeseensa.</p> <p>Visual Components on eri teollisuuden tehtaiden ja niiden osien suunnitteluun erikoistunut tietokonesimulointiohjelmisto. Sen mahdollisuuksia hyvinvointiteknologian sovellusten simulointiin ei ole kuitenkaan paljoakaan aiemmin tutkittu.</p> <p>Simuloitaville skenaarioille laadittiin toiminnan kuvaus ja tavoitteet. Niiden mukaan mobiilirobotti suorittaa sen normaaleja logistisia tehtäviä palvelutalon yleisissä tiloissa, joissa muistisairas asukas samanaikaisesti kävelee eksyneenä. Mobiilirobotin ja asukkaan kohdatessa mobiilirobotti tunnistaa asukkaan ja saattaa hänet takaisin omaan huoneeseensa. Tämän jälkeen mobiilirobotti jatkaa logistisissa tehtävissään.</p> <p>Vain Visual Componentsin omia toiminnallisuuksia käyttämällä skenaarion simulaatio ei ole tehtävissä. Itse luotuja toimintoja käytettäessä skenaarion simulaation tavoitteet kuitenkin täyttyvät ja toimintojen monipuolisuus riippuu vain kehittäjän ohjelmointitaidoista. Visual Components soveltuu hyvinvointiteknologian sovellusten simulointiin edellyttäen kehittäjältään vahvaa ohjelmointiosaamista, jolloin iso osa Visual Componentsin omista toiminnallisuuksista jää hyödyntämättä.</p> | | |
| Avainsanat mobiilirobotiikka, simulointi, hyvinvointiteknologia | | |

| | | |
|---|--|-------------------------------------|
| Author(s) Seessalo Toni | Type of Publication Bachelor's thesis | Date May 2023 |
| | 26 | Language of publication: Finnish |
| Title of publication Possibilities of Visual Components simulation environment in simulation of welfare technology applications | | |
| Degree programme Degree program in electrical and automation technology | | |
| <p>In this thesis, the possibilities of the Visual Components simulation environment in the simulation of welfare technology applications were investigated. The investigation was carried out by simulating the same scenario in two different ways. First, Visual Components' own functions were used, and then the same application was made using self-created functions. The operations were performed in the Python3 programming language in the code editor of Visual Components. In this way, the possibilities of Visual Components can be clearly evaluated.</p> <p>This thesis was carried out for the project called Ensuring the operation of critical healthcare environments using modern technology - RoboRFID, which Satakunta University of Applied Sciences and the University of Tampere have jointly implemented with funding from the Academy of Finland. One part of the RoboRFID project was to create a scenario where, for example, a mobile robot would help a resident with memory impairment who got lost in the public spaces of a care home at night to find their way back to their own room.</p> <p>Visual Components is computer simulation software specialized in the design of factories and their parts in various industries. However, its possibilities for simulating welfare technology applications have not been studied much before.</p> <p>A description of the operation and goals were prepared for the scenarios to be simulated. According to them, the mobile robot performs its normal logistical tasks in the public areas of the care home building, where the resident with memory impairment is simultaneously walking lost. When the mobile robot and the resident meet, the mobile robot recognizes the resident and takes them back to their own room. After this, the mobile robot continues with its logistical tasks.</p> <p>It is not possible to simulate the scenario using only Visual Components' own functionalities. However, when using self-created functions, the goals of the scenario simulation are met, and the versatility of the functions depends only on the developer's programming skills. Visual Components is suitable for simulating welfare technology applications, requiring strong programming skills from the developer, in which case a large part of Visual Components' own functionality remains unused.</p> | | |
| Keywords mobile robotics, simulation, welfare technology | | |

ALKUSANAT

Haluan aloittaa kiittämällä kaikkia, jotka ovat auttaneet minua opinnäytetyöni toteutamisessa Satakunnan ammattikorkeakoulussa. Erityisesti haluan kiittää Mirka Leinoa, Tommi Lehtistä, Sari Merilampea sekä puolisoani Maria.

Tommi Lehtinen, opinnäytetyön teknisen osuuden tarkastaja ja alan rautainen ammattilainen, on ollut hyvällä tavalla vaativa. Hän on jaksanut muistuttaa minua siitä, että edistäisin opinnäytetyöni hetkinä, jolloin se on jäänyt työlistallani taka-alalle. Hänen asiantuntemuksensa ja vaatimustasonsa ovat auttaneet minua kehittämään taitojani ja varmistamaan, että opinnäytetyöni on riittävän laadukas ja ammattimainen. Kiitos Tommi, että olet ollut mukana tukemassa minua.

Mirka Leino, opinnäytetyön toinen tarkastaja ja mentori, on ollut minulle tärkeä tuki koko projektin ajan. Hän on ollut kannustamassa minua ylittämään itseni opinnäytetyöni kanssa. Hänen ammattitaitonsa, asiantuntemuksensa ja sitoutuneisuutensa ovat auttaneet minua ottamaan edistysaskeleita opinnäytetyöni parissa. Mirka on ollut kärsivällinen ja ymmärtäväinen, ja hänen kannustuksensa on auttanut minua jaksamaan vaikeina hetkinä. Kiitos Mirka, ilman sinua en olisi päässyt tähän pisteeseen.

Sari Merilampi, opinnäytetyön tilaaja, on ollut korvaamaton osa projektiani. Hänen sitoutuneisuutensa on auttanut minua ymmärtämään projektin tärkeyden ja merkityksen. Kiitos Sari, että olet mahdollistanut tämän projektin toteutuksen.

Lopuksi haluan kiittää puolisoani Maria, joka on tukenut minua koko opinnäytetyöni ajan. Hänen kannustuksensa ja tukensa ovat olleet korvaamattomia, ja hän on ollut aina läsnä tukemassa minua ja auttamassa, kun olen tarvinnut sitä. Kiitos Mari, ilman sinua en olisi pystynyt saavuttamaan tätä tavoitetta.

SISÄLLYS

| | |
|---|----|
| 1 JOHDANTO | 6 |
| 2 KÄSITTEET | 7 |
| 2.1 Simulointi | 7 |
| 2.1.1 Tietokonesimulointi | 7 |
| 2.1.2 Visual Components | 8 |
| 2.2 Hyvinvointiteknologia | 8 |
| 2.3 Tietokonesimuloinnin käyttö ja sovellukset hyvinvointiteknologiassa | 9 |
| 3 PROJEKTIN MÄÄRITTELY, TAUSTA JA TOIMINNALLINEN YMPÄRISTÖ | 10 |
| 3.1 Skenaario | 10 |
| 3.2 Projektin toimeksianto | 11 |
| 3.3 Projektin rajausta ja riskejä | 11 |
| 4 SIMULAATIOIDEN VALMISTUS | 12 |
| 4.1 Skenaarioiden toiminnan kuvaus ja tavoitteet | 13 |
| 4.2 Oleellisten komponenttien kuvaus ja tavoitteet | 14 |
| 4.2.1 Mobiilirobotti | 14 |
| 4.2.2 Asukas | 15 |
| 4.3 Visual Componentsin omat toiminnallisuudet | 15 |
| 4.4 Itse rakennettu toiminnallisuus | 17 |
| 5 YHTEENVETO | 23 |
| LÄHTEET | |

1 JOHDANTO

Teollisuuden tuotannon 3D-simulointimahdollisuudet ovat kehittyneet viimeisen vuosikymmenen aikana. Teollisuusyritykset hyödyntävät yhä enemmän simulointia kehittäessään tuotantoprosessejaan, päivittäessään laitekantaa ja kokeillessaan vaihtoehtoisia tuotantotapoja. Tuotannon simuloinnin tarpeeseen on kehitetty useita erilaisia simulointiohjelmistoja, ja yksi näistä on Visual Components.

Myös hyvinvointiteknologian kehitys on ottanut suuria harppauksia ja uusia sovelluksia kehitetään enenevässä määrin. Teollisuuden kehitystyössä käytetään osittain samoja tai samankaltaisia laitteistoja ja komponentteja kuin hyvinvointiteknologian kehitystyössä, mutta simulointitarpeet eroavat suurilta osin toisistaan. Hyvinvointiteknologian simuloinnin tarpeeseen ei tällä hetkellä ole yhtä kattavaa simulointiympäristövalikoimaa kuin teollisuuden simulointiin varsinkaan niissä tapauksissa, kun tarkoituksena on kuvata hyvinvointiteknologiasovelluksen prosesseja ja ominaisuuksia realistisesti.

Tämä opinnäytetyön tehtävänä on tutkia Visual Components -simulointiympäristön mahdollisuuksia hyvinvointiteknologian sovellusten simulointiin. Tämän opinnäytetyötä tavoitteena on toimia työkaluna tilanteissa, joissa hyvinvointiteknologiaa tutkittaessa tai kehitettäessä kartoitetaan siihen soveltuvaa simulointiohjelmistoa. Lisäksi tämän opinnäytetyön tavoitteena ja tehtävänä on osaltaan tukea ja jouduttaa hyvinvointiteknologian tutkimus- ja kehitystyötä.

2 KÄSITTEET

2.1 Simulointi

Tämän työn kannalta on oleellista ymmärtää, mitä simulointi tarkoittaa. Termit simulaatio ja simulaattori sekoittuvat usein virheellisesti julkaisuissa ja puheissa keskenään. Näin ollen on oleellista ymmärtää näiden termien eroavaisuudet. (Salmi, 2021, 6:20.) Todellisen tilanteen jäljittely on simulaatiota, kuten lasten leikkiminen ja puolustusvoimien sotaharjoitukset (Leino ym., 2019, s. 6). Simulaattori taas on laite tai muu kokonaisuus, jolla simuloinnin voi toteuttaa. On hyvä ottaa huomioon, että simulointia voi olla myös ilman simulaattoria. (Salmi, 2021, 7:35; 9:25.) Esimerkkinä tästä voidaan pitää vaikka ensiapukoulutusta. Ensiapukoulutuksissa harjoitellaan (simuloidaan) hätäelvytystä Anne-nuken (simulaattori) avulla. Hätäelvytystä on mahdollista simuloida myös ilman simulaattoria toisella ihmisellä. Se ei tosin tehokkaan paineluelvytysharjoituksen kannalta ole turvallista. (Salmi, 2021, 11:00.)

2.1.1 Tietokonesimulointi

Yksi simuloinnin muodoista on tietokonesimulointi, jossa simulointi toteutetaan tietokoneohjelmistolla. Teknologian kehityksen myötä nykyaikainen teollisen tuotannon simulointi voidaan toteuttaa 3D-virtuaalimaailmassa todellisuutta muistuttavassa ympäristössä. 3D-virtuaalimaailmaan on mahdollista luoda toiminnallisuudeltaan ja ulkonäöltään oikean kaltaisia laitteita tai jopa kokonaisia tuotantolaitoksia. Mitä yksityiskohtaisemmin simulointi toteutetaan, sitä tarkemmin 3D-virtuaalimaailmaan luotu ympäristö laitteineen muistuttaa ulkonäöltään ja toiminnallisuuksiltaan todellista. Simuloinnilla on mahdollista testata tuotannon tai tuotannon osan soveltuvuus ennen varsinaisten laitehankintojen tekemistä säästäen näin resursseja ja energiaa. (Leino ym., 2019, s. 6; National Institutes of Health, kohta Computer Modeling and Simulation.)

2.1.2 Visual Components

Visual Components on suomalainen, eri teollisuuden tehtaiden ja niiden osien suunnitteluun erikoistunut tietokonesimulointiohjelmisto. Ohjelmisto sisältää yksityiskohdaisesti mallinnettuja komponentteja ja toiminnallisuuksia muun muassa auto-, lääke-, ja raskaskoneteollisuuden tuotannon simulointitarpeisiin. (Visual Components, 2022.) Visual Componentsin komponenttikirjasto sisältää 2968 eri komponenttia (Visual Components -Ecatalog, 2023) ja sillä on mahdollista tehdä monimutkaisiakin toiminnallisuuksia ilman varsinaista ohjelmointiosaamista. Valmiiden komponenttien lisäksi Visual Components -ohjelmistoon voi viedä itse mallinnettuja komponentteja ja siellä on mahdollista kehittää itse suunniteltuja toiminnallisuuksia. Tässä opinnäytetyössä käytettiin Visual Componentsin Premium versiota 4.6.

2.2 Hyvinvointiteknologia

Hyvinvointiteknologia on laaja käsite ja käsittää useita erilaisia tekniikoita. Karkeasti voidaan todeta, että kaikki teknologia, millä pyritään parantamaan käyttäjänsä elämänlaatua, on hyvinvointiteknologiaa. Hyvinvointiteknologian käyttö työkaluna myös vähentää hoito- ja hoivapalveluiden painetta. Arjen apuvälineiden, kuten kuulolaitteiden lisäksi hyvinvointiteknologiaa hyödynnetään asiakkaan turvallisuuden parantamisessa, esimerkkinä turvahälyttimet tai matkapuhelimet. (Pohjoismainen hyvinvointikeskus, 2022.) Markkinoilta löytyy myös matkapuhelimia, joissa on turvahälyttimen ominaisuudet ja puhelimen käyttäjän kuulolaitetta varten induktiosilmukka (Doro AB, kohta Doro 731X; Doro AB, kohta Tutustu tuotteeseen Response by Doro).

Hyvinvointiteknologia edesauttaa asiakkaan aktiivisuutta, yhteiskuntaan osallistumista sekä elämisen omavaraisuutta. Myös omaisille annettavassa tuessa sovelletaan hyvinvointiteknologiaa muun muassa potilasnostimien muodossa. Hyvinvointiteknologia myös mahdollistaa monipuolisen tiedon keräämisen, jota voi hyödyntää erilaisiin tarkoituksiin. (Pohjoismainen hyvinvointikeskus, 2022.) Esimerkkinä tiedon keräämisestä on hyvinvointiteknologiaan lukeutuva lihasten puolieroja mittaavaa laite, jonka tuloksia pystytään hyödyntämään oikeanlaisen ja tehokkaan kuntoutussuunnitelman valinnassa. Huomioitavaa on, että hyvinvointiteknologian määrittely on kansallista (Dennis C Søndergaard, s. 10). Suomessa hyvinvointiteknologiaa ei ohjata

lainsäädännöllä, eikä sitä ole tarkoitettu pääasialliseen lääketieteelliseen käyttöön (Sailab, 2019, kohta Mitä on terveysteknologia?). Tässä opinnäytetyössä keskityttiin suomalaisen määrittelyn mukaiseen hyvinvointiteknologian 3D-simulointisovelluksen tutkimiseen.

2.3 Tietokonesimuloinnin käyttö ja sovellukset hyvinvointiteknologiassa

Tämän opinnäytetyön aiheen kannalta on aiheellista todeta, että varsinaisia kriittisiä terveydenhuoltoympäristöjä tai hoitotyön resurssipulaa helpottavia hyvinvointiteknologian sovellusten 3D-tietokonesimulaatioita ei tällä hetkellä ole. Myöskään teollisuuden tarpeisiin kehitetyn 3D-simulointiympäristön hyödynnettävyyttä hyvinvointiteknologian tarpeisiin ei ole tutkittu juurikaan. Hyvinvointiteknologian tarpeisiin on kuitenkin tuotettu muita tietokonesimulaatioita. Sosiaali- ja terveysalan opetuksessa erilaisia tietokonesimulaatioita hyödynnetään pedagogisesta näkökulmasta käsin (Merilampi, 2023). Muita hyvinvointiteknologian simulaatioita on esimerkiksi Human motion analysis and simulation tools -tutkimus (Francisco, 2015, s. 356), joka tutkii ihmisen liikkeiden simulointia, sekä kaupallinen Take The Wind S.A.:n kehittämä Body interact -sovellus (Body interact, 2023), joka puolestaan on hoitotyön henkilökunnalle ja opiskelijoille suunnattu potilassimulointiympäristö.

3 PROJEKTIN MÄÄRITTELY, TAUSTA JA TOIMINNALLINEN YMPÄRISTÖ

Tämä opinnäytetyö toteutettiin Satakunnan ammattikorkeakoulun ja Tampereen yliopiston toteuttamalle, Suomen Akatemian rahoittamalle, Kriittisten terveydenhuoltoympäristöjen toiminnan varmistaminen modernin teknologian avulla - RoboRFID-hankkeelle. Hankkeen tavoitteena on tutkia ja kehittää kriittisten terveydenhuoltoympäristöjen, kuten sairaaloiden ja hoivakotien, kriisien aikaisen toiminnan varmistamista modernin teknologian avulla. Tulokset ovat hyödynnettävissä myös normaaliajan hoitotyötä tekevän henkilökunnan resurssipulan helpottamiseksi. Käytettävä teknologia koostui mobiili- ja palvelurobotiikasta sekä puettavasta passiivisesta RFID-anturiteknologiasta. Käytettävään teknologiaan kuului myös edellä mainittujen teknologioiden tuottaman tiedon yhdistäminen ja saadun tiedon älykäs hyödyntäminen. RoboRFID-hanke koostui Satakunnan ammattikorkeakoulun RoboAI:n tutkimusryhmän lisäksi Tampereen yliopiston älyvaateteknologian tutkimusryhmästä.

3.1 Skenaario

RoboRFID-hankkeen yhtenä osana oli luoda skenaario, jossa mobiilirobotti auttaa esimerkiksi muistisairasta palvelukodin asukasta löytämään takaisin omaan huoneeseensa. Skenaariossa hoivakodin asukkaiden vaatteisiin oli kiinnitetty yksilöivät RFID-tunnisteet. Vastaavasti mobiilirobottiin oli asennettu RFID-antennit ja -lukija sekä konenäkökamerat niin, että ne kuvaavat 360 asteen kulmassa nähden samanaikaisesti joka puolelle ympärilleen. Itsenäisesti hoivakodin yleisissä tiloissa logistisia tehtäviä suorittava mobiilirobotti tunnistaa ja yksilöi muistisairaahan asukkaan RFID-teknologiaa käyttäen. Konenäköteknologiaa käyttäen selvitetään asukkaan asento, jolloin voidaan varmistaa, että asukas ei esimerkiksi ole kaatumisen seurauksena jäänyt mahaan makaamaan. Konenäköteknologiaa hyödynnetään myös täydentämään RFID-teknologian tuottamaa tunnistetietoa, koska RFID-teknologia saattaa tunnistaa asukkaan seinien läpi. Näin ollen pelkkä RFID-tunnistetieto ei riitä varmistamaan asukkaan sijaintia riittävällä tarkkuudella. Mobiilirobotti toimii ennalta ohjelmoitujen toimintamallien mukaan esimerkiksi ohjaten asukkaan takaisin huoneeseensa tai kutsumalla

hoitohenkilökunnan paikalle riippuen siitä, minkä tulkinnan robotti tekee asukkaan tarpeesta.

3.2 Projektin toimeksianto

Tässä opinnäytetyöprojektissa selvitettiin Visual Components -simulointiympäristön mahdollisuuksia tilanteessa, jossa logistisia tehtäviä suorittava, mutta samalla tunnistus- ja opastustehtäviin ohjelmoitu robotti löytää päämäärättömästi vaeltavan muistisaira-
raan hoivakodin asukkaan ja saattaa hänet takaisin omaan huoneeseensa. Mahdollisuudet selvitetään tutkimalla, miten Visual Components soveltuu skenaarion simulointiin.

3.3 Projektin raja- us ja riskit

Ensin skenaario toteutettiin vain Visual Components -ohjelmiston omia työkaluja ja toiminnallisuuksia käyttäen. Tämän jälkeen sama skenaario toteutettiin toistamiseen, mutta toiminnallisuus toteutettiin käyttäen itse luotuja toiminnallisuuksia, jotka tehtiin Visual Componentsin koodieditorissa Python3-ohjelmointikielellä kaikkia käytettävissä olevia kirjastoja hyödyntäen.

Omia toiminnallisuuksia käytettäessä keskityttiin vain toiminnallisuuden luomiseen, joten esimerkiksi ihmishahmon motoristen liikkeiden simulointi on rajattu tämän opinnäytetyön ulkopuolelle. Omia toiminnallisuuksia tehtäessä riskinä oli, että toiminnallisuuksista tulee opinnäytetyön laajuutta silmällä pitäen liian yksityiskohtaisia, mikä edellyttää kohtuuttoman paljon aikaa. Tämä on omiaan paisuttamaan opinnäytetyötä ja viivästyttämään opinnäytetyön valmistumista.

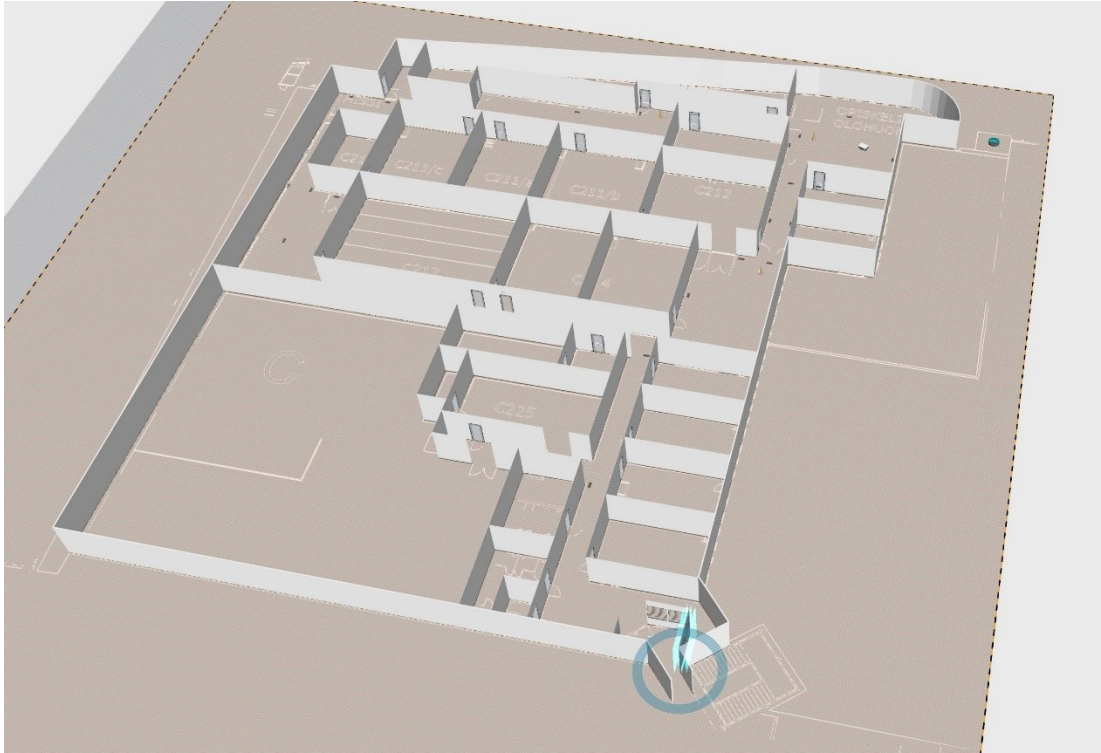
Visual Componentsin toiminnallisuuksiksi lasketaan myös ne sovelletut toiminnallisuudet, joita ei sellaisenaan ilman soveltamista ole tarkoitettu suoritettavaksi. Itse luoduiksi toiminnallisuuksiksi lasketaan myös ne kokonaisuudet, joissa hyödynnetään osittain Visual Componentsin olemassa olevaa toiminnallisuutta, mikäli se ei sellaisenaan täytä toiminnallisuuden tavoitteita, vaan edellyttää itse tehtyä yksilöllistä räätälöintiä. Tässä opinnäytetyössä tehdyt toiminnallisuudet ovat karkeita, eivätkä sisällä käytössä olevaa RFID- ja konenäkötekniikan simulointia.

4 SIMULAATIOIDEN VALMISTUS

Molempien tämän opinnäytetyön simulaatioiden maailma toteutettiin kuvitteellisessa palvelutalossa. Simulaatioissa oleva palvelutalo mallinnettiin Satakunnan ammattikorkeakoulun C-osan pohjakuvasta. Pohjakuva lisättiin Visual Components -ohjelmistoon (kuva 1). Pohjakuvan päälle rakennettiin seinät, ovet ja ikkunat Visual Componentsin omaa toiminnallisuutta eCatalogin Fence and Wall builderia käyttäen. (kuva 2).



Kuva 1. SAMKin c-päädyn pohjakuva



Kuva 2. Pohjakuvan päälle rakennetut seinät, ovet ja ikkunat

4.1 Skenaarioiden toiminnan kuvaus ja tavoitteet

Skenaariossa on määritelty mobiilirobotin toimintaa sekä palvelutalossa harhailevan muistisairaankin toimia. Skenaarion kuvaus ja tavoitteet näyttävät seuraavalta:

1. Mobiilirobotti suorittaa sille suunniteltuja logistisia tehtäviä palvelutalon yleisissä tiloissa kiertäen säännönmukaisesti kahdeksan eri reittipisteen kautta tarkkaillakseen samalla ympäristöä. Käytyään viimeisessä pisteessä robotti aloittaa kierroksen uudelleen.
2. Muistisairas asukas kulkee samoja kahdeksaa reittipistettä satunnaisesti. Saa-vuttuaan paikkapisteelle asukas menee satunnaisesti joko eteenpäin seuraavalle pisteelle tai taaksepäin edelliselle pisteelle.
3. Tilanteessa, jossa robotti ja asukas kohtaavat, asukas alkaa robotin kehotuksesta seurata robottia.
4. Mobiilirobotti lopettaa säännönmukaisen reittipisteiden kiertämisen ja kulkee lyhintä reittiä reittipisteeseen, jossa asukkaan huone sijaitsee. Asukas seuraa robottia edelleen.

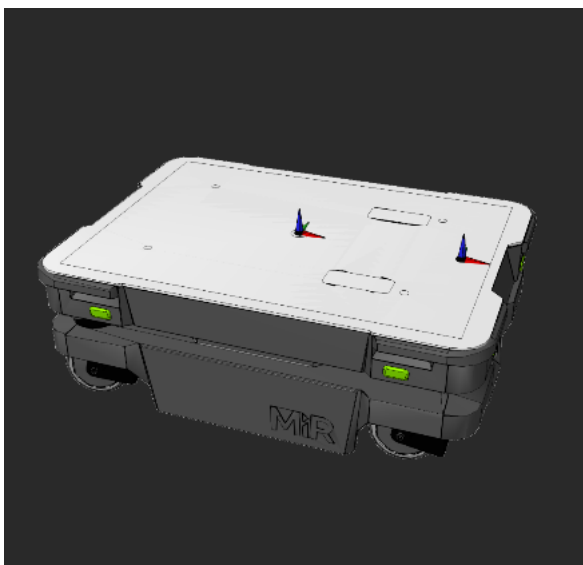
5. Asukas jää huoneeseensa ja mobiilirobotti palaa takaisin reittipisteelle yksi ja jatkaa tehtäviään kiertäen samalla säännönmukaisesti reittipisteiden kautta.

4.2 Oleellisten komponenttien kuvaus ja tavoitteet

Skenaarion tärkeimmät komponentit ovat mobiilirobotti ja muistisairas asukas. Seuraavassa on kuvattu näiden toimintaa.

4.2.1 Mobiilirobotti

Mobiilirobotin toiminnallisuudet soveltuvat mainiosti perinteisiin teollisuuden tarpeisiin – logistiset tehtävät, jossa robotti saapuu käskystä noutamaan tai viemään, toimivat loogisesti. Visual Components -ympäristössä on helppo määrittää robotin toiminta-alue ja robotin reitille on yksinkertaista luoda erilaisia alueita, jossa robotti kulkee esimerkiksi tiettyä nopeutta, tai jonne se ei mene ollenkaan, mikäli alueella on jo robotti. Robotin saa myös helposti ohjelmoitua varomaan muita tilassa liikkuvia ja antamaan tilaa. Kaikki edellä mainitut toiminnallisuudet ovat valmiina Visual Componentsissa, eikä niiden käyttö edellytä varsinaista ohjelmointiosaamista. Tässä simulaatiossa käytettiin MiR250-mobiilirobottia (kuva 3), koska sama robottimalli oli käytössä myös RoboRFID-projektissa.



Kuva 3. MiR250-mobiilirobotti

4.2.2 Asukas

Visual Componentsin komponenttivalikoimasta E-catalogissa löytyy ihmishahmoa esittävät 3D-mallit nimeltään Anna ja Otto. Tässä opinnäytetyössä muistisairasta asukasta kuvataan Anna-hahmolla (kuva 4). Ihmishahmot toimivat simuloidun tuotannon resursseina, joilla niin ikään on logistisen käyttötarkoituksen lisäksi kokoonpanoa simuloiva tarkoitus. Visual Componentsin toiminnallisuuksien ansiosta ihmishahmot liikkuvat luonnollisesti ilman määrittelyä.



Kuva 4. Anna-hahmo

4.3 Visual Componentsin omat toiminnallisuudet

Tässä kappaleessa käydään läpi Visual Componentsin omat toiminnallisuudet, joiden avulla skenaarion simulaatiota voitaisiin toteuttaa. Nämä toiminnallisuudet on jaettu viiteen kohtaan, jotka ovat seuraavat:

✓ Kohta 1, Mobiilirobotin perustoiminnot

Kohdassa 1 esitetty robotin logististen tehtävien suorittaminen ja ympäristön valvonta onnistuvat ongelmitta. Visual Componentsin maailmaan luodaan Mobile Robot Transport Controller, Idle Path -node ja Pathway Area. Mobile Robot Transport Controller yhdistää mobiilirobotin, Idle Path- noden ja Pathway Areen

toisiinsa ja mahdollistaa 1. kohdan tavoitteiden mukaisen määrittelyn. Tämä vaihe on hyvin tyypillinen myös teollisuuden tuotannon simuloinnissa.

✓ Kohta 2, Reitin määrittely satunnaiseksi

Satunnaisesti määriteltä reittiä pitkin eteneminen on myös Visual Componentsin omilla toiminnallisuuksilla mahdollista. Visual Componentsin omia toiminnallisuuksia hyödyntäen on mahdollista generoida satunnaisia tapahtumia. Generoiminen edellyttää ohjelmointikielistä tutun uniform-metodin hallitsemista, jossa metodi palauttaa satunnaisen luvun kahden määritetyn arvon välistä. Satunnaisesti eteneminen on myös mahdollista korvata määrittämällä asukkaan liikkuminen reitillä niin, että se vaikuttaa satunnaiselta, mutta todellisuudessa reitti on määritetty ennalta. Tässä opinnäytetyössä satunnaisuus on kuitenkin keskeisessä osassa simulaation soveltuvuuden määrittelyssä. Ilman sitä simulaatio on ennalta käsikirjoitettu tapahtuma, joka toistuu aina samalla tavalla.

✓ Kohta 3, Ihmishahmon perustoiminnot

Visual Componentsin ihmishahmolla ei ole sellaista toiminnallisuutta, joka mahdollistaa toisen komponentin (mobiilirobotin) seuraamisen. Visual Componentsin omia toiminnallisuuksia käyttämällä ei myöskään voida simuloida tilanteita, joissa määritetty komponenttien välinen etäisyys johtaa tietynlaiseen toimintaan. Tämä onnistuu vain, mikäli se on osa ennalta määrättyä prosessia, ja tässä skenaariossa tilanne ei ole ennalta määrätty.

✓ Kohta 4, Reitin uudelleenmäärittely

Visual Componentsin omilla toiminnallisuuksilla on mahdollista määrittää mobiilirobotille uusi reitti, kun tietty ehto täyttyy. Kohdan 4 tyypinen reitin vaihto määritetään hyödyntämällä toiminnallisuutta, jossa robotin reitti määräytyy sen mukaan, mitä asioita sillä on kyydissä. Tässä tilanteessa kyydissä on esimerkiksi näkymätön esine, jonka perusteella robotin reitti määräytyy uudelleen. Kohdan 4 tavoite voidaan näin ollen katsoa toteutuneeksi.

✓ Kohta 5, Skenaarion päättäminen

Työkierron uudelleenmäärittäminen on mahdollista samalla tavalla kuin kohdan 4 esimerkissä.

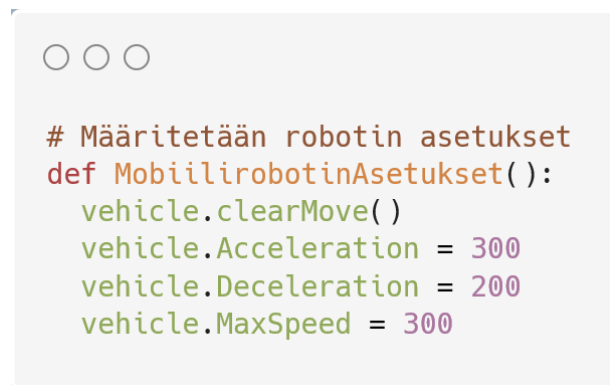
Visual Componentsin toiminnallisuuksia käyttäen komponentit ovat tarkasti ja näyttävästi animoitu, mutta simulaation tavoitteita ei saada täytettyä kuin osittain. Kiistatta voidaan todeta, että hyödyntämällä pelkästään Visual Componentsin omia toiminnallisuuksia se ei ole käyttökelpoinen ohjelmisto skenaarion toteuttamiseen.

4.4 Itse rakennettu toiminnallisuus

Kaikki tässä opinnäytetyössä tehty ohjelmointi on tehty Python3-kielellä Visual Componentsin omassa ohjelmointiympäristössä. Python3-koodista otettujen kuvien tarkoitus on auttaa hahmottamaan toiminnallisuutta, joten ohjelmointikoodia ei ole esitetty kokonaisuudessaan.

✓ Kohta 1, Mobiilirobotin perustoiminnot

Jotta mobiilirobotin liikkuminen simulaatiossa on mahdollista kohdan 1 mukaisesti, tulee robotille tehdä liikkumisen mahdollistavat määitykset (kuva 5).

A screenshot of a code editor window with a light gray background. At the top left, there are three small, empty circles. The code is written in a Python-like syntax with syntax highlighting: comments are in brown, function definitions in orange, and variable assignments in green. The code defines a function to set robot parameters for a vehicle object.

```
# Määritetään robotin asetukset
def MobiilirobotinAsetukset():
    vehicle.clearMove()
    vehicle.Acceleration = 300
    vehicle.Deceleration = 200
    vehicle.MaxSpeed = 300
```

Kuva 5. Mobiilirobotin asetukset liikkumista varten

Reittipisteet on nimetty ohjelmistoympäristöön checkpoint00 – checkpoint07. Toistolauseessa ohjelma päivittää 0,3 sekunnin välein omaa, asukkaan ja seuraavan reittipisteen sijaintia. Ohjelmassa annetaan myös robotille käsky mennä seuraavalle reittipisteelle (kuva 6).

```

def Ajoreitti():
    try:
        global c

        while True:
            delay(0.3)
            #Robotti kiertää jokaista kahdeksaa reittipistettä järjestyksessä tässä loopissa.

            # Checkpointit ovat:
            # checkpoint00
            # checkpoint01
            # checkpoint02
            # checkpoint03
            # checkpoint04
            # checkpoint05
            # checkpoint06
            # checkpoint07

            # Määritetään, että seuraava checkpoint on c:n arvo
            checkpoint = app.findComponent("Checkpoint0" + str(c))

            # Päivitetään checkpointin sijaintia
            wpm = (checkpoint).WorldPositionMatrix
            checkpoint = wpm.P

            # Päivitetään omaa sijaintia
            roboWpm = (robo).WorldPositionMatrix

            # Päivitetään asukaas sijaintia
            asukasWpm = (asukas).WorldPositionMatrix

            # Päivitetään robon etäisyyttä asukkaasta
            robonEtäisyysAsukaasta = math.hypot(asukasWpm.P.X - roboWpm.P.X, asukasWpm.P.Y - roboWpm.P.Y)

            # Päivitetään robon etäisyys seuraavasta checkpointista
            robonEtäisyysCheckpointista = math.hypot(roboWpm.P.X - checkpoint.X, roboWpm.P.Y - checkpoint.Y)

            # Siirrytään yllä määritettyyn checkpointtiin
            vehicle.addControlPoint(checkpoint)

```

Kuva 6. Mobiilirobotin kulkeminen reitillä

Robotille on määritelty ehto, että sen ollessa 100 yksikön sisällä kohteena olevasta reittipisteestä sen seuraava reittipiste määräytyy uudelleen. Reittipisteen määrittäminen on toteutettu niin, että checkpoint0x-muuttujan x-kirjaimen tilalla on sama numero kuin toistolauseen c-muuttuja. C-muuttuja kasvattaa arvoaan yhdellä aina, kun robotti on 100 yksikön sisällä kohteena olevasta checkpointista. Näin robotti etenee reittipisteet järjestyksessä. Poikkeuksena tästä on c-muuttujan arvo 8, jolloin arvo muutetaan nolaksi ja robotti aloittaa kierroksen alusta (kuva 7).

```

○ ○ ○

# Robotin lähestyessä kohteena olevaa checkpointtia, määritetään robotille uusi checkpoint
elif robotEtäisyysCheckpointista < 100:

    # Nostetaan c:n arvoa joka kierroksella.
    c += 1
    delay(0.3)
    # c:n arvon ollessa enemmän kuin checkpointtien määrä, aloitetaan kierros alusta
    if c == 8:
        c = 0

```

Kuva 7. Mobiilirobotin kulkeminen reittipisteeltä

✓ Kohta 2, Reitin määrittely satunnaisiksi

Asukas on teknisesti määritetty mobiilirobotiksi, jolloin sen liikkuminen simuloitinympäristössä noudattaa samoja periaatteita kuin mobiilirobotti. Mobiilirobotin tavoin asukkaan liikkumiseen liittyvät asetukset tulee ensin määrittää (kuva 8).

```

○ ○ ○

def asukkaanAsetukset( ):
    vehicle.clearMove( )
    vehicle.Acceleration = 200
    vehicle.Deceleration = 200
    vehicle.MaxSpeed = 250

```

Kuva 8. Asukkaan asetukset simulaatiossa

Muistisairaan asukkaan satunnainen liikkuminen on toteutettu Python Random seed() -menetelmällä (Python random — Generate pseudo-random numbers, 2021) (kuva8), joka hyödyntää nykyistä järjestelmän aikaa. Simulointiohjelman käynnistyessä taustalle generoituu lista 200 alkioista, jotka saavat satunnaisesti arvon 1 tai -1 (kuva 9).

```

○ ○ ○

def GeneroidaanReitti():
    global reittilista, prop
    for i in range(200):
        num = prop.Value
        num = int(num)
        if (num % 2) == 0:
            reittilista.append(-1)
        elif (num % 2) != 0:
            reittilista.append(1)

```

Kuva 9. Aukkaan reitin generointi.

Aukkaan koodi lukee listalta indeksinumeroltaan seuraavan alkion arvon aina uudelle reittipisteelle saapuessaan, ja tämä alkion arvo lisätään kohteena olevan checkpointin arvoon. Esimerkiksi jos nykyinen checkpoint on 07, ja listasta generoituu arvo -1, $07 + (-1) = 6$, jolloin asukas menee taaksepäin. Poikkeuksena on ensimmäinen ja viimeinen reittipiste: aukkaan meneminen pisteille -1 tai 8 on estetty, koska tällaisia pisteitä ei ole olemassa.

✓ Kohta 3, Ihmishahmon perustoiminnot

Tilanne, jossa asukas seuraa mobiilirobotia, on simulaatiossa toteutettu niin, että mobiiliroboti päivittää koko ajan omaa etäisyyttään aukkaasta. Aukkaan ollessa riittävän lähellä mobiilirobotin koodi kytkee aukkaan robotin omaksi perävaunukseen (kuva 10), jolloin se ei kulje enää kohdan 2 mukaisten reittipisteiden välillä.

○ ○ ○

```
def Etsiasukas():  
    # Ilmoitetaan asukaan scriptille, että asukas on löytynyt  
    asukasSignal.signal(True)  
  
    # Katkaistaan robotin aikaisempi reittisuunnitelma  
    vehicle.rePlan()  
  
    # Liitetään asukas perävaunuksi.  
    vehicle.detachAllWagons()  
  
    # Määritetään perävaunun etäisyydeksi 200  
    vehicle.attachWagonAtTheEnd(asukas, 200)
```

Kuva 10. Asukkaan liittäminen perävaunuksi.

✓ Kohta 4, Reitin uudelleenmäärittely

Tämä kohta on toteutettu niin, että mobiilirobotin koodissa ohjelma tarkkailee jatkuvasti robotin etäisyyttä asukkaasta (kuva 11). Robotti menee joko eteenpäin edeten tai taaksepäin kääntyen asukkaan huoneeseen riippuen siitä, kumpi reitti on lyhyempi. Takaisinpäin mentäessä c-muuttujan arvoa vähennetään joka reittipisteen kohdalla, kunnes saavutaan ensimmäiselle reittipisteelle, jossa asukaan huone sijaitsee. Eteenpäin mentäessä robotti vapauttaa asukkaan saavuttuaan asukkaan huoneessa sijaitsevalle reittipisteelle.

○ ○ ○

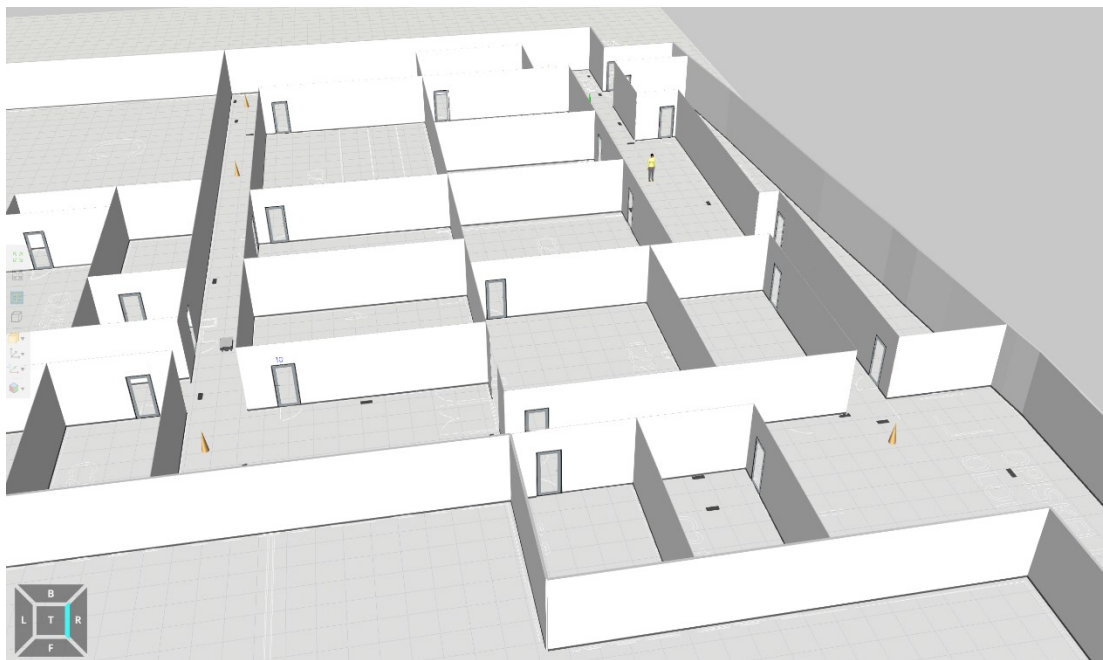
```
#Kun asukkaan ja robotin etäisyys toisistaan on < 2000, mennään lyhintä reittiä checkpoint00 -reittipisteelle.  
  
if robotEtäisyysAsukaasta < 2000:  
    Etsiasukas()  
    if c >= 5:  
        Takakautta(c)  
        break  
    if c < 5:  
        Etukautta(c)  
        break
```

Kuva 11. Lyhimmän reitin valinta.

✓ Kohta 5, Skenaarion päättäminen

Uuden työkierron aloittaminen onnistuu ehtolauseella, joka tekee skenaarion toteuttamisesta päättymättömän.

Omia toiminnallisuuksia käyttäen komponenttien liikkeet simulointiympäristössä ovat kömpelöitä, esimerkiksi asukkaan askeleita ei tässä ole ollenkaan. Tämä johtuu siitä, että teknisesti asukas määritellään mobiilirobotiksi, eikä mobiilirobotilla ole ihmisas-
kelten kaltaisia ominaisuuksia. Tämä osaltaan vaikuttaa simulaation laatuun heikentä-
västi. Asukkaan askeleita on mahdollista simuloida, mutta se edellyttää suuren määrän
ohjelmointia. Asukkaan askeleilla ei ole tämän opinnäytetyön kannalta niin suurta
merkitystä, että olisi tarkoituksenmukaista sisällyttää sitä tähän opinnäytetyöhön. Toi-
minnallisuudet puolestaan toteuttavat kaikki asetetut tavoitteet (Kuva 12).



Kuva 12. Valmis simulaatio, jossa näkyvät robotti (vasemmalla pystysuoralla käytävällä), asukas (oikealla pystysuoralla käytävällä) sekä checkpointit (ruskeat kartiot)

Tässä opinnäytetyössä rakennetut toiminnallisuudet on tehty opinnäytetyön laajuutta silmällä pitäen. Tästä syystä on huomioitava, että jokaista tehtyä toiminnallisuutta on mahdollista jatkokehittää ja parantaa.

5 YHTEENVETO

Tämän opinnäytetyön tavoitteena oli arvioida Visual Componentsin mahdollisuuksia hyvinvointiteknologian sovellusten simulointiin. Arviointia tehtiin vertailemalla Visual Componentsin omia toiminnallisuuksia itse luotuihin toiminnallisuuksiin siten, että luotiin molempia toiminnallisuuksia varten sama skenaario. Skenaariossa mobiilirobotti suorittaa sille suunniteltuja logistisia tehtäviä kiertäen säännönmukaisesti palvelutalon yleisissä tiloissa ja tarkkaillee samalla ympäristöä. Muistisairas asukas liikkuu samassa tilassa ja robotin ja asukaan kohdatessa asukas alkaa robotin kehotuksesta seurata robottia. Mobiilirobotti lopettaa säännönmukaisen reittipisteiden kiertämisen ja kulkee lyhintä reittiä asukkaan huoneeseen asukaan seurattessa robottia. Asukas jää huoneeseensa ja mobiilirobotti palaa takaisin suorittamaan kesken jääneitä logistisia tehtäviä.

Visual Components -simulointiympäristöön on mahdollista tuoda itse mallinnettuja komponentteja, ja niiden toiminnallisuudet ovat lähes rajattomasti muokattavissa. Vaikka Visual Components on erikoistunut teollisuuden simuloinnin tarpeisiin, eivätkä ohjelmiston omat toiminnallisuudet täytä simuloinnin tarpeita, sen tarjoamat mahdollisuudet ovat hyvinvointiteknologian simuloinnin tarpeisiin erinomaiset. Ohjelmiston avoimuus omille muokkauksille ja laaja tuki Python-ohjelmoinnille mahdollistavat sen, että simuloinnista saa erittäin yksityiskohtaisen ja tarkasti räätälöidyn. Visual Components -simulointiympäristön hallitseminen yhdessä ohjelmointiosaamisen kanssa tekee siitä hyvin toimivan työkalun hyvinvointiteknologian simuloinnin tarpeisiin.

Tässä opinnäytetyössä tehdyt toiminnallisuudet ovat vain esimerkki siitä, miten minäkin kohdan toiminnallisuus ja tavoite voidaan saada täytettyä. Samat toiminnallisuuksien tavoitteet on mahdollista saada täytettyä monilla eri tavoin. Esimerkkinä mainittakoon, että tilanteessa, jossa asukas seuraa mobiilirobottia, on mahdollista, että perävaunuksi kytkemisen sijaan mobiilirobotti ”tiputtaa” jälkeensä uusia reittipisteitä, joita asukas seuraa.

Tätä opinnäytetyötä on mahdollista jatkokehittää siten, että oikeassa elämässä aistinvaraisesti tarkasteltuna näkymättömissä olevat RFID-signaalit ja konenäkökameran näkökenttä tuodaan simulaatiossa näkyviksi keiloiksi. Keilojen tarkka ja realistinen määrittämien ja näkyviin tuominen antaa simulaatiolle lisäarvoa, koska se mahdollistaa mobiilirobotin testaamisen uusissa ympäristöissä ilman että sitä viedään fyysisesti paikalle.

Tätä opinnäytetyötä tehdessä haastavinta oli selvitystyö, jossa arvioitiin Visual Componentsin toiminnallisuuksia skenaarion tavoitteiden täyttämiseksi. Tämä johtui siitä, että kaikki internetistä saatavilla olevat esimerkit ja oppaat oli tehty teollisuuden simuloinnin tarpeisiin. Toki on todettava, että juurikin esimerkkien ja oppaiden puuttuminen teki tästä opinnäytetyön tekemisestä mielenkiintoisen.

Omia toiminnallisuuksia ohjelmoitaessa oli yllättävää huomata, että yksinkertaisilla koodirakenteilla pystyi saamaan aikaiseksi monipuolisia toiminnallisuuksia. Ohjelmoinnissa työläintä oli kokonaisuuden hallitseminen ja hahmottaminen, mikä tätä opinnäytetyötä tehdessä onnistui kuitenkin hyvin. Jos aloittaisin opinnäytetyön uudelleen, panostaisin nykyistä enemmän simulaation visuaalisuuteen, koska visuaalisesti heikko simulaatio antaa lopputuloksesta puutteellisen vaikutelman, vaikka toiminnallisuudet olisivatkin viimeisteltyjä.

LÄHTEET

Body interact. Haettu 10.5.2023 osoitteesta <https://bodyinteract.com/>

Doro AB. Doro 731X. Haettu Haettu 9.6.2022 osoitteesta <https://www.doro.com/fi-fi/shop/mobile-devices/easy-phones/doro-731x-2c5b5001>

Doro AB. Tutustu tuotteeseen Response by Doro. Haettu Haettu 9.6.2022 osoitteesta <https://www.doro.com/fi-fi/miksi-doro/response-by-doro>

Francisco. (2015). Handbook of Research on Computational Simulation and Modeling in Engineering. IGI Global. https://www.researchgate.net/publication/281845818_Human_Motion_Analysis_and_Simulation_Tools_A_Survey

Leino, M., Lehtinen, T., Kortelainen, J., Suvela, T., Asmala, H., Jokinen, S., Valo, P. (2019). Simulointiympäristöllä uusia ulottuvuuksia tuotannon automatisointiin, Loppuraportti. Satakunnan Ammattikorkeakoulu. https://www.theseus.fi/bitstream/handle/10024/170858/2019_B_9_SAMK_Simulointiymparistolla_uusia_ulottuvuuksia_tuotannon_automatisointiin.pdf

Merilampi, S. (16.5.2023). Henkilökohtainen keskustelu Satakunnan ammattikorkeakoulun johtavan tutkijan, Sari Merilammen, kanssa

National Institutes of Health. Computer Modeling and Simulation. Haettu 6.6.2022 osoitteesta <https://ors.od.nih.gov/OD/OQM/cms/Pages/default.aspx>

Pohjoismainen hyvinvointikeskus. Haettu 8.6.2022 osoitteesta <https://nordicwelfare.org/fi/hyvinvointipolitiikka/hyvinvointiteknologia/>

random — Generate pseudo-random numbers. Haettu 11.5.2023 osoitteesta <https://docs.python.org/3/library/random.html>

Sailab-MedTech Finland ry. (2019). Mitä on terveysteknologia?
https://www.sailab.fi/wp-content/uploads/2019/09/mitaterveysteknologiaon_opas.pdf

Salmi, A. (Väitöskirjatutkija). (4.5.2021). Osa1. Simulaation lyhyt oppimäärät [Podcast-jakso]. Sarjassa Simutarinat by Simulation Wizard. Anchor by Spotify. <https://anchor.fm/simulationwizard>

Søndergaard, D. Hyvinvointiteknologia työkalupakki, Pohjoismainen hyvinvointikeskus. https://nordicwelfare.org/wp-content/uploads/2017/10/finland_webb.pdf

Visual Components. Haettu 8.6.2022 osoitteesta <https://www.visualcomponents.com>

Visual Components -Ecatalog. Haettu 10.5.2022 osoitteesta <https://www.visualcomponents.com/ecatalog/#/>