



**TURUN
YLIOPISTO**

Mallipohjaisen suunnittelun ja mallipohjaisen järjestelmäsuunnittelun erot ja soveltuvuus teollisuudessa

Konetekniikka/ Kone- ja materiaalitekniikan laitos

TkK-tutkielma

Saga Salmi

13.05.2026

Turku

Turun yliopiston laatu järjestelmän mukaisesti tämän julkaisun alkuperäisyys on tarkastettu

Turnitin OriginalityCheck -järjestelmällä.

Kandidaatintutkielma

Koulutusohjelma, oppiaine: Tekniikan kandidaatti, konetekniikka

Tekijä: Saga Salmi

Otsikko: Mallipohjaisen suunnittelun ja mallipohjaisen järjestelmäsuunnittelun erot ja soveltuvuus teollisuudessa

Ohjaaja: Jani Heikkinen

Sivumäärä: 37 sivua

Päivämäärä: 13.05.2026

Modernien teknisten järjestelmien monimutkaisuuden kasvu on lisännyt tarvetta tehokkaammille suunnittelumenetelmille korkean teknologian aloilla. Perinteiset dokumentaatiopohjaiset suunnittelumenetelmät eivät enää riitä hallitsemaan järjestelmien kasvavaa ohjelmisto- ja järjestelmäkompleksisuutta, minkä vuoksi mallipohjaiset lähestymistavat ovat yleistyneet. Tässä tutkielmassa tarkastellaan mallipohjaista suunnittelua ja mallipohjaista järjestelmäsuunnittelua, niiden keskeisiä eroja, yhtäläisyyksiä ja käyttökohteita eri toimialoilla.

Työn tavoitteena on selvittää, miten mallipohjainen suunnittelu ja mallipohjainen järjestelmäsuunnittelu eroavat toisistaan sekä miten niitä hyödynnetään teollisuudessa. Työssä tarkastellaan menetelmien soveltamista muun muassa auto-, ilmailu-, lääke-, rakennus- ja infrastruktuurialalla sekä kemianteollisuudessa. Työ toteutettiin kirjallisuuskatsauksen hyödyntämällä aiheeseen liittyvää tieteellistä kirjallisuutta, standardeja ja järjestelmäsuunnittelun lähde teoksia.

Työssä havaittiin, että mallipohjainen järjestelmä suunnittelu keskittyy järjestelmän vaatimusten, rakenteen ja arkkitehtuurin määrittelyyn, kun taas mallipohjainen suunnittelu painottuu järjestelmän simulointiin, toteutukseen ja testaukseen. Menetelmät eivät kuitenkaan ole toisistaan erillisiä, vaan ne täydentävät toisiaan osana yhtenäistä kehitysprosessia. Mallipohjaiset menetelmät mahdollistavat simulointien, jäljitettävyyden ja järjestelmätason analyysin hyödyntämisen jo kehityksen alkuvaiheessa, mikä auttaa vähentämään virheitä ja tukee monimutkaisten järjestelmien hallintaa.

Tulosten perusteella mallipohjaiset menetelmät soveltuvat erityisesti monimutkaisten ja ohjelmistointensiivisten järjestelmien kehitykseen. Menetelmien merkitys kasvaa jatkuvasti digitalisaation, teollisuus- 4.0 kehityksen ja verkottuneiden järjestelmien yleistymisen myötä.

Avainsanat: mallipohjainen suunnittelu, mallipohjainen järjestelmäsuunnittelu, SysML, järjestelmäsuunnittelu, teollisuus

Sisällysluettelo

1	Johdanto	4
2	Mallipohjaisen suunnittelun käsitteellinen tausta	6
2.1	Mallipohjainen suunnittelu	6
2.2	Mallipohjainen järjestelmäsuunnittelu	7
2.3	Suunnittelumenetelmien erot	9
3	Mallipohjainen suunnittelu eri aloilla	14
3.1	Autoteollisuus	14
3.2	Ilmailu ja avaruustekniikka	18
3.3	Teollisuusautomaatio ja koneenrakennus	22
3.4	Lääketekniikka	25
3.5	Muut alat	29
4	Johtopäätökset	32
5	Yhteenveto	35
	Lähteet	36

1 Johdanto

Teknisten järjestelmien kompleksisuus on kasvanut merkittävästi digitalisaation ja ohjelmistointensiivisyyden lisääntymisen myötä [1], [2]. Modernit järjestelmät eivät enää koostu yksittäisistä komponenteista, vaan ne muodostuvat usein toisistaan riippuvaisista alijärjestelmistä [2]. Erityisesti autoteollisuudessa järjestelmien monimutkaisuus on kasvanut ohjelmistojen, antureiden ja verkottuneiden järjestelmien lisääntyessä [3]. Teollisuus-4.0 kehitys on lisännyt järjestelmien integraatiota ja reaaliaikaisen datan hyödyntämistä useilla teollisuudenaloilla [4]. Järjestelmien välisten riippuvuuksien kasvu vaikeuttaa monimutkaisten järjestelmien suunnittelua ja hallintaa [1]. Nykyisten teknisten järjestelmien voidaan nähdä järjestelmien järjestelmänä, joissa useat eri teknologiat toimivat yhdessä yhtenäisen kokonaisuuden muodostamisen [2].

Perinteinen järjestelmäsuunnittelu perustuu laajaan dokumentaatioon, jonka hallinta vaikeutuu järjestelmien kompleksisuuden kasvaessa [1], [2]. Dokumenttipohjaisessa suunnittelussa tieto on usein hajautunut useisiin eri dokumentteihin, mikä vaikeuttaa tiedon ylläpitoa ja hallintaa [2]. Monimutkaisten järjestelmien muutosten hallinta on haastavaa perinteisissä dokumenttipohjaisissa suunnittelumenetelmissä, sillä muutokset eivät päivity automaattisesti kaikkiin järjestelmän osiin [5], [6]. Järjestelmien ohjelmistointensiivisyyden kasvu on lisännyt tarvetta tehokkaammille suunnittelumenetelmille kuin perinteinen dokumenttipohjainen lähestymistapa [3]. Dokumenttipohjaisessa suunnittelussa eri suunnittelualojen välinen tiedonkulku voi olla hidasta ja altista virheille [2]. Kasvava järjestelmäkompleksisuus on lisännyt tarvetta systemaattisille ja mallipohjaisille suunnittelumenetelmille [1], [2].

Mallipohjaiset suunnittelumenetelmät on kehitetty vastaamaan monimutkaisten järjestelmien suunnitteluhaasteisiin [1], [2]. Mallipohjaisessa lähestymistavassa mallit toimivat järjestelmään liittyvän tiedon keskeisenä lähteenä perinteisen dokumentaation sijaan [1]. Mallipohjainen suunnittelu hyödyntää simulointia ja analysointia järjestelmän toiminnan arvioimiseksi jo kehityksen alkuvaiheessa [7], [8]. Simulointipohjainen kehitys mahdollistaa virheiden tunnistamisen ennen fyysisten prototyyppien valmistamista, mikä voi vähentää kehityskustannuksia [8]. Mallipohjainen järjestelmäsuunnittelu keskittyy järjestelmätason suunnitteluun, kun taas mallipohjainen suunnittelu painottuu simulointiin, toetukseen ja

testaukseen[1], [7]. Mallipohjaiset menetelmät parantavat järjestelmien jäljitettävyyttä ja helpottavat muutosten hallintaa kehitysprosessien aikana [2], [6] Mallipohjainen lähestymistapa mahdollistaa eri suunnittelualojen tehokkaamman yhteistyön yhteisen järjestelmämallien avulla [6].

Tutkimuksen tavoitteena on tarkastella mallipohjaisen suunnittelun ja mallipohjaisen järjestelmäsuunnittelun keskeisiä eroja ja yhtäläisyyksiä [1], [7]. Työssä analysoidaan, miten mallipohjainen suunnittelu ja mallipohjainen järjestelmäsuunnittelu eroavat toisistaan järjestelmätason suunnittelun, simuloinnin ja toteutuksen näkökulmasta[1]. Työn tavoitteen on tarkastella mallipohjaisen suunnittelun ja mallipohjaisen järjestelmäsuunnittelun keskeisiä eroja sekä niiden soveltamista eri teollisuudenaloilla. Työssä pyritään vastaamaan seuraaviin tutkimuskysymyksiin:

TK1: Miten mallipohjainen suunnittelu ja mallipohjainen järjestelmäsuunnittelu eroavat toisistaan?

TK2: Miten mallipohjaisia menetelmiä sovelletaan eri teollisuudenaloilla monimutkaisten järjestelmien kehityksessä?

Työssä tarkastellaan mallipohjaisten menetelmien hyödyntämistä autoteollisuudessa, lääketekniikassa, rakennus- ja infrastruktuurialalla sekä kemianteollisuudessa[9], [10], [11]. Tutkimuskysymysten avulla pyritään selvittämään, millaisia hyötyjä mallipohjaiset menetelmät tarjoavat monimutkaisten järjestelmien suunnittelussa ja hallinnassa [2]. Työ toteutetaan kirjallisuuskatsauksen hyödyntämällä aiheeseen liittyvää tieteellistä kirjallisuutta ja järjestelmäsuunnittelun lähdeaineita [1], [2].

2 Mallipohjaisen suunnittelun käsitteellinen tausta

2.1 Mallipohjainen suunnittelu

Mallipohjainen suunnittelu on suunnittelumenetelmä, jossa otetaan huomioon tuotteen valmistuksen kaikki vaiheet. Tällä menetelmällä tuotekehitys aloitetaan luomalla järjestelmämalli, jonka avulla simuloidaan tuotteen toimintaa. Mallipohjaisen suunnittelun avulla pystytään luomaan valmistettavan tuotteen täydellinen ympäristö, jossa simuloidaan toimivuuden muutoksia eri olosuhteissa. Mallipohjainen suunnittelu on korvaava menetelmä periteisille menetelmille, kuten piirustuksille ja kaavioille. Mallipohjainen suunnittelu on varsin edullinen menetelmä, sillä tuotteen toimivuus voidaan testata ennen kalliiden investointien tekoa. [4]

Simulaatioon luodaan tuote ja sen ohjausalgoritmi, jonka jälkeen simulaatio voidaan ajaa. Näin valmistaja pääsee seuraamaan näiden kahden osion yhteen sopivuutta. Simulaation toimivuudesta voidaan huomata virheet aikaisemmin, jolloin säästytään mahdolliselta taloudelliselta haitalta. Simulaatio varmistaa turvallisen testauksen, sekä mahdollistaa hypoteettisten skenaarioiden testauksen ilman fyysistä prototyyppiä. Mallipohjaisen suunnittelun simulaatioprosessi mahdollistaa mallin toiminnan arvioinnin jo kehitysvaiheessa, jolloin voidaan tarkistaa täyttääkö järjestelmä sille asetetut vaatimukset. [2], [8]

Mallipohjaisella suunnittelun yksi tärkeimmistä osista automaattinen koodigenerointi. Automaattinen koodigenerointi linkittyy suoraan malliin. Kun mallinnus on tehty se kuvaa algoritmin ja toiminnan, jonka pohjalta ulkoinen työkalu, kuten Simulink tai Matlab muuntaa mallin automaattisesti koodiksi. Koska ihmisen ei tarvitse kirjoittaa kaikkea koodia, se minimoi inhimilliset virheet ja pienentää virhemarginaalia. Koodigeneroinnin avulla pystytään nopeuttamaan näin ollen kehitystä ja parantamaan jäljitettävyyttä. [12]

Mallipohjaisen suunnittelun toimivuus testataan vaiheittain koko valmistusprosessin ajan. Testauksessa hyödynnetään kolmivaiheista testausta, jonka vaiheet ovat mallisimulointi, ohjelmistosimulointi, ja laitteistosimulointi. Mallisimuloinnilla testataan mallinnettua tuotetta yhdessä sen ohjausalgoritmin kanssa tilanteessa, jossa molemmat osat ovat vielä simuloituja malleja. Tämän avulla voidaan havaita jo kehityksen alkuvaiheessa järjestelmien

yhteensopivuuteen ja toiminnallisuuteen liittyviä virheitä. Kun mallisimulointivaiheessa havaitut virheet on korjattu, siirrytään ohjelmistosimulointiin. Tässä vaiheessa automaattisesti generoitua koodia testataan simuloidun mallin kanssa. Testauksen avulla arvioidaan, toimiiko simulaatiosta generoitua koodi ulkoisella tietokoneella samalla tavalla kuin simulaatiossa oleva algoritmi. Ohjelmistosimuloinnin jälkeen voidaan aloittaa viimeinen vaihe eli laitteistosimulointi. Tässä vaiheessa simulaation pohjalta luotua fyysistä tuotemallia ja generoituja ohjelmistoja testataan yhdessä simuloidussa ympäristössä. Näin saadaan tietoa lopullisen tuotteen käyttäytymisestä mahdollisimman realistisissa käyttöolosuhteissa. [8]

2.2 Mallipohjainen järjestelmäsuunnittelu

Mallipohjainen järjestelmäsuunnittelu on suunnittelumenetelmä, jossa malli korvaa perinteisemmän dokumentaatiopohjaisen lähestymistavan. Mallipohjainen järjestelmäsuunnittelun mallinuksessa malli toimii keskeisenä tiedonlähteenä ja sisältää järjestelmään liittyvät vaatimukset, rakenteen sekä toiminnallisuudet. Menetelmää hyödynnetään erityisesti suurten ja monimutkaisen järjestelmien mallintamisessa, sillä sen avulla voidaan hallita useita riippuvuuksia, alijärjestelmiä ja teknologioita yhtenäisen mallin avulla. Mallipohjainen järjestelmäsuunnittelu ei ole pelkästään yksittäinen työkalu, vaan kokonaisvaltainen lähestymistapa monimutkaisten järjestelmien suunnitteluun. [1]

Mallipohjainen järjestelmäsuunnittelu perustuu hierarkkiseen rakenteeseen, jossa järjestelmä jaetaan eri tasoihin. Tämä rakenne tekee mallista hallittavan ja mahdollistaa sekä kokonaisuuden että yksityiskohtien tarkastelun. Mallipohjaisen järjestelmäsuunnittelun malli koostuu järjestelmä- ja komponenttitasosta. Järjestelmätasolla tuotetta tarkastellaan kokonaisuutena, ja keskitytään sen päätoimintaan sekä toiminnallisuuksiin, kun taas komponenttitasolla tarkastellaan järjestelmän yksittäisiin osiin. Mallin eri tasot ovat yhteydessä toisiinsa, minkä vuoksi järjestelmätason päätökset vaikuttavat myös komponenttitason ratkaisuihin. [6, 7]

Mallipohjaisessa järjestelmäsuunnittelussa keskeisessä roolissa ovat järjestelmän vaatimukset, jotka määrittävät mitä järjestelmän tulee tehdä ja millä ehdoilla sen tulee toimia. Vaatimukset voidaan jakaa funktionaalisiin vaatimuksiin, ei-funktionaalisiin vaatimuksiin, sekä

rajoitteisiin. Funktionaaliset vaatimukset kuvaavat järjestelmän toimintoja, eli mitä järjestelmä tekee, kun taas ei-funktionaaliset vaatimukset määrittävät, kuinka hyvin järjestelmä suorittaa tehtävänsä ja millaisia suorituskykyyn liittyviä ominaisuuksia siltä vaaditaan. Rajoitteet puolestaan kuvaavat järjestelmän teknisiä tai ympäristöön liittyviä rajoja. Vaatimukset ovat tärkeä osa järjestelmäsuunnittelua, sillä ne ohjaavat järjestelmän arkkitehtuuria ja määrittävät testauksen tavoitteet. Ilman selkeitä vaatimuksia ei voida arvioida, milloin järjestelmä täyttää sille asetetut tavoitteet. Järjestelmän arkkitehtuurin tehtävänä on varmistaa, että vaatimukset toteutuvat suunnitteluratkaisuissa. Rakenteellinen arkkitehtuuri määrittää, mistä osista järjestelmä koostuu ja miten ne liittyvät toisiinsa. Toiminnallinen arkkitehtuuri puolestaan kuvaa järjestelmän suorittamat toiminnot ja niiden jakautumisen eri osille. Näitä arkkitehtuurin osia yhdistävät rajapinnat, jotka määrittävät kuinka järjestelmän osat kommunikoivat keskenään ja miten tieto liikkuu niiden välillä. Järjestelmän arkkitehtuuri vaikuttaa suoraan sen suorituskykyyn, laatuun ja toimivuuteen. Yksi mallipohjaisen järjestelmäsuunnittelun keskeisimmistä ominaisuuksista on jäljitettävyyden, joka yhdistää vaatimukset, arkkitehtuurin ja testauksen yhtenäiseksi kokonaisuudeksi. Jäljitettävyydellä tarkoitetaan järjestelmän kykyä seurata vaatimusten toteutumista järjestelmän eri osissa. Vaatimusten jäljitettävyyden voidaan jakaa eteenpäin jäljitettävyyteen ja taaksepäin jäljitettävyyteen. Eteenpäin jäljitettävyydellä tarkoitetaan vaatimusten seuraamista suunnitteluratkaisuihin ja testaukseen asti, kun taas taaksepäin jäljitettävyyden mahdollistaa vaatimusten yhdistämisen takaisin niiden alkuperään. Mallinnympäristössä jäljitettävyyden toteutetaan kohdistus- ja toteutussuhteiden avulla. Näiden avulla vaatimukset voidaan tunnistaa, miten järjestelmän osia vaatimukset koskevat. [2], [6]

Mallipohjaisessa järjestelmäsuunnittelussa järjestelmän kuvaamiseen käytetään SysML-mallinnuskieltä. SysML mahdollistaa järjestelmän kuvaamisen mallien avulla ja toimii keskeisenä työkaluna erilaisten näkökulmien mallintamisessa. Sen avulla voidaan tarkastella järjestelmän rakennetta, toimintaa ja vaatimuksia. Vaatimusdiagrammeilla kuvataan järjestelmän vaatimuksia. Niiden avulla voidaan seurata, mitä järjestelmän tulee tehdä sekä yhdistää vaatimukset järjestelmän muihin osiin. Lohkomäärittelydiagrammi ja sisäinen lohkokaavio, kuvaavat järjestelmän rakennetta ja sen osien välisiä suhteita. Lohkomäärittelydiagrammi kuvaa järjestelmän osat eli lohkot, niiden väliset suhteet sekä järjestelmän hierarkian. Hierarkia voidaan esittää koostumussuhteiden ja periytyksen avulla.

Sisäinen lohko-kaavio puolestaan kuvaa, kuinka järjestelmän osat ovat yhteydessä toisiinsa ja miten tieto, materia tai energia liikkuu niiden välillä. Toimintadiagrammit kuvaavat järjestelmän toimintaa ja käyttäytymistä. Ne voidaan jakaa aktiviteetti- ja sekvenssidiagrammeihin sekä tilakonekaavioihin. Aktiviteettidiagrammi toimii prosessikaavion tavoin ja kuvaa toimintojen etenemistä, niiden järjestystä sekä mahdollisia rinnakkaisia toimintoja järjestelmässä. Sekvenssidiagrammi kuvaa järjestelmän osien välistä vuorovaikutusta ajan suhteen ja näyttää, missä järjestyksessä viestintä tapahtuu sekä mitä tietoa osien välillä välitetään. Tilakonekaavio puolestaan kuvaa järjestelmän tai sen osan tiloja ja niiden välisiä siirtymiä. Eri diagrammit ovat keskeinen osa mallipohjaista järjestelmäsuunnittelua, sillä niiden avulla voidaan hallita monimutkaisia järjestelmiä ja seurata järjestelmän tilaa eri näkökulmista. Mallipohjainen lähestymistapa auttaa vähentämään epäselvyyksiä sekä parantaa järjestelmän hallittavuutta ja ymmärrettävyyttä. [2], [6]

2.3 Suunnittelumenetelmien erot

Modernit järjestelmät ovat monimutkaisia, ja on tärkeää, että on luotettavia ja erilaisia lähestymistapoja, joita yleistyneet mallipohjaiset suunnittelu menetelmät edustavat. Kehityksessä edetään perinteisistä dokumentaatiomalleista uuden ajan mallipohjaisiin järjestelmiin. Mallipohjainen suunnittelu ja mallipohjainen järjestelmäsuunnittelu molemmat perustuvat malleihin, ja ovat alkuperäisiltä termeiltään hyvin saman kaltaisia, ja kirjallisuudessa, niitä käytetään helposti epä johdonmukaisesti, ja eri lähestymistapojen roolit eivät ole aina selkeästi eroteltuja. Vaikka molemmilla suunnittelumenetelmillä on paljon samankaltaisuuksia, ne ei eivät ole sama asia, sillä ne toimivat hyvin eri tasoilla, mutta ne pystyvät hyvin täydentämään toisiaan. [1, 2]

Sekä mallipohjainen suunnittelu että mallipohjainen järjestelmäsuunnittelu perustuvat mallien hyödyntämiseen ja mallipohjaiseen ajatteluun, mutta niiden lähestymistavat eroavat merkittävästi. Mallipohjainen järjestelmäsuunnittelu keskittyy järjestelmätason suunnitteluun, mikä sisältää vaatimukset, arkkitehtuurin ja järjestelmärakenteen. Sen tavoitteena on hallita kompleksisuutta ja varmistaa kokonaisuuden toimivuus. Näin ollen mallipohjaisessa järjestelmäsuunnittelussa ei keskitytä yksityiskohtaiseen toteutukseen. Mallipohjainen suunnittelu keskittyy järjestelmän käyttäytymiseen, eli käsitellään algoritmeja, simulointia ja

testausta. Tavoitteena on toiminnallisuuksien varmistaminen ja totutuksen mahdollistaminen. Mallipohjaisessa suunnittelussa keskitytään enemmän koodiin ja sen toteutukseen. Mallipohjainen järjestelmä suunnittelu tarkastelee järjestelmää rakenteellisesta näkökulmasta keskittyen vaatimuksiin ja arkkitehtuuriin, kun taas mallipohjainen suunnittelu keskittyy dynaamiseen käyttäytymiseen, simulointiin ja toteutukseen. Tämän vuoksi mallipohjainen järjestelmäsuunnittelu toimii korkeammalla abstraktiotasolla ja mallipohjainen suunnittelu käytännön tasolla. [1, 2, 4]

Nämä molemmat lähestymistavat perustuvat mallien hyödyntämiseen, mutta mallien rooli ja käyttötarkoitukset eroavat merkittävästi. Mallipohjainen järjestelmäsuunnittelu käyttää rakenteellisia ja loogisia malleja. Mallit kuvaavat järjestelmän osat, suhteet, rajapinnat ja vaatimukset.[1, 5] Mallit auttavat ymmärtämään kokonaisuutta ja tukevat suunnittelua ja kommunikaatiota. Mallipohjaisen järjestelmäsuunnittelun mallit keskittyvät järjestelmän rakenteen ja suhteiden kuvaamisella korkealla abstraktitasolla. Mallipohjainen suunnittelu puolestaan käyttää dynaamisia ja simuloitavia malleja, ja mallit kuvaavat järjestelmän toimintaa, algoritmeja, sekä signaaleja ja dynamiikka. [2, 6] Tämän menetelmän mallia voidaan ajaa simulaatiossa ja näin ollen testata toimivuutta. Mallipohjaisen suunnittelun mallit kuvaavat järjestelmän käyttäytymistä ja mahdollistavat simuloinnin sekä analyysin. Mallipohjaisen järjestelmä- ja mallipohjaisen suunnittelun mallit eroavat toisistaan erityisesti niiden tarkoituksen ja suoritettavuuden osalta. Mallipohjaisessa järjestelmäsuunnittelussa keskitytään järjestelmän rakenteen ja sen osien välisten suhteiden kuvaamiseen, minkä vuoksi ne eivät yleensä ole suoritettavia. Sen sijaan mallipohjaisessa suunnittelussa mallit kuvaavat järjestelmän dynaamista käyttäytymistä ja ovat suoritettavia, mikä mahdollistaa simuloinnin ja analyysin. Näin ollen mallipohjaisen järjestelmäsuunnittelun mallit tukevat ensisijaisesti järjestelmän suunnittelua ja ymmärtämistä, kun taas mallipohjaisen suunnittelun mallit keskittyvät toiminnan analysointiin ja toteutukseen. Mallipohjainen järjestelmäsuunnittelu hyödyntää rakenteellista tietoa, ja tulkitsee mitä osia järjestelmässä on ja miten ne liittyvät toisiinsa, sekä mitä vaatimuksia niillä on, kun taas mallipohjainen suunnittelu keskittyy toiminnalliseen tietoon ja toimivuuteen, kuten mitä laskentaa tapahtuu ja miten signaalit muuttuvat. Mallipohjaisesta järjestelmäsuunnittelua käytetään aikaisessa kehityksessä, kun kaikki ei ole vielä tiedossa. Malli voi olla yleisellä tasolla, sillä mallipohjainen järjestelmäsuunnittelu sietää epävarmuutta, mutta mallipohjainen suunnittelu taas ei, sillä sitä

käytetään myöhäisemmässä vaiheessa, koska järjestelmä tarvitsee tarkat arvot ja mallit. Jotta mallia voidaan simuloida oikein. Mallipohjainen järjestelmäsuunnittelu keskittyy järjestelmätason kompleksisuuden hallintaan järjestelmän rakenteen ja riippuvuudet. Mallipohjainen suunnittelu puolestaan keskittyy toiminnallisen kompleksisuuden hallintaan analysoimalla järjestelmän käyttäytymistä ja algoritmeja. [1, 2]

Niin mallipohjaista suunnittelua kuin mallipohjaista järjestelmäsuunnittelua käytetään monimutkaisten järjestelmien kehityksessä, mutta niiden käyttötarkoitukset eroavat merkittävästi kehitysprosessin eri vaiheissa. Mallipohjainen järjestelmäsuunnittelu hyödynnetään erityisesti tilanteissa, joissa järjestelmän kokonaisuuden hallinta, vaatimusten jäljitettävyys ja arkkitehtuurin suunnittelu ovat keskeisiä. [1], [2] Mallipohjaista suunnittelua käytetään erityisesti tilanteissa, joissa järjestelmän toiminnan simulointi, analysointi ja toteutuksen validointi ovat keskeisiä [4], [8]. Mallipohjainen järjestelmäsuunnittelu painottuu kehitysprosessin alkuvaiheeseen, jossa määritellään järjestelmän rakenne ja vaatimukset, kun taas mallipohjainen suunnittelu painottuu myöhempisiin vaiheisiin, joissa järjestelmän toiminta toteutetaan ja validoidaan. Näin ollen mallipohjainen järjestelmäsuunnittelu ja mallipohjainen suunnittelu eivät ole vaihtoehtoisia lähestymistapoja, vaan ne täydentävät toisiaan eri vaiheissa järjestelmän kehitysprosessia. [1, 2]

Mallipohjainen suunnittelu ja mallipohjainen järjestelmäsuunnittelu sisältävät verifiointi- ja validointivaiheita, mutta niiden lähestymistavat eroavat merkittävästi riippuen kehitysvaiheesta ja mallin tarkoituksesta. Mallipohjainen järjestelmäsuunnittelu keskittyy varmistamaan, että järjestelmä täyttää sille asetetut vaatimukset ja että sen arkkitehtuuri loogisesti toimiva ennen toteutusvaihetta[2]. Mallipohjainen suunnittelu mahdollistaa järjestelmän toiminnan testaamisen simuloinnin avulla ennen fyysistä toteutusta, mikä vähentää virheitä ja kehityskustannuksia[3, 5]. Mallipohjaisen järjestelmäsuunnittelun ja mallipohjaisen suunnittelun verifiointimenetelmät eroavat toisistaan erityisesti kohteen ja ajankohdan osalta. Mallipohjainen järjestelmäsuunnittelu keskittyy järjestelmän suunnittelun arviointiin ennen toteutusvaihetta varmistaen, että vaatimukset ja arkkitehtuuri ovat johdonmukaisia, kun taas mallipohjainen suunnittelu pohjautuu järjestelmän toiminnan testaamiseen toteutusvaiheessa simuloinnin avulla. Mallipohjaisen järjestelmäsuunnittelun tavoitteena on varmistaa oikean järjestelmän suunnittelu, kun taas mallipohjaisen suunnittelun tavoitteena varmistaa

järjestelmän toimivuus käytännössä. Mallipohjaisen järjestelmäsuunnittelun vähentää väärän järjestelmän suunnittelun riskiä, kun taas mallipohjaisen suunnittelu vähentää virheellisen toteutuksen riskiä. Mallipohjaisen järjestelmäsuunnittelu ja mallipohjainen suunnittelu täydentävät toisiaan testauksen näkökulmasta, sillä toinen varmistaa järjestelmän oikullisuuden suunnitteluvaiheessa ja toinen sen toimivuuden toteutusvaiheessa. [5], [6]

Mallipohjainen suunnittelu ja mallipohjainen järjestelmäsuunnittelu eivät ole vaihtoehtoisia menetelmiä, vaan ne muodostavat yhdessä yhtenäisen kehitysprosessin. Vaikka mallipohjainen suunnittelu ja mallipohjainen järjestelmäsuunnittelu esitetään usein erillisinä lähestymistapoina, ne eivät ole toisiaan poissulkevia, vaan toimivat rinnakkain osana samaa kehitysprosessia. Mallipohjainen järjestelmäsuunnittelu määrittelee järjestelmän rakenteen, vaatimukset ja arkkitehtuurin, kun taas Mallipohjainen suunnittelu keskittyy näiden määrittelyjen toteuttamiseen, simulointiin ja testaamiseen. Mallipohjaisen järjestelmäsuunnittelun tuottamat vaatimukset ja arkkitehtuurimallit ohjaavat mallipohjaisen suunnittelun simulointia ja toteutusta, kun taas mallipohjaisen suunnittelun tuottamat testitulokset ja analyysit tarjoavat palautetta järjestelmätason suunnitteluun. Näiden lähestymistapojen yhdistäminen parantaa jäljitettävyyttä vähentää virheitä ja mahdollistaa tehokkaamman kehitysprosessin, jossa järjestelmätaso suunnittelu ja toteutustason analyysi tukevat toisiaan. Yhdessä mallipohjainen suunnittelu ja mallipohjainen järjestelmäsuunnittelu mahdollistavat iteratiivisen kehitysprosessin, jossa järjestelmätason suunnittelu ja toteutustason analyysi kehittyvät rinnakkain. Näin ollen mallipohjainen suunnittelu ja mallipohjainen järjestelmäsuunnittelu muodostavat yhtenäisen mallipohjaisen kehitysparadigman, jossa järjestelmän suunnittelu ja toteutus yhdistyvät saumattomasti. [1], [2], [6]

Tässä luvussa tarkasteltiin mallipohjaisen suunnittelun ja mallipohjaisen järjestelmäsuunnittelun keskeisiä eroja ja yhtäläisyyksiä. Keskeinen ero lähestymistapojen välillä liittyy niiden tarkastelutasoon. Mallipohjainen järjestelmäsuunnittelu keskittyy järjestelmän rakenteeseen, vaatimuksiin ja arkkitehtuuriin, kun taas mallipohjainen suunnittelu keskittyy järjestelmän toimintaan, simulointiin ja toteutukseen. Molemmat lähestymistavat perustuvat mallien hyödyntämiseen ja pyrkivät parantamaan järjestelmien kehityksen tehokkuutta sekä vähentämään virheitä. Lähestymistavat eivät ole toisiaan poissulkevia, vaan

ne täydentävät toisiaan eri vaiheissa järjestelmän kehitysprosessia. Mallipohjainen suunnittelu ja mallipohjainen järjestelmäsuunnittelu muodostavat yhdessä yhtenäisen mallipohjaisen kehitysparadigman, jossa järjestelmätason suunnittelu ja toteutustason kehitys tukevat toisiaan. [1, 2, 5, 8]

3 Mallipohjainen suunnittelu eri aloilla

3.1 Autoteollisuus

Autoteollisuudessa on ollut jo jonkin aikaan käynnissä muutos. Muutoksella tarkoitetaan, että teollisuus on siirtynyt mekaanisista tuotteista ohjelmistointensiivisiksi järjestelmiksi. Mekaaniset järjestelmät ovat tuotteita, mitkä sisältävät vähän elektroniikka, kuten moottori, jarrut, ja vaihteisto. Nykyajan autot ovat monimutkaisia ohjelmistojärjestelmien kokonaisuus, eli autot sisältävät kymmeniä ohjauslaitteita, satoja antureita ja miljoonia rivejä koodia. [13]

Modernia ajoneuvoa voidaan kuvata järjestelmien järjestelmänä, jossa useat itsenäiset, mutta keskenään vuorovaikutuksessa olevat alijärjestelmät muodostavat kokonaisuuden. Tämän toiminta perustuu niiden väliseen yhteistyöhön [6, 11]. Ajoneuvojen kompleksisuus ei synny pelkästään komponenttien määrästä, vaan erityisesti alijärjestelmien välisten rajapintojen, datavirtojen ja riippuvuuksien lisääntymisestä [3, 12]. Mallipohjainen järjestelmäsuunnittelu mahdollistaa koko ajoneuvon arkkitehtuurin kuvaamisen yhtenäisenä mallina, mikä helpottaa järjestelmän rakenteen ja toiminnallisten riippuvuuksien ymmärtämistä [11, 13]. mallipohjaisen järjestelmäsuunnittelun avulla järjestelmien väliset rajapinnat voidaan määritellä tarkasti, mikä vähentää integraatiovirheitä ja parantaa eri alijärjestelmien yhteensopivuutta [13, 14]. Mallipohjainen lähestymistapa mahdollistaa järjestelmien välisten riippuvuuksien systemaattisten analysoinnin, mikä auttaa tunnistamaan kriittiset kohdat jo suunnitteluvaiheissa [15, 16]. Mallipohjainen järjestelmäsuunnittelu mahdollistaa koko järjestelmien käyttäytymisen simuloinnin jo kehityksen alkuvaiheissa, mikä vähentää myöhemmässä vaiheessa ilmenevien virheiden riskiä [3, 5]. Perinteisesti ajoneuvojen arkkitehtuuri on ollut hajautettu, koostuen useista ohjausyksiköistä, mikä lisää järjestelmien välisten vuorovaikutusten määrä ja kompleksisuutta [10, 11]. Autoteollisuudessa useat alijärjestelmät kehitetään eri toimittajien toimesta, mikä korostaa yhtenäisten mallien merkitystä yhteisen ymmärryksen ja integraation varmistamisessa [12, 13]. Järjestelmien järjestelmä rakenteessa voi ilmetä emigranttia käyttäytymistä, jossa kokonaisuuden toiminta ei ole suoraan pääteltävissä yksittäisten alijärjestelmien toiminnassa [6, 10]. Ajoneuvojen kehitys kohti autonomisia ja verkottuneita järjestelmiä lisää entisestään järjestelmätason

kompleksisuuta, mikä tekee mallipohjaisesta suunnittelusta keskeisen työkalun tulevaisuuden kehityksessä [11, 12].

Autoteollisuudessa mallipohjainen kehitys siirtää suunnittelun painopisteistä koodikeskeisestä lähestymistavasta kohti järjestelmämallien hyödyntämistä, joissa järjestelmän toiminta rakenne kuvataan abstraktiolla ennen toteutusta [1], [7]. Mallit toimivat keskeisenä välineenä järjestelmät toiminnallisuuden, käyttäytymisen ja rakenteen kuvaamisessa, mahdollistaen eri näkökulmien yhdistämisen samaan kokonaisuuteen [7, 9]. Mallipohjainen kehitys tukee monialaista suunnittelua yhdistämällä mekaaniset, sähköiset ja ohjelmistotekniset näkökulmat yhdeksi yhtenäiseksi järjestelmäksi [8, 13]. Mallipohjaisessa kehityksessä järjestelmämalleista voidaan generoida automaattisesti ohjelmakoodi, mikä vähentää manuaalisen ohjelmoinnin virheitä ja parantaa kehitysprosessin tehokkuutta [2, 10]. Mallien avulla voidaan simuloida järjestelmän toimintaa jo ennen fyysistä toteutusta, mikä mahdollistaa suunnitteluratkaisujen arvioinnin ja optimoinnin varhaisessa vaiheessa [3, 5]. Mallipohjainen lähestymistapa tukee iteratiivista kehitystä, jossa malleja voidaan jatkuvasti tarkentaa ja validoida kehitysprosessin edetessä [1, 7]. Mallipohjainen kehitys mahdollistaa virheiden tunnistamisen jo suunnitteluvaiheessa, mikä vähentää myöhemmissä kehitysvaiheissa ilmenevien ongelmien määrää [12, 16]. Mallit mahdollistavat vaatimusten, suunnitteluratkaisujen ja toteutuksen välisen jäljitettävyyden, mikä parantaa järjestelmän hallittavuutta ja dokumentointia [6, 9]. Yhtenäiset järjestelmämallit helpottavat eri vaalijärjestelmien integraatiota, koska ne tarjoavat yhteisen viitekehyksen järjestelmien väliselle kommunikaatiolle [13, 14]. Mallipohjaisen kehityksen keskeisiä hyötyjä autoteollisuudessa ovat lyhyempi kehitysaika, parempi laatu sekä monimutkaisempi järjestelmien hallinnan parantuminen [10, 12].

Autoteollisuudessa simulointi on keskeinen osa mallipohjaista kehitystä, sillä sen avulla ajoneuvon järjestelmien toimintaa voidaan arvioida ennen fyysisten prototyyppien rakentamista [3]. Mallisimulointivaiheessa ajoneuvon järjestelmämallia, kuten ohjausalgoritmeja, testataan simulaatioympäristössä, mikä mahdollistaa suunnitteluratkaisujen varhaisen validoinnin [2]. Ohjelmistosimulointi testauksessa järjestelmämallista generoitu ohjelmakoodi suoritetaan simulaatioympäristössä, jolloin voidaan varmistaa koodin toiminnallinen vastaavuus mallin kanssa [10]. Laitteistosimulointitestauksessa ajoneuvon ohjausyksiköitä testataan yhdistämällä ne simuloituvat ympäristöön, mikä mahdollistaa realististen käyttötilanteiden

arvioinnin ilma fyysistä ajoneuvoa[3]. Autoteollisuudessa mallisimulointi-, ohjelmistosimulointi- ja laitteistosimulointivaiheet muodostavat vaiheittaisen testausprosessin, jossa järjestelmää validoidaan asteittain mallitasolta kohti fyysistä toteutusta[12]. Mallipohjainen simulointi mahdollistaa virheiden, kuten ohjausalgoritmien virheellisen toiminnan, tunnistamisen jo ennen fyysistä testausta, mikä vähentää kehityskustannuksia. Simulointiin perustuva testaus vähentää fyysisten prototyyppien tarvetta, mikä lyhentää ajoneuvojen kehitysaikaa ja pienentää kustannuksia [12, 14]. Autoteollisuudessa simulointia hyödynnetään erityisesti turvallisuuskriittisten järjestelmien, kuten jarru- ja ajonvakautusjärjestelmien, testaamiseen ennen niiden käyttöönottoa[10]. Erityisesti Laitteistosimuloinnissa simulaatioiden tulee toimia reaaliaikaisesti, jotta ajoneuvon ohjausyksiköiden käyttäytyminen vastaa todellisia käyttöolosuhteita[3]. Mallisimulointi-, ohjelmistosimulointi- ja laitteistosimulointimenetelmät ovat keskeisiä työkaluja autoteollisuuden mallipohjaisessa kehityksessä sillä ne mahdollistavat järjestelmien kattava testauksen ennen tuotantovaiheita [12]. [8]

Autoteollisuudessa turvallisuus on keskeinen suunnittelutekijä, sillä ajoneuvojen järjestelmien virheet voivat johtaa vakaviin onnettomuuksiin ja henkilövahinkoihin[6, 10]. Monet ajoneuvon järjestelmät, kuten jarrujärjestelmät ja ajonvakautus, ovat turvallisuuskriittisiä, mikä edellyttää, niiden toiminnan huolellista analysointia ja validointia kehityksen aikana[5, 10]. Autoteollisuudessa turvallisuutta ohjaavat standardit, määrittelevät vaatimukset järjestelmien toteutukselle ja testaukselle[14]. Mallipohjainen järjestelmäsuunnittelu tukee turvallisuuskriteereitä järjestelmän kehitystä mahdollistamalla. Vaatimusten, toimintojen ja riskien mallintamisen yhtenäisessä ympäristössä. [9, 10] Mallipohjainen järjestelmäsuunnittelu mahdollistaa riskien tunnistamisen jo suunnitteluvaiheessa analysoimalla järjestelmien välisiä riippuvuuksia ja mahdollisia virhetilanteita. [15, 16] Mallipohjainen lähestymistapa helpottaa turvallisuusvaatimusten hallintaa ja jäljitettävyyttä koko ajoneuvon kehitysprosessin ajoneuvon kehitysprosessin ajan [6, 9]. Mallipohjainen järjestelmäsuunnittelu tukee järjestelmien verifiointi ja validointia tarjoamalla mahdollisuuden testata turvallisuuskriittisiä toimintoja simulaatioiden avulla ennen fyysistä toteutusta[5, 3]. Mallipohjainen kehitys vähenee turvallisuusriskien todennäköisyyttä mahdollistamalla virheiden havaitseminen varhaisessa kehitysvaiheessa [10, 12]. Ajoneuvojen järjestelmien monimutkaisuuden kasvaessa mallipohjaiset menetelmät ovat keskeisiä turvallisuuden varmistamisessa koko

järjestelmätasolla [11, 12]. Autonomisten ja ohjelmistointensiiviset ajoneuvojen yleistyessä mallipohjaisen suunnittelun merkitys turvallisuuden varmistamisessa korostuu entisestään [10, 12].

Autoteollisuudessa ajoneuvojen kehitys edellyttää mekaanisten, sähköisten ja ohjelmistoteknisten järjestelmien tiivistä yhteensovittamista, mikä tekee kehityksestä luonteeltaan monialaista [19]. Eri suunnittelualojen ja toimittajien osallistuminen ajoneuvon kehitykseen lisää integraatiohaasteista, erityisesti järjestelmien rajapintojen ja yhteensopivuuden osalta [12, 13]. Mallipohjainen järjestelmäsuunnittelu tukee integraatiota tarjoamalla yhtenäisen mallin, jossa eri alijärjestelmien väliset suhteet ja rajapinnat voidaan kuvata selkeästi [3]. Mallipohjainen järjestelmäsuunnittelu mahdollistaa eri suunnittelutiimien välisen kommunikaation tarjoamalla yhteisen mallipohjaisen esitystavan, joka vähentää väärinymmärryksiä kehitysprosessien aikana [15]. Ajoneuvojen alijärjestelmien rajapintojen tarkka mallintaminen on keskeistä onnistuneessa integraatiossa, sillä se varmistaa järjestelmien yhteensopivan toiminnan [16]. Autoteollisuudessa alijärjestelmät kehitetään usein eri toimittajien toimesta, mikä korostaa standardoitujen mallien merkitystä integraation hallinnassa [14]. Mallipohjainen järjestelmäsuunnittelu mahdollistaa järjestelmien integraation testaamisen jo varhaisessa vaiheessa simuloinnin avulla, mikä vähentää myöhemmässä vaiheessa ilmeneviä yhteensopivuusongelmia [8]. Mallipohjaiset työkalut luovat digitaalisen ympäristön, jossa ajoneuvon eri järjestelmien yhteistoimintaa voidaan analysoida ilman fyysistä prototyyppiä [3]. Integraation tehostuminen mallipohjaisen suunnitteluun avulla lyhentää ajoneuvojen kehitysaikaa ja parantaa lopputuotteen laatua [14]. Ajoneuvojen ohjelmistointensiivisyyden kasvaessa integraation merkitys korostuu entisestään, mikä lisää mallipohjaisen suunnittelun roolia kehitysprosessissa [13].

Autoteollisuudessa ajoneuvojen kasvava ohjelmistointensiivisyys ja järjestelmien monimutkaisuus lisäävät merkittävästi kehitysprosessin vaativuutta ja kustannuksia [14]. Perinteisessä kehitysprosessissa virheet havaitaan usein vasta myöhäisessä vaiheessa, mikä johtaa kalliisiin muutoksiin ja viivästyksiin ajoneuvojen kehityksessä [18]. Mallipohjainen järjestelmäsuunnittelu lyhentää ajoneuvojen kehitysaikaa mahdollistamalla järjestelmien varhaisen analysoinnin ja validoinnin [14]. Mallipohjainen järjestelmäsuunnittelu vähentää kehityskustannuksia, koska virheet voidaan tunnistaa ja korjata ennen fyysisten prototyyppien

rakentamista[8]. Simulointiin perustuva kehitys vähentää fyysisten prototyyppien tarvetta autoteollisuudessa, mikä pienentää materiaalikustannuksia ja testauksen kestoa[16]. Mallipohjainen kehitys tukee iteratiivista suunnittelua, jossa ajoneuvon järjestelmiä voidaan optimoida vaiheittain ilman merkittäviä lisäkustannuksia [1]. Mallipohjainen järjestelmäsuunnittelu parantaa ajoneuvojen järjestelmien laatua mahdollistamalla kattavamman testauksen ja analyysin jo suunnitteluvaiheessa [13]. Mallipohjainen suunnittelu vähentää integraatiokustannuksia, koska järjestelmien yhteensopivuus voidaan varmistaa ennen niiden fyysistä yhdistämistä [15]. Mallipohjainen järjestelmäsuunnittelu mahdollistaa kehitysresurssien tehokkaamman käytön, koska suunnittelupäätöksiä voidaan tehdä mallien perusteella ilman laajoja fyysisiä testejä [3]. Kehityksen tehostuminen ja kustannusten pieneneminen parantavat autoteollisuuden yritysten kilpailukykyä nopeasti kehittyville markkinoilla [14].

3.2 Ilmailu ja avaruustekniikka

Ilmailu- ja avaruustekniikassa järjestelmien suunnittelua ohjaavat erittäin korkeat turvallisuusvaatimukset, sillä pienetkin virheet voivat johtaa vakaviin seurauksiin [17]. Ilmailualalla järjestelmien tulee täyttää tiukat sertifiointivaatimukset, mikä edellyttää kattavaa dokumentointia ja järjestelmällistä suunnitteluprosessia[2]. Järjestelmien korkea luotettavuus varmistetaan usein redundanssille, jossa kriittiset toiminnot toteutetaan useiden rinnakkaisten järjestelmien avulla [17]. Ilmailujärjestelmillä on tyypillisesti pitkä elinkaari, mikä asettaa vaatimuksia järjestelmien ylläpidettävyydelle ja päivitettävyydelle[6]. Ilmailujärjestelmien korkea kompleksisuus edellyttää järjestelmätason tarkastelua, jossa kaikki alijärjestelmät ja niiden väliset riippuvuudet huomioidaan [2], [17]. Ilmailualalla järjestelmien kehitys on kallista ja riskialtista, mikä korostaa virheiden varhaisen havaitsemisen merkitystä [17]. Näiden vaatimusten vuoksi mallipohjainen järjestelmäsuunnittelu on muodostunut keskeiseksi lähestymistavaksi ilmailualan monimutkaisten järjestelmien hallinnassa [9, 15].

Lentokone voidaan kuvata järjestelmien järjestelmänä, jossa useat itsenäiset, mutta keskenään vuorovaikuttavat alijärjestelmät muodostavat toiminnallisen kokonaisuuden[6, 15]. Lentokoneen keskeisiin alijärjestelmiin kuuluvat muun muassa ohjausjärjestelmät, propulsiojärjestelmät sekä avioniikkajärjestelmät, joiden yhteistoiminta on kriittistä koko

järjestelmän toiminnalle [17]. Ilmailujärjestelmien kompleksisuus syntyy erityisesti alijärjestelmien välisten rajapintojen ja riippuvuuksien määrästä ei pelkästään yksittäisten komponenttien lukumäärästä [14]. Alijärjestelmien välinen vuorovaikutus edellyttää tarkkaa synkronointia sillä yhden järjestelmän toiminta voi vaikuttaa suoraan muiden järjestelmien suorituskykyyn [17]. Järjestelmien järjestelmä rakenteessa voi ilmetä kehittyvää käyttäytymistä, jossa koko järjestelmän toiminta ei ole suoraan ennustettavissa yksittäisten alijärjestelmien perusteella [2]. Lentokoneen järjestelmien integraatio on keskeinen haaste, sillä alijärjestelmien on toimittava saumattomasti yhdessä kaikissa käyttötilanteissa [17]. Järjestelmien kasvava kompleksisuus lisää kehityksen vaativuutta ja korostaa järjestelmätason suunnittelun merkitystä ilmailualalla [14]. Mallipohjainen järjestelmäsuunnittelu tarjoaa keinon hallita ilmailujärjestelmien kompleksisuutta mallintamalla alijärjestelmien väliset suhteet ja riippuvuudet yhtenäisessä ympäristössä [9, 15].

Ilmailu- ja avaruustekniikassa mallipohjainen järjestelmäsuunnittelu on keskeinen menetelmä monimutkaisessa järjestelmien kehittämisessä ja hallinnassa [9, 15]. Mallipohjainen järjestelmäsuunnittelu mahdollistaa lentokoneen järjestelmäarkkitehtuurin kuvaamisen yhtenäisenä mallina, jossa eri alijärjestelmät ja niiden väliset suhteet voidaan esittää selkeästi [17]. Mallipohjainen lähestymistapa tukee vaatimusten hallintaa mahdollistamalla vaatimusten liittämisen suoraan järjestelmämalleihin [6]. Ilmailualalla Mallipohjainen järjestelmäsuunnittelu parantaa jäljitettävyyttä yhdistämällä vaatimukset, suunnitteluratkaisut ja testauksen tulokset yhtenäiseen mallipohjaiseen kokonaisuuteen [6, 9]. Mallipohjainen järjestelmäsuunnittelu tarjoaa keinot hallita ilmailujärjestelmien monimutkaisuutta mallintamalla alijärjestelmien väliset riippuvuudet ja vuorovaikutukset [17]. Mallipohjainen suunnittelu mahdollistaa riskien tunnistamisen jo varhaisessa kehitysvaiheessa analysoimalla järjestelmien käyttäytymistä mallien avulla [17]. Mallipohjainen järjestelmäsuunnittelu tukee alijärjestelmien integraatiota tarjoamalla yhteisen mallin, jonka avulla eri järjestelmien yhteensopivuus voidaan varmistaa ennen toteutusta [6]. Ilmailualalla mallipohjainen järjestelmäsuunnittelu mahdollistaa järjestelmien hallinnan koko elinkaaren ajan suunnittelusta ylläpitoon asti [2]. Mallipohjainen järjestelmäsuunnittelu tukee ilmailualan sertifiointiprosesseja tarjoamalla rakenteellisen ja jäljitettävän tavan dokumentoida järjestelmän kehitys [6]. Ilmailujärjestelmien kompleksisuuden kasvaessa mallipohjaisen

järjestelmäsuunnittelun merkitys korostuu keskeisenä työkaluna järjestelmien hallinnassa ja kehityksessä [17].

Ilmailu- ja avaruustekniikassa simulointi on keskeinen osa järjestelmien kehitystä, sillä sen avulla voidaan analysoida järjestelmien toimintaa ennen kalliita ja riskialttiita fyysisiä testejä [17]. Mallipohjaisen suunnittelun avulla lentokoneen järjestelmien toimintaa voidaan simuloida jo kehityksen alkuvaiheessa, mikä mahdollistaa suunnitteluratkaisuja arvioinnin varhaisessa vaiheessa[6]. Simulointi mahdollistaa koko lentokoneen järjestelmätason analyysin, jossa alijärjestelmien väliset vuorovaikutukset voidaan ottaa huomioon [17]. Simulointien avulla voidaan tunnistaa mahdolliset virhetilanteita ja riskejä ennen fyysisiä testejä, mikä vähentää kehitykseen liittyvää epävarmuutta [17]. Ilmailualallasimulointia hyödynnetään erityisesti turvallisuuskriittisten järjestelmien analysointiin, jotta toiminta voidaan varmistaa kaikissa käyttötilanteissa [2]. Simulointi mahdollistaa erilaisten käyttö- ja vikatilanteiden testaamisen, joita olisi vaikea tai vaarallista toteuttaa fyysisissä testeissä[3]. Simulointiin perustuva analyysi vähentää kehityskustannuksia ja riskejä, koska fyysisten testien määrää voidaan minimoida [17]. Mallipohjainen järjestelmäsuunnittelu mahdollistaa simulointimallien integroinnin osaksi järjestelmämallia, mikä parantaa analyysin kattavuutta ja yhtenäisyyttä[6]. Simulointi tukee järjestelmien validointia ennen fyysisiä testejä, mikä on erityisen tärkeää ilmailualan tiukkojen turvallisuusvaatimusten vuoksi [2]. Ilmailujärjestelmien kompleksisuuden kasvaessa simuloinnin merkitys korostuu keskeisenä työkaluna järjestelmien analysoinnissa ja kehityksessä [17].

Ilmailu- ja avaruustekniikassa turvallisuus on keskeisin suunnittelua ohjaava tekijä, sillä järjestelmien virheet voivat johtaa vakaviin ja laajamittaisiin seurauksiin [17]. Ilmailualalla järjestelmien tulee täyttää tiukat sertifiointivaatimukset, jotka edellyttävät kattavaa dokumentointia, analyysiä ja testausnäyttöä [2]. Turvallisuutta ohjaavat standardit määrittelevät vaatimuksia järjestelmien suunnittelulle, kehitykselle ja validoineille ilmailualalla [6]. Mallipohjainen järjestelmäsuunnittelu tukee turvallisuutta mahdollistamalla järjestelmien rakenteen ja toiminnan analysoinnin jo kehityksen alkuvaiheessa [9, 15]. Mallipohjainen järjestelmäsuunnittelu mahdollistaa vaatimusten, suunnitteluratkaisujen ja testitulosten jäljitettävyyden, mikä on keskeinen osa ilmailualan sertifiointiprosessia [6, 9]. Mallipohjainen lähestymistapa tukee riskienhallintaa tunnistamalla mahdolliset virhetilanteet ja niiden

vaikutukset jo suunnitteluvaiheessa [17]. Ilmailualalla järjestelmien verifiointi ja validointi ovat keskeisiä vaiheita, joissa varmistetaan, että järjestelmät täyttävät niille asetetut turvallisuusvaatimukset [2]. Sertifiointiprosessi edellyttää yksityiskohtaista dokumentointia, jonka avulla voidaan osoittaa järjestelmän vaatimustenmukaisuus [6]. Järjestelmien kasvava kompleksisuus lisää turvallisuushaasteita, mikä korostaa systemaattisten suunnittelumenetelmien, kuten mallipohjaisen järjestelmäsuunnittelun merkitystä [14]. Uusien teknologioiden käyttöönotto ilmailussa lisää turvallisuusvaatimuksia, mikä vahvistaa mallipohjaisen suunnittelun roolia sertifiointiprosesseissa [17].

Ilmailu- ja avaruustekniikka on jatkuvassa muutoksessa uusien teknologioiden, kuten sähköisten ja vetykäyttöisten lentokoneiden kehittyessä [17]. Uusien teknologioiden käyttöönotto lisää järjestelmien kompleksisuutta, sillä ne tuovat mukanaan uusia alijärjestelmiä ja riippuvuuksia [14]. Järjestelmien monimutkaisuuden kasvaessa mallipohjaisen järjestelmäsuunnittelun merkitys korostuu keskeisenä työkaluna järjestelmien hallinnassa [17]. Esimerkiksi vetykäyttöisten lentokoneiden kehitys edellyttää uusien järjestelmäarkkitehtuurien suunnittelua, mikä lisää mallipohjaisen lähestymistavan tarvetta [17]. Digitaalisten työkalujen ja simulointimenetelmien kehitys mahdollistaa entistä tarkemman järjestelmien analysoinnin ennen fyysistä toteutusta [3]. Autonomisten järjestelmien yleistyminen ilmailussa lisää vaatimuksia järjestelmien suunnittelulle, erityisesti turvallisuuden ja luotettavuuden näkökulmasta [14]. Tulevaisuudessa järjestelmien koko elinkaaren hallinta korostuu, mikä lisää mallipohjaisen suunnittelun merkitystä ylläpidon ja päivitysten näkökulmasta [2]. Uudet teknologiat lisäävät järjestelmien integraatiohaasteita, mikä korostaa yhtenäisten mallien merkitystä eri alijärjestelmien yhteensovittamisessa [15]. Uusien teknologioiden käyttöönotto tuo mukanaan uusia sertifiointihaasteita, jotka edellyttävät entistä tarkempaa järjestelmien mallintamista ja analysointia [17]. Kokonaisuudessaan. Ilmailualana kehitys kohti monimutkaisempia ja ohjelmistointensiivisempiä järjestelmiä vahvistaa mallipohjaisen suunnittelun roolia keskeisenä kehitysmenetelmänä [12, 15].

3.3 Teollisuusautomaatio ja koneenrakennus

Teollisuusautomaatio ja koneenrakennus yhdistävät mekaanisen suunnittelun, elektroniikan ja ohjelmistotekniikan, mikä tekee alasta luonteeltaan monialaisen [16]. Alan kehitystä ohjaa vahvasti digitalisaatio, joka mahdollistaa tuotantojärjestelmien entistä tehokkaamman suunnittelun ja ohjauksen [3]. Teollisuusautomaatio on keskeisessä roolissa teollisuus 4.0 – kehityksessä, jossa tuotantojärjestelmä muuttuvat älykkäiksi ja verkottuneiksi [3]. Modernit tuotantojärjestelmät voidaan kuvata kyberfysisiksi järjestelmiksi, jossa fyysiset prosessit ja digitaaliset ohjausjärjestelmät toimivat tiiviissä vuorovaikutuksessa [3]. Tuotantojärjestelmien monimutkaisuus on kasvanut merkittävästi, mikä johtuu järjestelmien lisääntyneestä automaatiosta ja ohjelmistojen intensiivisyydestä [14]. Teollisuusautomaatiossa järjestelmät ovat yhä enemmän verkottuneita, mikä mahdollistaa reaaliaikaisen tiedonvaihdon eri järjestelmien välillä [3]. Toisin kuin esimerkiksi autoteollisuudessa, teollisuusautomaatiossa keskeistä on tuotantoprosessien suunnittelu ja optimointi [16]. Näiden ominaisuuksien vuoksi mallipohjainen järjestelmäsuunnittelu tarjoaa tehokkaan lähestymistavan teollisuusautomaation monimutkaisten järjestelmien hallintaan [9, 3].

Teollisuusautomaatiossa järjestelmien kompleksisuus syntyy erityisesti mekaanisten, sähköisten ja ohjelmistopohjaisten järjestelmien yhdistämisestä yhdeksi toimivaksi kokonaisuudeksi [17]. Tuotantojärjestelmät koostuvat useista alijärjestelmistä, kuten koneista, ohjausjärjestelmistä ja ohjelmistoista, joiden saumaton integraatio on keskeistä järjestelmän toiminnan kannalta [18]. Järjestelmien väliset rajapinnat ovat keskeinen kompleksisuuden lähde, sillä niiden kautta tapahtuva tiedonsiirto ja ohjaus vaativat tarkkaa määrittelyä [15]. Teollisuusautomaation järjestelmät toimivat usein reaaliaikaisesti, mikä lisää vaatimuksia järjestelmien synkronoinnille ja luotettavuudelle [3]. Verkottuneiden järjestelmien lisääntyminen kasvattaa integraatiohaasteita, koska järjestelmien tulee kommunikoida keskenään luotettavasti ja tehokkaasti [3]. Eri valmistajien komponenttien käyttö lisää yhteensopivuushaasteita, mikä korostaa standardoituja rajapintojen merkitystä [14]. Järjestelmien tuottaman datan määrä kasvaa jatkuvasti, mikä tekee tiedonhallinnasta keskeisen haasteen teollisuusautomaation järjestelmissä [3]. Mallipohjainen järjestelmäsuunnittelu tukee integraation hallintaa tarjoamalla yhtenäisen mallin, jossa järjestelmien väliset suhteet ja rajapinnat voidaan kuvata selkeästi [6]. Mallipohjainen

järjestelmäsuunnittelu mahdollistaa järjestelmien integraation testaamisen jo varhaisessa vaiheessa simuloinnin avulla, mikä vähentää myöhemmin ilmeneviä ongelmia [4, 5]. Järjestelmien kompleksisuuden hallinta on keskeinen tekijä teollisuusautomaation kehityksessä, mikä tekee mallipohjaista menetelmistä erityisen hyödyllisiä [14].

Teollisuusautomaatiossa mallipohjainen järjestelmäsuunnittelu toimii keskeisenä menetelmänä monimutkaisten tuotantojärjestelmien suunnittelussa ja hallinnassa [9, 13]. Mallipohjainen järjestelmäsuunnittelu mahdollistaa mekaanisten, sähköisten ja ohjelmistopohjaisten järjestelmien yhdistämisen yhdeksi yhtenäiseksi malliksi [15]. Mallipohjainen lähestymistapa tukee tuotantojärjestelmien arkkitehtuurin suunnittelua kuvaamalla alijärjestelmien rakenteen ja niiden väliset suhteet [6]. Mallipohjainen järjestelmäsuunnittelu mahdollistaa vaatimusten liittämisen suoraan järjestelmämalleihin, mikä parantaa niiden hallintaa ja jäljitettävyyttä [2]. Mallipohjainen suunnittelu helpottaa eri alijärjestelmien integraatiota tarjoamalla yhteisen viitekehyksen niiden yhteensovittamiselle [15]. Mallipohjainen järjestelmäsuunnittelu mahdollistaa virheiden tunnistamisen jo suunnitteluvaiheessa, mikä vähentää myöhemmässä vaiheessa ilmeneviä integraatio- ja toimintaongelmia [14]. Mallipohjainen järjestelmäsuunnittelu tukee simulointia tarjoamalla tarkkoja järjestelmämalleja, joita voidaan käyttää tuotantoprosessien analysointiin ja optimointiin [2], [6]. Mallit toimivat yhteisenä kommunikointivälineenä eri suunnittelutiimien välillä, mikä parantaa yhteistyötä ja vähentää väärinymmärryksiä [15]. Mallipohjainen järjestelmäsuunnittelu tehostaa kehitysprosessia mahdollistamalla nopeamman suunnittelun ja päätöksenteon mallien pohjalta [14]. Teollisuus 4.0 – ympäristössä mallipohjainen järjestelmäsuunnittelu on keskeinen työkalu, koska se mahdollistaa älykkäiden ja verkottuneiden tuotantojärjestelmien hallinnan [3].

Teollisuusautomaatiossa simulointi on keskeinen työkalu tuotantojärjestelmien suunnittelussa, sillä sen avulla voidaan arvioida järjestelmien toimintaa ennen fyysistä toteutusta [16]. Simulointimallit mahdollistavat tuotantoprosessien analysoinnin ja optimoinnin, mikä parantaa järjestelmien tehokkuutta ja suorituskykyä [3]. Digitaalinen kaksonen on fyysisen järjestelmän virtuaalinen vastine, jonka avulla järjestelmän toimintaa voidaan seurata ja analysoida reaaliaikaisesti [3, 5]. Digitaalinen kaksonen hyödyntää reaaliaikaista dataa, mikä mahdollistaa tuotantojärjestelmien tilan jatkuvan seurannan ja

optimoinnin [3]. Simuloinnin avulla voidaan testata erilaisia käyttötilanteita ja prosessimuutoksia ilman fyysistä järjestelmää, mikä vähentää kehityskustannuksia[16]. Digitaalisen kaksosen avulla tuotantojärjestelmien suorituskykyä voidaan optimoida tunnistamalla pullonkauloja ja tehottomuuksia[3]. Mallipohjainen järjestelmäsuunnittelu tukee simulointia tarjoamalla tarkkoja järjestelmämalleja, jotka toimivat digitaalisten kaksosten perustana [6]. Mallipohjainen suunnittelu yhdistää simuloinnin ja järjestelmäkehityksen, mikä mahdollistaa suunnitteluratkaisuja jatkuvan arvioinnin [15]. Simulointi vähentää tuotantojärjestelmien käyttöönottoon liittyviä riskejä, koska mahdolliset virheet voidaan havaita ennen käyttöönottoa [14]. Teollisuus- 4.0 ympäristössä digitaalinen kaksonen on keskeinen työkalu älykkäiden ja verkottuneiden tuotantojärjestelmien hallinnassa [3].

Teollinen esineiden internet mahdollistaa laitteiden ja järjestelmien verkottumisen, mikä parantaa tuotantojärjestelmien tehokkuutta ja joustavuutta [3]. Teollisuusautomaationjärjestelmät ovat yhä enemmän verkottuneita, jolloin koneet ja laitteet voivat kommunikoida keskenään reaaliaikaisesti [3]. Esineiden internet mahdollistaa reaaliaikaisen tiedonkeruun ja -vaihdon, mikä tukee tuotantoprosessien optimointia ja päätöksentekoa [3]. Verkottuneet tuotantojärjestelmät ovat usein hajautettuja, mikä lisää järjestelmien kompleksisuutta ja integraatiovaatimuksia [14]. Esineiden internetin hyödyntäminen lisää integraatiohaasteita, koska eri järjestelmien tulee toimia yhteensopivasti ja kommunikoida tehokkaasti keskenään [15]. Suurten datamäärien käsittely ja analysointi muodostavat keskeisen haasteen teollisissa esineiden internetjärjestelmissä[3]. Verkottuneiden järjestelmien lisääntyminen korostaa kyberturvallisuuden merkitystä teollisuusautomaation ympäristöissä [3]. Mallipohjainen järjestelmäsuunnittelu auttaa hallitsemaan esineiden internetjärjestelmien kompleksisuutta mallintamalla järjestelmien väliset suhteet ja tiedonkulun yhtenäisessä mallissa [6]. Mallipohjainen suunnittelu tukee eri järjestelmien yhteensopivuutta tarjoamalla selkeän kuvauksen rajapinnoista ja tiedonsiirrosta [15]. Teollinen esineiden internet on keskeinen osa teollisuus-4.0 kehitystä, jossa tuotantojärjestelmät muuttuvat älykkäiksi, verkottuneiksi ja itseohjautuviksi [3].

Mallipohjainen järjestelmäsuunnittelu tehostaa teollisuusautomaation kehitysprosessia mahdollistamalla järjestelmien analysoinnin ja optimoinnin jo suunnitteluvaiheessa [14]. Mallipohjainen järjestelmäsuunnittelu vähentää kehityskustannuksia, koska virheet voidaan

tunnistaa ja korjata ennen fyysisten järjestelmien toteutusta [14]. Mallipohjainen lähestymistapa parantaa tuotantojärjestelmien laatua mahdollistamalla kattavan analyysin ja testauksen ennen käyttöönottoa [18]. Mallipohjainen järjestelmäsuunnittelu helpottaa eri alijärjestelmien integraatiota tarjoamalla yhtenäisen mallin, jonka avulla järjestelmien yhteensopivuus voidaan varmistaa [15]. Mallipohjainen suunnittelu tukee modulaarista tuotekehitystä, jossa järjestelmät voidaan rakentaa standardoiduista ja uudelleenkäytettävistä komponenteista [18]. Teollisuus-4.0 kehityksen myötä tuotantojärjestelmät muuttuvat älykkäiksi ja itseohjautuviksi, mikä korostaa mallipohjaisen suunnittelun merkitystä [4]. Autonomisten tuotantojärjestelmien yleistyminen lisää vaatimuksia järjestelmien suunnittelulle ja hallinnalle, mikä vahvistaa mallipohjaisen järjestelmäsuunnittelun rooli [14]. Järjestelmien kompleksisuuden kasvaessa mallipohjaiset menetelmät ovat keskeisiä tuotantojärjestelmien hallinnassa ja kehityksessä [14]. Digitaalisten työkalujen, kuten simuloinnin ja digitaalisten kaksosten, merkitys kasvaa tulevaisuudessa tuotantojärjestelmien kehityksessä [3]. Kokonaisuudessaan teollisuusautomaation kehitys kohti verkottuneita ja älykkäitä järjestelmiä vahvistaa mallipohjaisen suunnittelun asemaa keskeisenä kehitysmenetelmiä [3, 12].

3.4 Lääketekniikka

Lääketekniikassa järjestelmien suunnittelua ohjaa ensisijaisesti potilasturvallisuus, sillä järjestelmien virheet voivat vaikuttaa suoraan potilaan terveyteen [5, 18]. Lääketieteellisten laitteiden kehitystä ohjaavat tiukat sääntelyvaatimukset ja hyväksyntäprosessit, jotka edellyttävät kattavaa dokumentointia ja järjestelmällistä suunnittelua [9, 19]. Lääketekniset järjestelmät yhdistävät laitteiston, ohjelmiston ja sensorteknologian, mikä tekee niiden kehityksestä monialaista ja kompleksista [12, 19]. Monien lääketieteellisten järjestelmien tulee toimia reaaliaikaisesti ja erittäin tarkasti, mikä lisää suunnittelun vaativuutta [21]. Lääketieteellisten laitteiden kompleksisuus syntyy erityisesti järjestelmien välisestä vuorovaikutuksesta sekä suurten datamäärien käsittelystä [22]. Lääketieteellisillä laitteilla on usein pitkä elinkaari, mikä edellyttää järjestelmien ylläpidettävyyttä ja päivitettävyyttä koko käyttöiän ajan [6]. Terveystieteiden digitaalisten järjestelmien ja data merkitys on kasvanut, mikä lisää järjestelmien suunnittelun ja hallinnan vaativuutta [18, 20]. Näiden vaatimusten vuoksi mallipohjainen järjestelmäsuunnittelu tarjoaa tehokkaan lähestymistavan

lääketeknisten järjestelmien kompleksisuuden, turvallisuuden ja jäljitettävyyden hallintaan [5, 9, 19].

Lääketekniset järjestelmät ovat monimutkaisia kokonaisuuksia, jotka yhdistävät laitteiston, ohjelmiston ja sensorijärjestelmät yhdeksi toimivaksi kokonaisuudeksi [12, 19]. Järjestelmien kompleksisuus syntyy erityisesti eri alijärjestelmien välisestä vuorovaikutuksesta ja niiden keskinäisistä riippuvuuksista [14]. Lääketekniset järjestelmät käsittelevät suuria määriä potilasdataa, mikä lisää järjestelmien suunnittelun ja hallinnan vaativuutta [21]. Monien lääketieteellisten laitteiden on toimittava reaaliaikaisesti, mikä lisää järjestelmien suunnittelun kompleksisuutta ja luotettavuusvaatimuksia [21]. Lääketekniset järjestelmät ovat usein osa laajempaa terveydenhuollon infrastruktuuria, mikä edellyttää niiden yhteensopivuutta muiden järjestelmien kanssa [22]. Järjestelmien virheillä voi olla vakavia seurauksia potilasturvallisuudelle, mikä korostaa kompleksisuuden hallinnan merkitystä [21]. Kompleksisuuden vuoksi lääketeknisiä järjestelmiä on tarkasteltava kokonaisuutena, jossa huomioidaan kaikki alijärjestelmät ja niiden väliset suhteet [2]. Eri järjestelmien yhteensopivuus ja standardien noudattaminen muodostavat keskeisiä haasteita lääketeknisten järjestelmien kehityksessä [23]. Teknologian kehittyessä lääketeknisten järjestelmien kompleksisuus kasvaa entisestään, mikä lisää tarvetta systemaattisille suunnittelumenetelmille [14]. Mallipohjainen järjestelmäsuunnittelu tarjoaa keinon hallita lääketeknisten järjestelmän kompleksisuutta mallintamalla järjestelmien rakenteet ja riippuvuudet [9, 19].

Lääketekniikassa mallipohjainen järjestelmäsuunnittelu toimii keskeisenä lähestymistapana monimutkaisten ja turvallisuuskriittisten järjestelmien kehityksessä [9, 19]. Mallipohjainen järjestelmäsuunnittelu mahdollistaa lääketieteellisten järjestelmien arkkitehtuurin kuvaamisen yhtenäisenä mallina, jossa eri alijärjestelmien suhteet voidaan esittää selkeästi [6]. Mallipohjainen lähestymistapa tukee vaatimusten hallintaa yhdistämällä vaatimukset suoraan järjestelmämalleihin [2]. Lääketekniikassa mallipohjainen järjestelmäsuunnittelu parantaa jäljitettävyyttä yhdistämällä vaatimukset, suunnitteluratkaisut ja testauksen tulokset yhdeksi kokonaisuudeksi [5, 9]. Mallipohjainen järjestelmäsuunnittelu tukee riskienhallintaa mahdollistamalla järjestelmien toiminnan analysoinnin ja potentiaalisten virhetilanteiden tunnistamisen jo suunnitteluvaiheessa [18, 19]. Mallipohjainen suunnittelu helpottaa lääketieteellisten järjestelmien integraatioita tarjoamalla yhteisten mallien eri alijärjestelmien

yhteensovittamiselle [22]. Mallipohjainen järjestelmäsuunnittelu mahdollistaa simulointimallien hyödyntämisen järjestelmien analysoinnissa ja validoinnissa ennen niiden käyttöönottoa [21]. Mallipohjainen lähestymistapa tukee kattavaa dokumentointia, joka on keskeistä lääketeknisten järjestelmien hyväksyntäprosesseissa [6]. Mallipohjainen järjestelmäsuunnittelu yhdistää eri suunnittelualat, kuten ohjelmiston, laitteiston ja datankäsittelyn, yhdeksi hallittavaksi kokonaisuudeksi [14]. Lääketeknisten järjestelmien kompleksisuuden kasvaessa mallipohjaisen järjestelmäsuunnittelun merkitys korostuu keskeisenä työkaluna järjestelmien turvallisessa ja tehokkaassa kehityksessä [3, 19].

Lääketekniikassa simulointi on keskeinen työkalu järjestelmien kehityksessä, sillä sen avulla voidaan arvioida laitteiden toimintaa ennen niiden käyttöönottoa [21]. Simuloinnin avulla voidaan varmistaa lääketieteellisten järjestelmien turvallisuus analysoimalla niiden toimintaa erilaisissa käyttötilanteissa [11]. Lääketieteelliset laitteet testataan perusteellisesti simulointien avulla ennen niiden käyttöönottoa kliinisessä ympäristössä [21]. Simulointi mahdollistaa erilaisten virhetilanteiden analysoinnin ilman riskiä potilasturvallisuudelle [11]. Simulointimallit mahdollistavat lääketieteellisten järjestelmien tarkastelun kokonaisuutena, jossa kaikki alijärjestelmät ja niiden vuorovaikutukset huomioidaan [2]. Reaaliaikaisten lääketieteellisten järjestelmien testaus simulointien avulla auttaa varmistamaan niiden luotettavan toiminnan kriittisissä tilanteissa [21]. Mallipohjainen järjestelmäsuunnittelu tukee simulointia tarjoamalla tarkkoja järjestelmämalleja, joita voidaan hyödyntää testauksessa ja analyysissä [6]. Simulointi on keskeinen osa lääketeknisten järjestelmien verifiointi- ja validointiprosesseja [2]. Simulointiin perustuva testaus vähentää kehitykseen liittyviä riskejä, koska mahdolliset virheet voidaan tunnistaa ennen käyttöönottoa [14]. Digitaalisten työkalujen, kuten simuloinnin ja mallinnuksen merkitys kasvaa lääketeknisten järjestelmien kehityksessä jatkuvasti [18, 20].

Lääketekniikassa turvallisuus on keskeinen järjestelmäsuunnittelua ohjaava tekijä, sillä järjestelmän virheet voivat vaikuttaa suoraan potilaiden terveyteen [21]. Lääketieteellisten laitteiden kehitystä ohjaa tiukka sääntely, joka edellyttää järjestelmien vaatimustenmukaisuuden osoittamista ennen käyttöönottoa [11]. Turvallisuusvaatimuksia ohjaavat erilaiset standardit, jotka määrittelevät vaatimuksia järjestelmien suunnittelulle, testaukselle ja validoinnille [2]. Jäljitettävyyden on keskeinen osa lääketeknisten järjestelmien

sertifiointia, sillä sen avulla voidaan yhdistää vaatimukset, suunnittelu, ja testaus toisiinsa[6]. Mallipohjainen järjestelmäsuunnittelu tukee turvallisuutta mahdollistamalla järjestelmien rakenteen ja toiminnan systemaattisen analysoinnin [5, 9]. Mallipohjainen lähestymistapa tukee riskienhallintaa tunnistamalla mahdolliset virhetilanteet ja niiden vaikutukset jo suunnitteluvaiheessa [11]. Lääketekniikassa järjestelmien verifiointi ja validointi ovat keskeisiä vaiheita, joissa varmistetaan järjestelmien turvallisuus ja toimivuus [2]. Sääntely edellyttää kattavaa dokumentointia, jonka avulla voidaan osoittaa järjestelmien vaatimustenmukaisuus [6]. Järjestelmien kompleksisuuden kasvu lisää turvallisuushaasteita, mikä korostaa systemaattisten suunnittelumenetelmien merkitystä [11]. Uusien teknologioiden käyttöönotto lisää turvallisuusvaatimuksia, mikä vahvistaa mallipohjaisen suunnittelun roolia lääketekniikassa [21].

Lääketekniikka kehittyy nopeasti uusien teknologioiden, kuten tekoälyn ja kehittyneen data-analytiikan myötä [21, 22]. Uusien teknologioiden käyttöönotto lisää lääketeknisten järjestelmien kompleksisuutta, sillä ne tuovat mukanaan uusia alijärjestelmiä ja riippuvuuksia [22]. Lääketieteelliset laitteet ovat yhä enemmän verkottuneita ja älykkäitä, mikä mahdollistaa reaaliaikaisen datan hyödyntämisen potilaan hoidossa [22]. Reaaliaikaisen datan hyödyntäminen parantaa diagnostiikkaa ja hoidon seuranta, mutta lisää samalla järjestelmien suunnittelun vaativuutta [21]. Järjestelmien kehittyessä myös turvallisuusvaatimukset kasvavat, mikä korostaa systemaattisten suunnittelumenetelmien merkitystä [11]. Mallipohjaisen järjestelmäsuunnittelun merkitys kasvaa tulevaisuudessa, koska se mahdollistaa monimutkaisten järjestelmien hallinnan ja analysoinnin[6]. Uusien teknologioiden myötä integraatiohaasteet lisääntyvät, sillä järjestelmien tulee toimia saumattomasti yhdessä muiden terveydenhuollon järjestelmien kanssa [22]. Järjestelmien elinkaaren hallinta korostuu tulevaisuudessa, kun lääketieteellisiä laitteita päivitetään ja kehitetään jatkuvasti[2]. Digitaalisten työkalujen, kuten simuloinnin ja mallinnuksen merkitys kasvaa lääketeknisten järjestelmien kehityksessä [21]. Kokonaisuudessaan lääketekniikan kehitys kohti älykkäämpiä ja verkottuneempia järjestelmiä vahvistaa mallipohjaisen suunnittelun asemaa keskeisenä kehitysmenetelmänä [11], [22].

3.5 Muut alat

Mallipohjaista järjestelmäsuunnittelua hyödynnetään laajasti eri toimialoilla, eikä sen käyttö rajoitu tiettyihin teollisuudenaloihin [2]. Mallipohjainen järjestelmäsuunnittelu soveltuu erityisesti monimutkaisten ja moniulotteisten järjestelmien suunnitteluun ja hallintaan [14]. Eri toimialoja yhdistää järjestelmien kasvava kompleksisuus, mikä lisää tarvetta systemaattisille suunnittelumenetelmille [14]. Mallipohjainen järjestelmäsuunnittelu mahdollistaa järjestelmätason tarkastelun, jossa huomioidaan järjestelmän kaikki osat ja niiden väliset riippuvuudet [6]. Mallipohjainen lähestymistapa tarjoaa yhtenäisen tavan hallita järjestelmiä eri toimialoilla niiden erityispiirteistä huolimatta [2]. Digitalisaatio ja järjestelmien ohjelmistointensiivisyys ovat lisänneet mallipohjaisen järjestelmäsuunnittelun merkitystä useilla toimialoilla [14]. Mallipohjaisen järjestelmäsuunnittelun käyttö on laajentunut perinteisistä insinöörialoista myös uusille toimialoille teknologian kehittyessä [5, 8]. Kokonaisuudessaan mallipohjainen järjestelmäsuunnittelu tarjoaa tehokkaan lähestymistavan monimutkaisten järjestelmien hallintaan eri toimialoilla, ja sen merkitys kasvaa jatkuvasti [2], [14].

Rakennus- ja infrastruktuurijärjestelmät ovat komplekseja kokonaisuuksia, joissa useat sidosryhmät vaikuttavat toisiinsa ja järjestelmän toimintaan. Sidosryhmien vaatimusten määrittely on keskeinen tekijä onnistuneessa järjestelmäkehityksessä rakennus- ja infrastruktuuriprojekteissa. Mallipohjainen järjestelmäsuunnittelu laajentaa perinteistä dokumenttipohjaista suunnittelua tarjoamalla visuaalisia ja rakenteellisia malleja järjestelmän hallintaan. Mallipohjainen lähestymistapa mahdollistaa sidosryhmien vaatimusten systemaattisten mallintamisen osana järjestelmämallia. Mallipohjaisen järjestelmäsuunnittelun avulla voidaan rakentaa sidosryhmien arvoverkkomalleja, joissa kuvataan vaatimusten ja sidosryhmien väliset suhteet. Mallipohjaiset verkkomallit mahdollistavat järjestelmän rakenteen analysoinnin ja sidosryhmien priorisoinnin. Mallipohjainen järjestelmäsuunnittelu parantaa vaatimusten jäljitettävyyttä yhdistämällä ne suoraan järjestelmämalliin. Mallipohjainen lähestymistapa tukee järjestelmien hallintaa koko elinkaaren ajan suunnittelusta käyttöön ja ylläpitoon. Mallipohjainen järjestelmäsuunnittelu mahdollistaa mallien modulaarisuuden ja uudelleenkäytettävyyden, mikä tehostaa monimutkaisten projektien suunnittelua. Kokonaisuudessaan mallipohjainen

järjestelmäsuunnittelu tarjoaa tehokkaan lähestymistavan rakennus- ja infrastruktuurialan monimutkaisten järjestelmien ja sidosryhmien hallintaan. [24]

Kemianteollisuudessa keskeisessä roolissa ovat monimutkaiset tuotantoprosessit, joiden suunnittelu ja hallinta vaativat systemaattista lähestymistapaa [10]. Mallipohjaisia menetelmiä hyödynnetään kokeiden suunnittelussa, mikä mahdollistaa prosessien tehokkaamman analysoinnin ja optimoinnin. Mallipohjainen suunnittelu tukee kemiallisten prosessien optimointia parantamalla tuotannon tehokkuutta ja laatua. Kemianteollisuuden järjestelmät ovat usein dynaamisia ja monimutkaisia, mikä lisää niiden mallintamisen ja analysoinnin tarvetta. [10] Mallipohjainen järjestelmäsuunnittelu tarjoaa rakenteellisen tavan mallintaa kemiallisia prosesseja ja niiden välisiä riippuvuuksia [5, 8] Mallien avulla voidaan ennakoita prosessien käyttäytymistä ja tunnistaa mahdollisia ongelmia ennen niiden ilmenemistä tuotannossa. Mallipohjainen lähestymistapa parantaa prosessien tehokkuutta ja laatua mahdollistamalla tarkemman analyysin ja optimoinnin. Mallipohjaisia menetelmiä hyödynnetään kemianteollisuudessa erityisesti prosessien analysoinnissa ja optimoinnissa. [10]

Tuotesuunnittelussa tuotteita tarkastellaan kokonaisvaltaisesti järjestelminä, joissa eri komponenttien väliset suhteet ja riippuvuudet ovat keskeisiä [5, 9]. Mallipohjainen järjestelmäsuunnittelu tarjoaa rakenteellisen lähestymistavan tuotekehitykseen mahdollistamalla järjestelmien mallintamisen ja analysoinnin yhtenäisen mallin avulla [6]. Mallipohjainen järjestelmäsuunnittelu tukee modulaarista tuotesuunnittelua, jossa tuotteet voidaan rakentaa standardoiduista ja uudelleenkäytettävistä komponenteista [10]. Mallipohjainen lähestymistapa mahdollistaa järjestelmien osien uudelleen käytön eri tuotesukupolvissa, mikä tehostaa tuotekehitysprosessia [9]. Mallipohjainen järjestelmäsuunnittelu tukee tuotejärjestelmien arkkitehtuurin kehittämistä kuvaamalla komponenttien väliset suhteet ja rakenteet selkeästi [6, 9]. tuotesuunnittelussa keskeistä on semanttinen yhteensopivuus, joka mahdollistaa tuotetiedon yhdenmukaisen tulkinnan eri järjestelmien välillä [23]. Mallipohjainen järjestelmäsuunnittelu tukee tuotesuunnittelun tiedonhallintaa yhdistämällä eri suunnitteluvaiheiden tiedot yhdeksi yhtenäiseksi kokonaisuudeksi [9, 21]. Mallipohjainen lähestymistapa mahdollistaa tiedon integraation eri suunnittelutyökalujen välillä, mikä parantaa suunnitteluprosessien tehokkuutta [23].

Mallipohjainen järjestelmäsuunnittelu mahdollistaa tuotetiedon hyödyntämisen koko tuotteen elinkaaren ajan suunnittelusta ylläpitoon [5, 21]. Mallipohjainen järjestelmäsuunnittelu yhdistää tuotesuunnittelussa järjestelmien rakenteellisen kehittämisen sekä tiedonhallinnan ja toimivuuden, mikä parantaa suunnittelun tehokkuutta ja laatua [9, 21, 24].

4 Johtopäätökset

Teknisten järjestelmien kompleksisuus on kasvanut merkittävästi digitalisaation, ohjelmistointensiivisyyden ja järjestelmä integraation lisääntymisen myötä [1, 2]. Modernit järjestelmät muodostuvat useista toisistaan riippuvaisista alijärjestelmistä, mikä lisää järjestelmien suunnittelun ja hallinnan vaativuutta [2]. Erityisesti autoteollisuudessa järjestelmien ohjelmistopainotteisuus ja turvallisuusvaatimukset ovat lisänneet tarvetta tehokkaille suunnittelumenetelmille [3]. Perinteinen dokumenttipohjainen suunnittelu ei aina riitä monimutkaisten järjestelmien hallintaan, sillä tiedon määrä ja järjestelmien väliset riippuvuudet ovat kasvaneet merkittävästi [1, 6]. Mallipohjaiset suunnittelumenetelmät ovat nousseet keskeiseen rooliin modernien järjestelmien kehityksessä, koska ne mahdollistavat järjestelmien tarkastelun yhtenäisen mallien avulla. Mallipohjaisen suunnittelu ja mallipohjaisen järjestelmäsuunnittelu tarjoavat tehokkaita menetelmiä järjestelmien suunnitteluun, simulointiin ja analysointiin [1, 7]. Mallipohjaiset menetelmät tukevat järjestelmien jäljitettävyyttä, simulointia ja muutosten hallintaa koko kehitysprosessien ajan [2, 6].

Työssä havaittiin, että mallipohjainen suunnittelu ja mallipohjainen järjestelmäsuunnittelu perustuvat samaan mallipohjaiseen ajatteluun, mutta niiden käyttötarkoitukset eroavat toisistaan [1, 7]. mallipohjainen järjestelmäsuunnittelu keskittyy järjestelmän vaatimusten, rakenteen ja arkkitehtuurin määrittelyyn järjestelmätasolla [2, 6]. mallipohjainen suunnittelu puolestaan painottuu järjestelmän simulointiin toteutukseen ja testaukseen [7, 8]. Mallipohjainen järjestelmäsuunnittelu mahdollistaa järjestelmien tarkastelun kokonaisuutena, jossa huomioidaan järjestelmän eri osat ja niiden väliset riippuvuudet [1, 2]. Mallipohjainen suunnittelu hyödyntää simulointia järjestelmän toiminnan analysoinnissa ja virheiden tunnistamisessa jo kehityksen alkuvaiheessa [8]. Vaikka mallipohjainen suunnittelu ja mallipohjainen järjestelmäsuunnittelu esitetään usein erillisinä lähestymistapoina, ne täydentävät toisiaan osana yhtenäistä kehitysprosessia [1, 2]. Mallipohjaisen järjestelmäsuunnittelun tuottamat vaatimukset ja arkkitehtuurimallit ohjaavat mallipohjaisen suunnittelun simulointi ja toteutus [2, 6]. Simuloinnin ja testauksen tulokset voivat puolestaan tarjota palautetta järjestelmätason suunnittelun ja vaatimusten kehittämiseen [2, 8]. Yhdessä

mallipohjainen suunnittelu ja mallipohjainen järjestelmäsuunnittelu muodostavat yhtenäisen mallipohjaisen kehitysprosessin, joka tukee monimutkaisten järjestelmien hallintaa [1, 2].

Työssä tarkasteltiin mallipohjaisen järjestelmäsuunnittelun ja mallipohjaisen suunnittelun soveltamista useilla eri toimialoilla [2]. Mallipohjaisia menetelmiä hyödynnetään erityisesti toimialoilla, joissa järjestelmien kompleksisuus, turvallisuusvaatimukset ja integraatiohaasteet ovat merkittävät [3, 11]. Autoteollisuudessa mallipohjaiset menetelmät tukevat ohjelmistointensiivisten järjestelmien suunnittelua, simulointia ja testausta [9]. Lääketekniikassa mallipohjainen järjestelmäsuunnittelu parantaa järjestelmien jäljitettävyyttä, turvallisuutta ja vaatimusten hallintaa [2, 11]. Rakennus- ja infrastruktuurialalla mallipohjainen järjestelmäsuunnittelu tukee sidosryhmien vaatimusten hallintaa sekä monimutkaisten projektien koordinoitua [9]. Kemianteollisuudessa mallipohjaiset menetelmät hyödynnetään erityisesti prosessien analysoinnissa ja optimoinnissa [10]. Tuotesuunnittelussa mallipohjainen järjestelmäsuunnittelu mahdollistaa tuotetiedon hallinnan sekä komponenttien välisten riippuvuuksien tarkastelun yhtenäisten mallien avulla [18, 23]. Mallipohjaiset menetelmät tukevat järjestelmien simulointia ja analysointia ja muutosten hallintaa koko järjestelmän elinkaaren ajan. Eri toimialoja yhdistää tarve hallita monimutkaisia ja moniulotteisia järjestelmiä systemaattisesti ja tehokkaasti. Tulosten perusteella mallipohjainen järjestelmäsuunnittelu tarjoaa tehokkaan lähestymistavan monimutkaisten järjestelmien hallintaan eri toimialoilla. [2, 3]

Teknologian kehittyessä mallipohjaisten suunnittelumenetelmien merkitys kasvaa jatkuvasti eri teollisuudenaloilla [2]. Digitalisaatio, verkottuneet järjestelmät ja ohjelmistointensiivisyys lisäävät tarvetta tehokkaille järjestelmäsuunnittelumenetelmille [3, 20]. Teollisuus-4.0 kehitys on lisännyt reaaliaikaisen datan, automaation ja älykkäiden järjestelmien hyödyntämistä modernissa teollisuudessa [4]. Tulevaisuudessa mallipohjaisia menetelmiä hyödynnetään yhä enemmän digitaalisten kaksosten, simuloinnin ja reaaliaikaisen analysoinnin tukena [4, 22]. Tekoälyn ja data-analytiikan hyödyntäminen lisää järjestelmien kompleksisuutta, mikä korostaa systemaattisten suunnittelumenetelmien merkitystä [22]. Verkottuneiden järjestelmien lisääntyminen kasvattaa integraatiohaasteita, mikä vahvistaa mallipohjaisen järjestelmäsuunnittelun roolia järjestelmän hallinnassa [2, 22]. Mallipohjainen järjestelmäsuunnittelu mahdollistaa järjestelmien rakenteen, vaatimusten ja riippuvuuksien

tehokkaan hallinnan myös tulevaisuuden monimutkaisissa järjestelmissä [1, 2]. Tulevaisuudessa mallipohjaiset menetelmät voivat mahdollistaa entistä tehokkaamman yhteistyön eri suunnittelualueiden välillä yhteisten järjestelmämallien avulla [6]. Kokonaisuudessaan mallipohjaisen järjestelmäsuunnittelun merkitys arvioidaan kasvavan tulevaisuudessa järjestelmien kompleksisuuden lisääntyessä [2].

5 Yhteenveto

Tutkimuskysymys 1:

Miten mallipohjainen suunnittelu ja mallipohjainen järjestelmäsuunnittelu eroavat toisistaan?

Työssä havaittiin, että mallipohjainen suunnittelu ja mallipohjainen järjestelmäsuunnittelu perustuvat samaan mallipohjaiseen ajatteluun, mutta niiden käyttötarkoitukset eroavat toisistaan. Mallipohjainen järjestelmäsuunnittelu keskittyy järjestelmän vaatimusten, arkkitehtuurin ja rakenteen määrittelyyn järjestelmätasolla, kun taas mallipohjainen suunnittelu painottuu järjestelmän simulointiin, toteutukseen ja testaukseen. Mallipohjainen järjestelmäsuunnittelu mahdollistaa järjestelmien tarkastelun kokonaisuutena, kun taas mallipohjainen suunnittelu keskittyy järjestelmien toiminnallisuuden analysointiin ja simulointiin. Menetelmät eivät kuitenkaan ole toisistaan erillisiä, vaan ne täydentävät toisiaan osana yhtenäistä kehitysprosessia. [1, 2, 7]

Tutkimuskysymys 2:

Miten mallipohjaisia menetelmiä sovelletaan eri teollisuudenaloilla?

Työssä havaittiin, että mallipohjaisia menetelmiä hyödynnetään laajasti eri toimialoilla erityisesti monimutkaisten ja ohjelmistointensiivisten järjestelmien kehityksessä. Autoteollisuudessa mallipohjaisia menetelmiä käytetään simulointiin, järjestelmien integraatioon ja turvallisuuskriittisten järjestelmien kehitykseen. Lääketekniikassa mallipohjainen järjestelmäsuunnittelu tukee turvallisuutta, jäljitettävyyttä ja järjestelmien hallintaa. Rakennus- ja infrastruktuurialalla mallipohjaisia menetelmiä hyödynnetään sidosryhmien vaatimusten hallinnassa ja monimutkaisten projektien koordinoinnissa, kun taas kemianteollisuudessa niitä käytetään erityisesti prosessien analysoinnissa ja optimoinnissa. Tulosten perusteella mallipohjaiset menetelmät tarjoavat tehokkaan lähestymistavan monimutkaisten järjestelmien suunnitteluun, analysointiin ja hallintaan useilla eri toimialoilla. [2, 3, 9, 10, 11]

Lähteet

- [1] J. A. Estefan, "Survey of Model-Based Systems Engineering (MBSE) Methodologies", 2008.
- [2] International Council on Systems Engineering, *Systems engineering handbook: a guide for system life cycle processes and activities*, Fourth Edition. Hoboken, N.J: Wiley, 2015.
- [3] T. Kolossváry, D. Feszty, ja T. Dóry, "Systems engineering in automotive product development: A guide to initiate organisational transformation", *J. Open Innov. Technol. Mark. Complex.*, vol. 9, nro 4, s. 100160, joulu 2023, doi: 10.1016/j.joitmc.2023.100160.
- [4] M. N., S. Srinivasan, K. Ramkumar, D. Pal, J. Vain, ja S. Ramaswamy, "A model-based approach for design and verification of Industrial Internet of Things", *Future Gener. Comput. Syst.*, vol. 95, s. 354–363, kesä 2019, doi: 10.1016/j.future.2018.12.012.
- [5] T. Weillkiens, *Systems engineering with SysML/UML: modeling, analysis, design*. Amsterdam Boston: Morgan Kaufmann OMG Press/Elsevier, 2007.
- [6] S. Friedenthal, *A practical guide to SysML: the systems modeling language*, 2nd ed. Waltham, MA: Morgan Kaufmann, 2012.
- [7] B. Selic, "The pragmatics of model-driven development", *IEEE Softw.*, vol. 20, nro 5, s. 19–25, syys 2003, doi: 10.1109/MS.2003.1231146.
- [8] J. Nibert, M. Herniter, ja Z. Chambers, "Model-Based System Design for MIL, SIL, and HIL", *World Electr. Veh. J.*, vol. 5, nro 4, s. 1121–1130, joulu 2012, doi: 10.3390/wevj5041121.
- [9] A. Albers, H. Scherer, N. Bursac, ja G. Rachenkova, "Model Based Systems Engineering in Construction Kit Development – Two Case Studies", *Procedia CIRP*, vol. 36, s. 129–134, 2015, doi: 10.1016/j.procir.2015.01.044.
- [10] M. Geremia, S. Macchietto, ja F. Bezzo, "A review on model-based design of experiments for parameter precision – Open challenges, trends and future perspectives", *Chem. Eng. Sci.*, vol. 319, s. 122347, tammi 2026, doi: 10.1016/j.ces.2025.122347.
- [11] N. U. I. Hossain, M. Lutfi, I. Ahmed, ja H. Debusk, "Application of systems modeling language (SysML) and discrete event simulation to address patient waiting time issues in healthcare", *Smart Health*, vol. 29, s. 100403, syys 2023, doi: 10.1016/j.smhl.2023.100403.
- [12] K. Goseva-Popstojanova, "Report: Survey on Model-Based Software Engineering and Auto-Generated Code", 2016.
- [13] Y. Sirgabsou, C. Baron, L. Pahun, ja P. Esteban, "Model-driven engineering to ensure automotive embedded software safety. Methodological proposal and case study", *Comput. Ind.*, vol. 138, s. 103636, kesä 2022, doi: 10.1016/j.compind.2022.103636.
- [14] S. Muthukumar, N. U. I. Hossain, ja S. L. Fong, "Mapping the landscape of model-based systems engineering (MBSE) through bibliometric and network analysis", *Digit. Eng.*, vol. 9, s. 100088, touko 2026, doi: 10.1016/j.dte.2026.100088.
- [15] B. Cole, V. Mittal, S. Gillespie, N. La, R. Wise, ja A. MacCalman, "Model-based systems engineering: application and lessons from a technology maturation project", *Procedia Comput. Sci.*, vol. 153, s. 202–209, 2019, doi: 10.1016/j.procs.2019.05.071.
- [16] A. Akundi ja V. Lopez, "A Review on Application of Model Based Systems Engineering to Manufacturing and Production Engineering Systems", *Procedia Comput. Sci.*, vol. 185, s. 101–108, 2021, doi: 10.1016/j.procs.2021.05.011.
- [17] N. Kuelper, A. K. Jeyaraj, S. Liscouët-Hanke, ja F. Thielecke, "Integration of a model-based systems engineering framework with safety assessment for early design phases: A case study for hydrogen-based aircraft fuel system architecting", *Results Eng.*, vol. 25, s. 104249, maaliskuu 2025, doi: 10.1016/j.rineng.2025.104249.

- [18] H. Scherer, A. Albers, ja N. Bursac, "Model Based Requirements Engineering for the Development of Modular Kits", *Procedia CIRP*, vol. 60, s. 145–150, 2017, doi: 10.1016/j.procir.2017.01.032.
- [19] S. Perry ja Holt, John, *SysML for Systems Engineering. IET Professional Applications of Computing Series, Volume 7*. The Institute of Engineering and Technology, 2008.
- [20] M. N., S. Srinivasan, K. Ramkumar, D. Pal, J. Vain, ja S. Ramaswamy, "A model-based approach for design and verification of Industrial Internet of Things", *Future Gener. Comput. Syst.*, vol. 95, s. 354–363, kesä 2019, doi: 10.1016/j.future.2018.12.012.
- [21] T. Golgolnia, T. Kipouros, P. John Clarkson, M. Kevdzija, ja G. Marquardt, "A computational model for linking healthcare architectural variables with health outcomes using systems engineering", *Build. Environ.*, vol. 287, s. 113763, tammi 2026, doi: 10.1016/j.buildenv.2025.113763.
- [22] S. Maleki, N. Jazdi, ja B. Ashtari, "Intelligent Digital Twin in Health Sector: Realization of a Software-Service for Requirements- and Model-based-Systems-Engineering", *IFAC-Pap.*, vol. 55, nro 19, s. 79–84, 2022, doi: 10.1016/j.ifacol.2022.09.187.
- [23] A. Sarkar ja D. Šormaz, "On Semantic Interoperability of Model-based Definition of Product Design", *Procedia Manuf.*, vol. 38, s. 513–523, 2019, doi: 10.1016/j.promfg.2020.01.065.
- [24] Y. Feng, G. Gao, P. Wang, M. Fu, ja Z. Zhang, "Towards model-based requirements engineering: Construction method of stakeholder value networks model based on MBSE", *Comput. Ind. Eng.*, vol. 195, s. 110398, syys 2024, doi: 10.1016/j.cie.2024.110398.