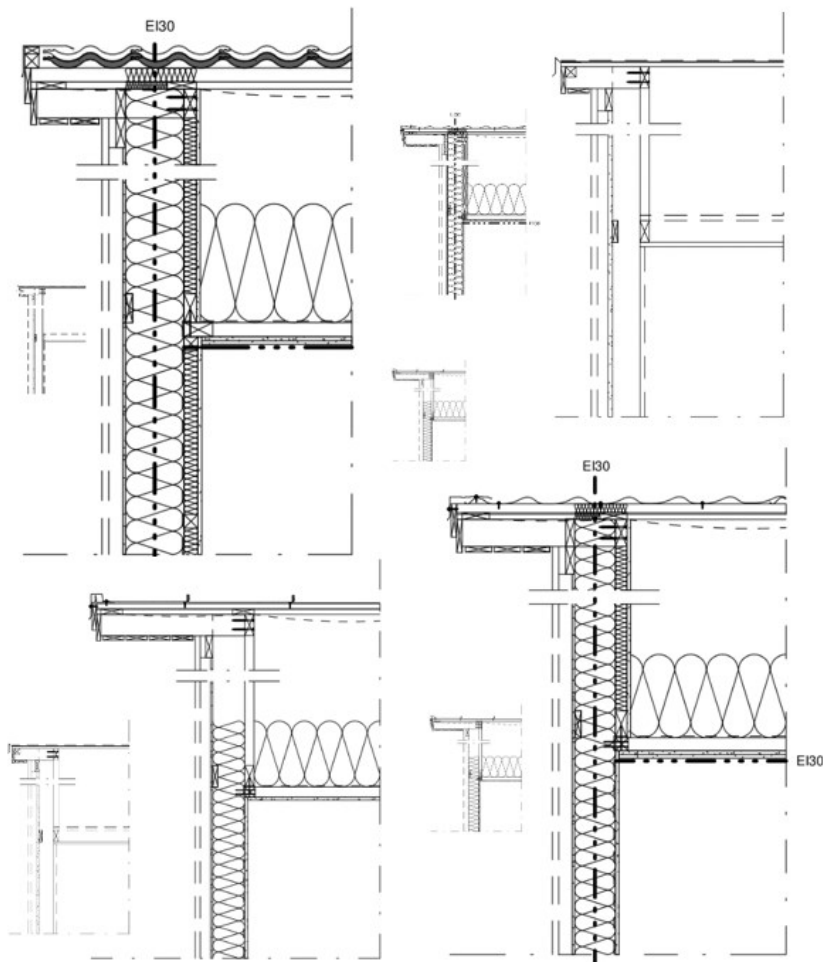


Heidi Maliniemi

Parametrisen detaljin luominen

Autodesk Revitin Family Editorilla



Insinööri (AMK)
Rakennus-
ja yhdyskuntatekniikka
Syksy 2024



KAMK • University
of Applied Sciences

Tiivistelmä

Tekijä(t): Maliniemi Heidi

Työn nimi: Parametrisen detaljin luominen Autodesk Revitin Family Editorilla

Tutkintonimike: Insinööri (AMK), rakennus- ja yhdyskuntatekniikka

Asiasanat: Autodesk Revit, detaljipiirustus, family, Family Editor, parametrit, rakennesuunnittelu

Tämän opinnäytetyön tarkoituksena oli luoda parametrinen detalji Autodesk Revit -ohjelmalla, selvittää detaljin kehitysvaiheet sekä ohjelman asettamat haasteet ja mahdollisuudet. Opinnäytetyö tehtiin toimeksiantona Insinööritoimisto Raksainssit Oy:lle. Yrityksen tarkoituksena on kehittää rakennesuunnittelun detaljointia rakentamalla Revit-pohjainen detaljikirjasto. Detaljien luomiseen käytettäväksi menetelmäksi on kaavailtu parametristä detaljia, jossa yhdestä toiminnallisesta detaljista saadaan tuotettua useita eri variaatioita muuttamalla toiminnallisuutta ohjaavan valikon parametrejä. Tässä työssä käytiin läpi parametrisen detaljin kehitysprosessi sen toteutuskelpoisuuden arvioimiseksi.

Työssä tarkasteltiin lainsäädäntöä ja ohjeistuksia, jotka säätelevät detaljipiirrosten sisältöä ja esitystapaa. Detaljeissa esitettävät rakenneratkaisut tulee toteuttaa määräysten mukaisesti, jotta rakennus täyttää hyvän rakennustavan vaatimukset. Määräykset kattavat useita tärkeitä alueita, kuten rakenteiden lujuuden ja vakauden, paloturvallisuuden, terveellisyyden, käyttöturvallisuuden sekä energiatehokkuuden. Lisäksi työssä perehdyttiin yleisellä tasolla kehitysprosessissa käytettävän Revit-ohjelman rakenteeseen, ominaisuuksiin ja parametrisuuteen.

Opinnäytetyö toteutettiin toiminnallisena kehittämistyönä, jossa rakennettiin parametrinen detalji Revit-ohjelman Family Editorilla. Prosessin kuvauksessa käytiin läpi parametrisen päätyräystäsdetaljin kehittämisen vaiheet. Detalji toteutettiin 2D-familynä, johon lisättiin parametreillä muokattavaa toiminnallisuutta rakennekerrosten dimensioihin, rakennusmateriaaleihin ja rakenneratkaisuihin.

Opinnäytetyön tuloksena saatiin aikaan parametrinen detalji, joka on muokattavissa Revitin projektitilassa. Detaljin toiminnallisuus mahdollistaa sen, että yksi detalji voi korvata useita kymmeniä detaljikirjaston perinteisiä detaljeja. Kehitysprosessin aikana selvitettiin yksityiskohtaisesti prosessin vaiheet, haasteet ja mahdollisuudet, mikä tarjoaa arvokasta tietoa jatkokehitystä varten. Tulosten perusteella toimeksiantajayritys voi jatkaa parametristen detaljien ja Revit-pohjaisen detaljikirjaston kehittämistä.

Abstract

Author(s): Maliniemi Heidi

Title of the Publication: Creating a Parametric Detail with Revit Family Editor

Degree Title: Bachelor of Engineering, Construction Engineering

Keywords: Autodesk Revit, detail drawing, Family Editor, parameter, structural design

The aim of this thesis was to create a parametric detail using Autodesk Revit, to investigate the development phases of the parametric detail as well as the challenges and opportunities presented by the software. This thesis was commissioned by Insinööritoimisto Raksainssit Oy. Company's goal is to enhance structural design detailing by developing a Revit-based detail library. The method to produce multiple variations from a single functional detail was to change the parameters in the control menu. In this work the development process of the parametric detail is reviewed to assess its feasibility.

The thesis examines the legislation and guidelines that regulate the content and presentation of detail drawings. The structural solutions must comply with regulations to ensure that the building meets the standards of good construction practice. The regulations cover several critical areas, including structural strength and stability, fire safety, health, usability, and energy efficiency. In addition, the thesis provides an overview of the Revit software's structure, features, and parametric capabilities.

The thesis was carried out as a functional development project, with a parametric detail created using Revit's Family Editor. The phases of developing the parametric detail were described in the process. A 2D family was implemented with adjustable parameters affecting structural layer dimensions, building materials, and structural solutions.

The result of the thesis is a parametric detail that can be modified in Revit's project environment. The detail functionality allows one detail to replace several dozen traditional details in the detail library. The development process presents the phases, challenges, and opportunities, providing valuable insights for further development. As a consequence the commissioning company is able to develop the parametric details and the Revit-based detail library.

Sisällys

1	Johdanto	1
2	Yrityksen esittely ja työn tavoitteet.....	3
2.1	AutoCAD-detaljikirjasto.....	3
3	Detaljipiirroksen vaatimukset.....	5
3.1	Piirustusten sisältöä ohjaava lainsäädäntö	5
3.2	Rakenteiden lujuus ja vakaus	7
3.3	Paloturvallisuus	7
3.4	Kosteustekninen toiminta	10
3.5	Energiatehokkuus.....	12
3.6	Detaljipiirrosten esitystapavaatimukset	12
4	Autodesk Revit.....	16
4.1	Revitin parametrisuus	16
4.2	Revitin rakenne	17
4.3	Family category	18
4.4	Revitin työskentelytilat.....	19
5	Parametrisen detaljien kehittämisprosessi	21
5.1	Suunnitteluvaihe	22
5.2	Parametrisen päätyräystäsdetaljin toteutus.....	25
5.3	Parametrisen detaljien toiminnallisuus.....	28
5.4	Rakenteiden nimeäminen ja selitetekstit.....	32
5.5	Ohjelman asettamat rajoitukset	34
5.6	Toimeksiantajayrityksen rooli kehitystyössä.....	35
6	Lopputulokset ja johtopäätökset.....	36
6.1	Kehittämisprosessi	36
6.2	Revitin ja käyttäjän ominaisuudet prosessissa.....	37
6.3	Käyttömahdollisuudet	38
6.4	Jatkokehitysmahdollisuudet.....	39
7	Pohdinta	40

Lähteet	42
---------------	----

1 Johdanto

”Vanhassa vara parempi”. Vai onko? Vanha, tuttu tyyli toimia voi tuntua tehokkaalta ja parhaalta, koska se osataan ja siihen on totuttu. Uudet näkökulmat ja toimintatavat voivat tuntua ajatuksena raskailta eikä kehityksen mukaan ehkä lähdetä sen vuoksi. Rakennesuunnitelmien dokumentoinnissa tietokoneavusteisuus ja CAD-ohjelmistot ovat olleet arkipäivää jo vuosikymmeniä. BIM-käsite on myös ollut käytössä jo reilut kaksi vuosikymmentä. Nykyään tietomallinnus on arkipäivää useimmissa suunnittelutoimistoissa. CADilla tehtävät rakennesuunnitelmat ovat jäämässä osittain historiaan ja BIM-ohjelmat korvaavat vanhat menetelmät. Joiltakin osin kehitys voi laahata kuitenkin jäljessä. Edelleen suunnittelussa CAD- ja BIM-ohjelmat kulkevat joiltain osin rinnakkain. Rakennuksesta tehdään tietomalli, joka toimii suunnittelun pääasiallisena dokumentina. Kuitenkin rakennedetaljit voidaan vielä piirtää CADilla tai ottaa CAD-detaljikirjastosta. CADilla tehty detaljipiirros tuntuu vaivattomimmalta tavalta, eikä BIM-ohjelman mahdollisuuksia hyödynnetä. BIM-ohjelmien koko potentiaali jää tällöin hyödyntämättä. Tästä lähtökohdasta alkoi tämän opinnäytetyön suunnittelu.

Opinnäytetyön toimeksiantajayritys Insinööritoimisto Raksainssit Oy on nuori ja innovatiivinen suunnittelutoimisto. Yrityksen tavoitteena on olla nykyaikainen ja kehittyvä. Yrityksessä koetaan, että detaljointi AutoCADilla on ollut yksinkertaisin ja sujuvin tapa tähän asti. AutoCAD taipuu monenlaiseen työskentelyyn ja valmiina olevat detaljikirjasto ja eri kohteiden detaljit ovat nopeasti ja helposti hyödynnettävissä. Yrityksen toiveena olisi kuitenkin saada käyttämänsä BIM-ohjelman koko potentiaali käyttöön ja näin kehittää toimintatapojaan. Yrityksen käyttämällä Autodesk Revit -ohjelmalla on mahdollisuus tehdä ns. älykkäitä detaljeja (intelligent detailing), jossa yhdistellään tietomallin näkyymiin 2D-komponentteja. Opinnäytetyössä älykkäiden detaljien ajatusta kehitetään vielä eteenpäin. Ajatuksena on luoda parametrisesti toimiva 2D-detalji, jossa yhdestä kuvasta on mahdollista muokata erilaisia detaljeja valikon arvoja muuttamalla.

Opinnäytetyö on toiminnallinen kehittämistyö, jolla pyritään löytämään uusia toimivia metodeja toimeksiantajayrityksen käyttöön. Työn tavoitteena on selvittää parametrinen detaljin kehittämisprosessin vaiheet sekä selvittää ohjelman mahdollisuudet, rajoitukset ja kompromissit. Tarkoituksena on löytää ratkaisut ohjelman asettamiin rajoituksiin. Opinnäytetyöllä haetaan vastauksia seuraaviin kysymyksiin: Miten Revit-ohjelmalla voidaan luoda parametrisiä detaljeja? Mitä toiminnallisuutta on mahdollista sisällyttää parametriseen detaljiin? Mitä mahdollisuuksia ja rajoituksia ohjelma asettaa parametrisen detaljin toiminnalle? Voiko parametrisillä detaljeilla korvata Autocad-detaljikirjaston?

Opinnäytetyössä käydään läpi työn lähtökohdat toimeksiantajayrityksen näkökulmasta. Teoriaosiossa käsitellään detaljipiirrokselle asetettuja vaatimuksia sekä Revit-ohjelmaa yleisesti. Parametrisen detaljin kehittämistyön kuvauksessa käydään läpi prosessin vaiheet ja käytetyt menetelmät sekä prosessin aikana ilmenneet ongelmakohdat ja ratkaisut niihin. Johtopäätöksissä käsitellään parametrisen detaljin toteutusta, saavutettua toiminnallisuutta, Revitin ominaisuuksia prosessissa sekä käyttö- ja jatkokehitysmahdollisuuksia.

2 Yrityksen esittely ja työn tavoitteet

Opinnäytetyön toimeksiantaja on Raahessa toimiva Insinööritoimisto Raksainssit Oy. Yrityksen toiminta on käynnistynyt jo vuonna 2017, mutta nykymuodossa yritys on toiminut vuodesta 2023 lähtien. Yritys tarjoaa rakennus- ja rakennesuunnittelupalveluita sekä maaperätutkimuksia. Suunnittelupalveluita yritys tarjoaa valtakunnallisesti ja maaperätutkimusten toimialueena on Pohjois-Suomi. [1.]

Yrityksen toiminnan pääpaino on suunnittelupalveluiden tuottamisessa rakennusalan yrityksille ja yksityisille. Suurin osa yritysasiakkaista on pitkäaikaisia yhteistyökumppaneita, joille tuotetaan vakioituja ja jatkuvia suunnittelupalveluita. Suunnittelutoiminta keskittyy puurakenteisten rakennusten rakennus- ja rakennesuunnitteluun. Pääasiassa kohteet ovat P3-paloluokan rakennuksia eli autosuojia, halleja, toimitiloja, omakoti-, pari- ja rivitaloja, vapaa-ajan asuntoja sekä talousrakennuksia.

Yrityksen tavoitteena on kehittää toimintaansa tehokkaampaan, tuottavampaan ja nykyaikaisempaan suuntaan tehostamalla suunnittelua. Yksi keino tehostaa suunnittelua on ottaa käyttöön suunnitteluohjelmien koko potentiaali ja tehostaa suunnitteluprosessin ajankäyttöä. Opinnäytetyön tavoitteena on kehittää rakennesuunnitteluprosessia kehittämällä ja yksinkertaistamalla detaljipiirrosten muokkaamista. Tällä hetkellä rakennesuunnittelussa detaljipiirrokset tehdään pelkästään AutoCADilla. Yrityksellä on käytössään AutoCAD-detaljikirjasto, jossa on valmiina yleisimmät detaljit. Rakennesuunnitelmat tehdään detaljeja lukuun ottamatta Autodesk Revit -ohjelmalla. Yrityksen tavoitteena on siirtää detaljien piirtämisen painopistettä enemmän Revitin suuntaan ja käyttää enemmän ohjelman tarjoamia mahdollisuuksia.

2.1 AutoCAD-detaljikirjasto

Yrityksen suunnitteluprosessissa rakennesuunnitelmat tehdään Revitillä. Detaljit tehdään kuitenkin tällä hetkellä lähes poikkeuksetta Autocadilla, josta ne viedään Revit-projektiin 2D-kuvana. Yrityksen käytössä on detaljikirjasto, joka on tehty erään asiakasyrityksen kohteita ja rakenteita vastaavaksi. Detaljeissa rakennusmateriaalit ovat asiakasyrityksen käyttämien mukaisia. Myös detaljien rakenneratkaisut on suunniteltu yhteistyössä asiakasyrityksen kanssa. Rakennusmateriaalit ja rakenneratkaisut ovat vakioituja ja pysyviä.

Kuvien käytettävyyden ja löytymisen helpottamiseksi kuvat on ryhmitelty detaljikirjastoon eri kategorioiden mukaan. Pääkategoriat ovat räystäät (pääty ja sivu), väliseinät/välipohjat, perustukset ja saunat. Näiden pääkategorioiden alle detaljit on ryhmitelty alakategorioihin eri muuttujien mukaisesti. Kategoriat ovat muodostuneet asiakasyrityksen tarjoamien materiaalien ja rakenneratkaisuiden mukaisiksi.

Eniten erilaisia variaatioita on räystäskategoriassa. Räystäsdetaljien suuresta määrästä huolimatta kaikissa sen alakategorioissa on puutteita. Uusia detaljikuvia lisätään kirjastoon sitä mukaa, kun niitä piirretään rakennuskohteisiin. Asiakasyritys on myös hiljattain laajentanut räystäsmallistoaan. Tavallisen mittaisen räystäään lisäksi he tarjoavat asiakkailleen lyhyempää ja erittäin lyhyttä räystästä. Tämä muutos laajentaa detaljikirjastoa entisestään. Detaljeissa on paljon eri muuttujia ja niiden yhdistelmiä, jolloin jokaisesta eri rakenneosan ja rakennusmateriaalin yhdistelmästä pitää olla oma detalji. Ongelmaksi detaljikirjaston koko tulee esimerkiksi silloin, jos asiakasyritys vaihtaa jonkin rakennusosan. Tällöin detaljikirjaston päivittäminen on työläs ja aikaa vievä prosessi, sillä pahimmassa tapauksessa muutos voi vaikuttaa esimerkiksi kaikkiin räystäsdetaljeihin.

Tässä työssä tehtäväksi parametriseksi detaljiksi valikoitui päätyräystäsdetalji umpiräystäällä. Tällä hetkellä detaljikirjastossa on 133 kpl päätyräystäsdetaljia umpiräystäällä. On todennäköistä, ettei kaikkien 133 detaljin muuttujia saada toimimaan yhdessä parametriseissä detaljissa. Kuitenkin yhteen parametriseen detaljiin pystytään sisällyttämään jo useita kymmeniä kirjaston detaljeista.

3 Detaljipiirroksen vaatimukset

Rakentamista säätelevät lait, asetukset, standardit ja ohjeistukset asettavat vaatimuksia myös detaljipiirroksen sisällölle ja esitystavalle. Detaljipiirroksen tarkoitus on välittää yksityiskohtaista tietoa rakennuksen ja rakenteiden toteuttamistavasta. Rakennepiirustusten osana se ohjaa rakentamaan vaatimusten mukaisesti ja hyvää rakennustapaa noudattaen.

Detaljipiirros on kuvaus kohdasta, joka vaatii erityiskuvausta. Piirustuksen tarkoituksena on kuvata rakennettava kohde yksityiskohtaisesti ja käyttötarkoituksen vaatimalla tarkkuudella. [2, s. 2.] Standardin SFS-EN ISO 128-1:2020 mukaan teknisten piirustusten esitystapaohjeiden perusvaatimusten mukaan piirustuksen tulee olla:

- yksiselitteinen ja selkeä
- täydellinen; vain piirustukseen ja asiakirjoihin merkityt vaatimukset toteutetaan
- oikeassa mittakaavassa; tästä huolimatta piirustuksesta ei mitata mittoja
- monistamiseen ja kopiointiin soveltuva
- riippumaton kielestä; vältetään sanallisia kuvauksia
- standardin mukainen; sovellettava kansainvälinen standardi ilmoitetaan piirustuksessa. Poikkeuksellisesti Suomessa ei merkitä standardia eivätkä piirustukset ole täydellisesti standardin mukaisia.

Esitystapaohjeiden lisäksi voidaan soveltaa standardeista poikkeavia ja puuttuvia vakiintuneita paikallisia käytäntöjä.

3.1 Piirustusten sisältöä ohjaava lainsäädäntö

Rakennesuunnitteluun ja suunnitteluasiakirjojen tietosisältöön vaikuttavat eri lähteet. Lähtökohdan muodostavat rakentamista koskevat lait ja asetukset, joita on noudatettava kaikissa hankkeissa. Lakeja ja asetuksia täydentävät viranomaisten laatimat ohjeet, jotka eivät ole sitovia, mutta niistä poikkeaminen on mahdollista yleensä vain rakennuttajan tai viranomaisten suostumuksella. Rakennusalan toimijat ovat yhteistyössä laatineet standardeja, jotka koskevat mm.

suunnittelua, toteutusta ja tuotteiden valmistusta. Standardit liittyvät säädöksiin tai ovat erillisiä. Suunnittelun helpottamiseksi alan toimijat ovat myös laatineet ohjeita ja oppaita, jotka selkeyttävät ja täydentävät alan säädöksiä, standardeja ja ohjeita. [3, s. 26.]

Maankäyttö- ja rakennuslaissa (132/1999) määritellään rakentamista koskevat olennaiset tekniset vaatimukset, jotka koskevat rakenteiden lujutta ja vakautta, paloturvallisuutta, terveellisyttä, käyttöturvallisuutta, esteettömyyttä, meluntorjuntaa ja ääniolosuhteita sekä energiatehokkuutta [4, 117 a §–117 g §]. Käytännössä detaljikuvalla voidaan säädellä näitä kaikkia osaluueita.

Ympäristöministeriön asetus rakentamista koskevista suunnitelmista ja selvityksistä määrää, että rakennepiirustuksiin on rakennushankkeen laatu ja laajuus huomioon ottaen riittävässä laajuudessa sisällyttävä tieto rakenteiden lämmön, kosteuden, veden ja vedenpaineen, äänen sekä värinän eristyksen ratkaisusta [5]. Rakennepiirustuksissa esitetään yleensä rakennuksen perustaminen, rakenteet ja niiden sijoitus tunnuksineen, mitat ja yksityiskohdat, kuten raudoitukset, kiinnitykset, liitokset, reiät ja heikennykset, rakenteiden ja niiden materiaalien ominaisuudet kantavuuden, äänen-, lämmön-, kosteuden- ja vedeneristyksen, paloturvallisuuden ja säilyvyyden suhteen sekä henkilöturvallisuuden kannalta merkittävien kaiteiden, suojarakenteiden, tikkaiden, kattosiltojen yms. rakenne [6, s. 12].

Detaljipiirroksessa esitetään käytettävät rakennustuotteet. Ympäristöministeriön [7] mukaan rakennustuotteita ovat kiinteästi rakennuksen osaksi tulevat tuotteet, kuten betonielementit, ikkunat, teräsrakenteet ja sahatavara. Rakennustuotteella tarkoitetaan sellaista pysyvää rakennusosaa, rakennetta tai rakennuskohteeseen kiinteästi liittyvää tarviketta, tuotetta tai laitetta, jolle on säädetty maankäyttö- ja rakennuslaissa tai sen nojalla olennaisia teknisiä vaatimuksia. Rakennustuote voi tarkoittaa myös edellä tarkoitettua tuotteen asentamista varten tarvittavaa osaa. Rakennustuotteiden tulee olla turvallisia ja kestävän kehityksen periaatteiden mukaisia, eivätkä ne saa aiheuttaa haittaa terveydelle. Rakennustuotteet ovat rakentamiseen kelpoisia, kun ne täyttävät maankäyttö- ja rakennuslaissa tai sen nojalla säädetyt em. olennaiset tekniset vaatimukset. Rakennustuotteiden kelpoisuus osoitetaan CE-merkinnällä, jolloin tuotteen ominaisuudet ovat eurooppalaisen harmonisoidun tuotestandardin tai eurooppalaisen teknisen hyväksynnän mukaisia. [7.]

3.2 Rakenteiden lujuus ja vakaus

Maankäyttö- ja rakennuslain mukaan kantavien rakenteiden suunnittelun tulee perustua rakenteiden mekaniikan sääntöihin ja yleisesti hyväksytyihin suunnitteluperusteisiin taikka luotettaviin koetuloksiin tai muihin käytettävissä oleviin tietoihin. Rakennuksen rakentamisessa on käytettävä rakenteiden lujuuden ja vakauden kannalta soveltuvia rakennustuotteita. [4, 117 a §.]

Kantavat rakenteet esitetään tasopiirustuksissa, joka on vaakaleikkauspiirros rakennuksen rakenteista. Suurimittakaavaisilla detaljipiirroksilla tarkennetaan ja selkeytetään tasokuvissa esitettyjen rakenteiden yksityiskohtia. Detaljipiirroksilla voidaan kuvata tarkemmin esimerkiksi rakenteiden liitoksia ja raudoitusta. [3, s. 137, 138.]

Kantavissa rakenteissa käytettävien rakennusmateriaalien tulee täyttää maankäyttö- ja rakennuslaissa määrätyt tekniset vaatimukset. CE-merkintä osoittaa rakennusmateriaalin kelpoisuuden. Detaljipiirroksissa kantavina rakenteina käytetään CE-merkittyjä puutuotteita ja kiinnikkeitä. Rankarunkoisissa rakennuksissa yleisesti sahatavarana on käytössä C24, joka on yleisin lujuusluokka rakentamisessa [8]. Toimeksiantajayrityksen rakennesuunnitelmissa sahatavaran lisäksi kantavissa rakenteissa käytetään liimapuuta (GL30c) ja Kertopuuta (Kerto-S). Rakenteiden liitoksissa käytetään CE-merkittyjä nautoja, ankkurinauloja, ruuveja ja erilaisia rakennuskiinnikkeitä.

3.3 Paloturvallisuus

Rakennus on suunniteltava ja rakennettava käyttötarkoituksen mukaisesti paloturvalliseksi. Palon syttymisen vaaraa on rajoitettava. Rakennuksen kantavien rakenteiden on oltava sellaiset, että ne palon sattuessa kestävät vähimmäisajan ottaen huomioon rakennuksen sortuminen, poistumisen turvaaminen, pelastustoiminta ja palon hallintaan saaminen. Palon ja savun kehittymistä ja leviämistä lähitöllä oleviin rakennuksiin on pystyttävä rajoittamaan. Rakennuksen rakentamisessa on käytettävä paloturvallisuuden kannalta soveltuvia rakennustuotteita ja teknisiä laitteistoja. [4, 117 b §.]

Palo-osasto on rakennuksen sisäpuolinen tila, josta palon leviäminen on määrätyn ajan estetty osastoivin rakennusosin tai muulla tehokkaalla tavalla. Palo-osastoivan rakennusosan on tarkoitus määritetyn palonkestoajan rajoittaa palon ja savun leviämistä, turvata poistumista, helpottaa sammutus- ja pelastustoimia sekä vähentää omaisuusvahinkoja. [9, s. 40.]

YM:n asetus rakennusten paloturvallisuudesta (848/2017, 3§) määrää, että pääsuunnittelijan, rakennussuunnittelijan ja erityissuunnittelijan on tehtäviensä mukaisesti huolehdittava, että rakennus käyttötarkoituksensa mukaisesti täyttää olennaiset tekniset vaatimukset koskien paloturvallisuutta. Vaatimukset täyttyvät, jos rakennus suunnitellaan ja rakennetaan asetuksen mukaisesti tai perustuen oletettuun palonkehitykseen. [10, 3§.] Asetuksen mukaan suunniteltaessa rakennukselle määritetään paloluokka, joka määrittelee paloturvallisuusratkaisut.

Rakennuksen paloluokkia ovat P0, P1, P2 ja P3. Ympäristöministeriön asetus rakennusten paloturvallisuudesta [10, 4§–8§] määrittelee kriteerit P1-, P2- ja P3-paloluokille, joissa korkeimmat vaatimukset ovat P1-paloluokalla. Näissä paloluokissa rakennus suunnitellaan taulukkomitoituksena eli asetuksessa esitettyjä lukuarvoja käyttäen. Rakennuksen paloluokkaan vaikuttavat rakennuksen koko, kerrosluku, käyttötarkoitus ja käyttäjien lukumäärä. P0-paloluokkaa käytettäessä rakennus suunnitellaan toiminnallisena palomitoituksena eli osittain tai kokonaan oletettuun palonkehitykseen perustuvaa menettelyä käyttäen. Tässä opinnäytetyössä käsitellään ainoastaan P3-paloluokan kohteita, joiden vaatimukset ovat kevyimmät. P3-paloluokassa kantaville rakenteille ei ole erityisvaatimuksia palonkeston suhteen ja riittävä turvallisuustaso saavutetaan rakennuksen kokoa ja henkilömäärää säätelemällä.

Detaljipiirroksissa paloturvallisuus näkyy osastoivina rakenneosina. Kuvassa tulee näkyä rakenneosan palonkestovaatimusluokka sekä millä rakennusmateriaalien yhdistelmillä osastointi on toteutettu. Rakenteille asetetut palonkestovaatimukset ilmaistaan REI-luokkina. Palonkestävyys tarkoittaa rakennusosan kykyä täyttää määrätyn ajan kantavuusvaatimus (R) ja/tai osastoivuusvaatimus (EI) määritellyn kuormituksen ja paloaltistuksen vallitessa. REI-vaatimuksen perässä oleva numero ilmaisee rakenteen palonkesto-aikavaatimuksen minuutteina. Käytännössä esimerkiksi EI30-seinän tulee säilyttää palotilanteessa tiiviytensä ja eristävyytensä 30 minuuttia. [11.] P3-paloluokan rakennuksien rakenteille ei ole asetettu kantavuusvaatimuksia. EI30-rakenteen on kuitenkin kestettävä palokuorma sortumatta 30 minuuttia voidakseen täyttää EI30-vaatimus.

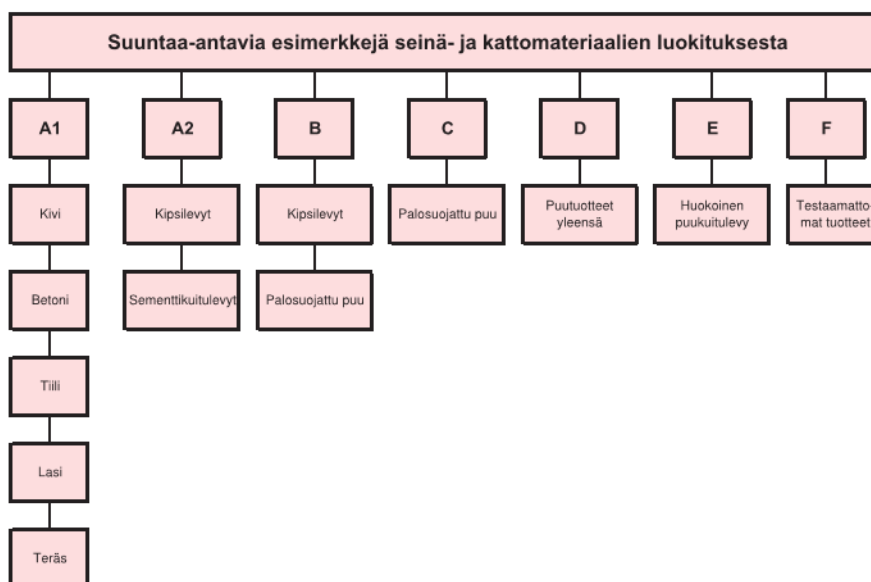
P3-Paloluokassa käytettävät osastoinnit tehdään rakenneosilla, jotka täyttävät vaatimuksen EI30, EI60 tai EI(M)60. Palo-osastoinnilla estetään palon leviäminen naapuritontilla tai omalla tontilla sijaitsevaan rakennukseen taikka rakennuksen sisällä toiseen palo-osastoon. Omalla tontilla ja rakennuksen sisällä olevat palo-osastoinnit toteutetaan aina EI30-luokan rakenneosilla. Naapuritontilla sijaitsevaan rakennukseen palon leviäminen estetään ulkoseinän vesikatteeseen asti yltävällä osastoinnilla. Seinärakenteen palonkestovaatimus voi olla EI30, EI60 tai EI(M)60.

Eurooppalaisessa luokituksessa rakennustarvikkeet jaotellaan eri luokkiin sen mukaan, miten ne käyttäytyvät syttymis- ja palotilanteessa. Luokituksessa tarkasteltavia ominaisuuksia ovat materiaalin syttymisherkkyys, palon leviäminen sekä savun ja palavien pisaroiden tuotto. Taulukossa 1 on luokittelukriteerit rakennustarvikkeille. Ainoastaan A1- ja A2-s1-, d0-rakennustarvikkeet toimivat kokonaisuudessaan samalla tavalla sekä palon alussa että myöhemmässä vaiheessa. Muissa luokissa rakennustuotteiden palonkestävyyttä on voitu lisätä esimerkiksi pinnoittamalla, mutta tuotteen ydin saattaa osallistua paloon myöhemmässä vaiheessa. [9, s. 23.] Taulukossa 2 esitetään seinä- ja kattomateriaalien luokittelu.

Taulukko 1 Rakennustarvikkeen luokkamerkinnän muodostuminen yleisesti [9, s. 23]

Osallistuminen paloon		Savun tuotto		Palavien pisaroiden ja osien tuotto	
Kuvaus	Merkintä	Kuvaus	Merkintä	Kuvaus	Merkintä
Ei osallistu paloon	A1	Erittäin vähäinen	s1	Ei esiinny	d0
Osallistuu erittäin rajoitetusti	A2	Vähäinen	s2	Nopeasti sammuvia esiintyy	d1
Osallistuu hyvin rajoitetusti	B	Muu kuin s1 tai s2	s3	Muu kuin d0 tai d1	d2
Osallistuu rajoitetusti	C				
Osallistuminen hyväksyttävää	D				
Käyttäytyminen hyväksyttävää	E				
Käyttäytymistä ei ole määritetty	F				

Taulukko 2 Seinä- ja kattomateriaalien jakautuminen luokkiin [9, s. 25]



3.4 Kosteustekninen toiminta

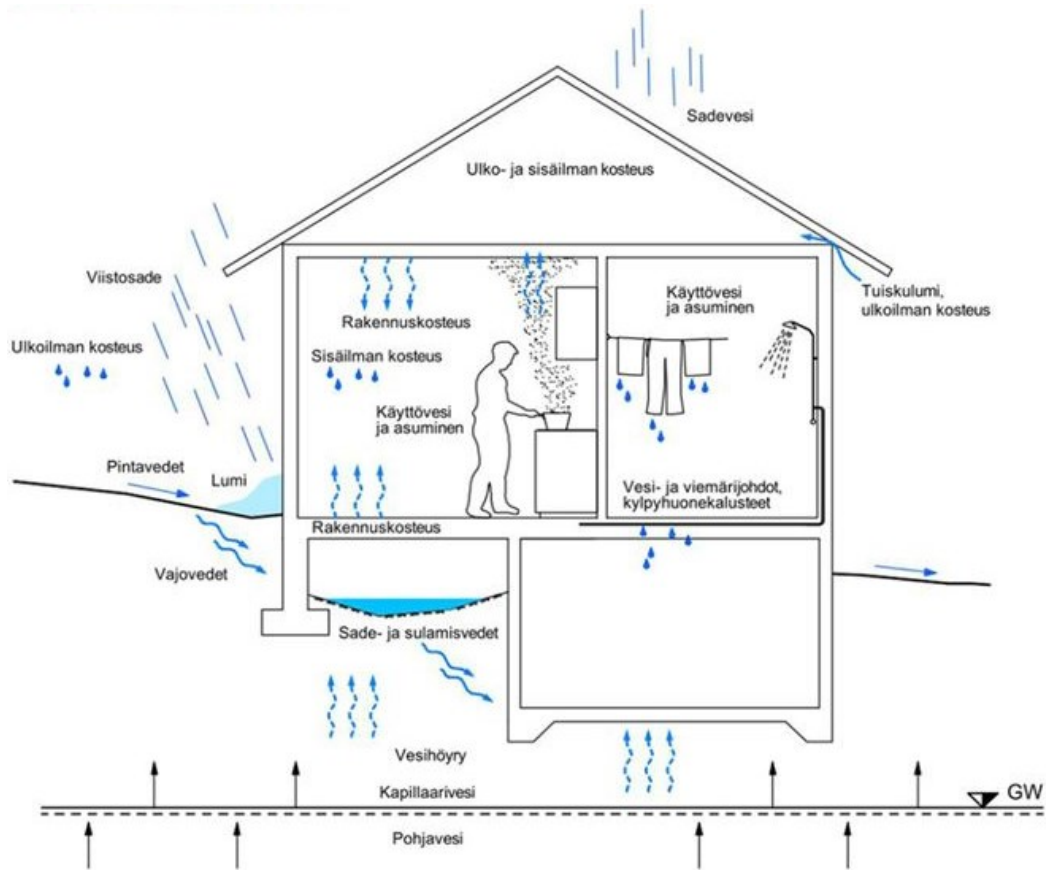
Detaljipiirros on olennainen osa rakennuksen kosteusteknisen toimivuuden suunnittelua ja toteuttamista. Detaljipiirroksessa suunnittelija määrittää ja esittää rakenneosat, materiaalit, liitokset ja niiden järjestyksen rakenteessa. Suunnittelijan vastuulla on rakennusfysikaalisesti oikein toimivan rakenteen suunnittelu.

Maankäyttö- ja rakennuslain [4, 117 c §] mukaan rakennus tulee suunnitella ja rakentaa niin, että se on terveellinen ja turvallinen rakennuksen sisäilman, kosteus-, lämpö- ja valaistusolosuhteiden sekä vesihuollon osalta. Rakennusosien ja rakenteiden kosteus ei saa aiheuttaa terveyden vaarantumista.

Ympäristöministeriön asetus rakennusten kosteusteknisestä toimivuudesta [12, 3 §] määrää rakennuksen kosteusteknisen toimivuuden suunnittelusta ja rakentamisesta. Asetuksen mukaan suunnittelijan on tehtäviensä mukaisesti huolehdittava rakennuksen kosteusteknisten olennaisien vaatimusten täyttymisestä. Rakennuksen on kestävä ja kestävä huomioiden oltava kosteusteknisesti toimiva suunnitellun käyttöajan ajan. Kosteus ei saa vaurioittaa rakennusta eikä aiheuttaa terveydellistä haittaa.

Rakennuksen sisäinen ja ulkoinen vesihöyry, vesi, lumi tai jää ei saa kulkeutua rakenteisiin rakennuksen vaipan, ikkunoiden, ovien tai muiden rakenteiden tai laitteiden kautta. Rakennuksen vaipan on suojattava rakenteita tuulen, viistosateen ja tuulenpaineen vaipan pintaa pitkin siirtäältä vedeltä. Rakennuksen vaipan sisään päässeeseen kosteuden on voitava poistua aiheuttamatta haittaa rakenteille. [12, 5 §.]

Rakenteet tulee suojata painovoimaisesti tai kapillaarisesti siirtyvältä vedeltä sekä konvektiolla eli ilmavirran mukana ja diffuusiolla eli vesihöyryn osapaine-erojen mukaan siirtyvältä vesihöyryltä. Painovoimaisesti rakenteisiin pääsee kosteutta esimerkiksi katto- ja seinävuotojen, putkivuotojen ja kosteiden tilojen vaurioiden kautta. [13, s. 367.] Kapillaarisesti kosteus voi päästä rakenteisiin esimerkiksi maaperästä alapohjan ja perustusten kautta. Kuvassa 1 on esitetty rakennuksen kosteuskäytännöt.



Kuva 1 Rakennuksen kosteuslähteet [14, s. 97]

Vaipparakenteiden oikeanlainen kosteustekninen suunnittelu estää kosteusvaurioiden syntymistä. Puuta ja puupohjaisia rakenteita ei saa jättää kahden vesihöyrytiivin kerroksen väliin ilman tuuletusta. Vaipparakenteessa tulee olla yhtenäinen ja tarpeeksi leveä tuuletusväli lämmöneristeen kylmällä puolella. Ulkoseinärakenteen tulisi harventua sisältä ulospäin niin, että rakennekerrosten vesihöyrynläpäisevyys kasvaa ulospäin mentäessä. [13, s. 367–368.]

Konvektiolla ja diffuusiolla siirtyvän vesihöyryn pääsy rakenteisiin estetään asentamalla yhtenäinen ilman- ja höyrynsulkukerros vaipparakenteen sisäpintaan eli lämmöneristeen sisäpuoleiseen pintaan. Ilmansulku estää konvektiona siirtyvän kosteuden pääsemisen rakenteisiin. Höyrynsulkukerroksella on oltava riittävä vesihöyrynvastus, jotta se voi estää diffuusiolla siirtyvän kosteuden pääsemisen rakenteisiin. Ilman- ja höyrynsulkuna voidaan käyttää yhtä kalvokerrosta tai kerrokset voivat olla myös erillisiä. Kerrosten tulee olla yhtenäinen ja ilman- ja höyrynsulun liitokset ja läpiviennit tulee tiivistää huolellisesti, joko puristettuna tiukasti kahden rakenteen väliin tai teippaamalla saumat höyrynsulkuteipillä. Liitosten ja detaljien huolellinen kosteustekninen suunnittelu ja toteutus estävät kosteusvaurioiden syntymistä. [13, s. 367, 369.]

3.5 Energiatehokkuus

Maankäyttö ja rakennuslain [4, 117 g §] mukaan rakennushankkeeseen ryhtyvällä on huolehtimisvelvoite rakennuksen energiatehokkuudesta. Rakennus on suunniteltava ja rakennettava niin, että energiaa ja luonnonvaroja käytetään säästeliäästi. Energiatehokkuus on pystyttävä osoittamaan laskelmilla, joissa otetaan huomioon rakennuksessa käytettävät energiamuodot, uusiutuvien energiamuotojen osuus, käytettävät rakennustuotteet, talotekniset järjestelmät sekä vallitsevat olosuhteet.

Rakennesuunnittelijan rooli rakennuksen energiatehokkuuden suunnittelussa on tärkeä. Suunnittelijan määrittämä vaipparakenne vaikuttaa rakennuksen energiankulutukseen. Rakenne- ja materiaalivalinnoilla suunnittelija voi myös vaikuttaa rakennuksen energiatehokkuuteen. [3, s. 40.]

Käytännössä energiatehokkuus näkyy rakenneteknisissä detaljipiirroksissa rakennuksen lämmöneristyksen ja ilmatiivistyksen ratkaisuissa sekä näiden liitoksissa. Lisäksi detaljeissa on huomioitava rakenteen mahdolliset kylmäsillat sekä kylmäsiltojen katkot. Kyseiset tekijät vaikuttavat rakennuksen lämpöhäviöön, jolla on oleellinen merkitys rakennuksen energiatehokkuudessa.

3.6 Detaljipiirrosten esitystapa-vaatimukset

Rakennuspiirustusten esitystapaohjeita määrittävät ISO-standardit. Joissain tapauksissa voidaan standardeista poiketen soveltaa vakiintuneita paikallisia ja kansallisia käytäntöjä piirustusten toteuttamisessa. Esitystapamäärittelyihin kuuluvat kuvausmenetelmät, käytettävät mittakaavat, viivalajit ja tekstien esitystapa. ISO-standardeja ei ole tarkoitettu mallintamalla tapahtuvaan suunnitteluun. Tietomallista luotujen kuvien on kuitenkin noudatettava samoja esitystapa-vaatimuksia. [2, s. 1–2.]

Rakennuspiirustusten mittakaavoja ei ole määritelty ISO-standardissa. Mittakaava valitaan piirroksessa vaadittavan esitystarkkuuden sekä piirustuksen käyttötarkoituksen mukaisesti. Taulukossa 3 on detaljipiirroksessa käytettäviä mittakaavoja. [2, s. 6.] Yrityksen asiakasyritykselle tuottamat detaljipiirrokset liitetään yleensä yhdistelmäpiirroksiksi detaljeja vastaavan tasopiirroksen kanssa. Tällöin detaljipiirrokset ovat mittakaavassa 1:20. Detaljipiirrosten ollessa omilla erillisillä piirustuslehdillään on mittakaava yleensä 1:10.

Taulukko 3 Työpiirustuksissa (detaljipiirustukset) käytettäviä mittakaavoja [2, s. 6]

piirustus	mittakaava/ peruskäyttö	mahdollinen käyttö
rakennus-	1:20	1:2 (ei suositella)
osat, muut	1:10	
suunnitelman	1:5	
yksityiskohdat	1:1	

Rakennuspiirustuksissa käytetään ISO 128-20 - tai RT 15-10635 -mukaisia viivoja. Piirroksissa käytetään taulukossa 4 esitettyjä rakennusosia kuvaavia viivoja sekä taulukossa 5 esitettyjä apuviivoja. Rakennusosia rajaavien viivojen tulee erottua selkeästi apuviivoista. Samaan tarkoitukseen käytetyn viivalajin tulee olla rakennuskohteen kaikissa kuvissa samanlainen. Viivojen käyttötarkoitus voidaan merkitä kuvaan. [2, s. 6.]









Taulukko 4 Rakennusosia kuvaavat viivat rakennuspiirustuksissa [2, s. 6]

esimerkki	viivalaji	viivaleveys	käyttö; rakennusosien kuvauksien viivat
	Ehyt viiva	leveä	Näkyvät reunat ja rajat
	Katkoviiva	leveä (tai kapea)	Näkyvän pinnan takana olevat reunat ja rajat
	Katkoviiva	kapea	Tilavaraus ¹
	Pistekatkoviiva	leveä (tai kapea)	Kuvaustason edessä ja yläpuolella olevat reunat ja rajat
	Ehyt viiva	kapea (tai leveä)	Pintojen jako
	Ehyt viiva	erittäin leveä	Leikkauspinnan reunat ja rajat korostettaessa Säilytettävän rakennusosan oikea ja alareuna
	Pisteviiva ²		Purettavat rakennusosat ¹

1) Ei sisälly standardiin ISO 128

2) CAD-piirustuksissa pisteosa tehdään lyhyinä viivanosina, jotta niiden leveys noudattaisi muiden viivajaksojen leveyttä

Taulukko 5 Rakennuspiirustusten apuviivat [2, s. 7]

esimerkki	viivalaji	viivaleveys	käyttö; apuviivat
	Ehyt viiva	kapea	Toiminnan osoittaminen (esimerkiksi oven aukeaminen) Leikkauspinnan tehosteet Mittaviivat Moduuliviivat Viittausviivat
	Pistekatkoviiva ³	kapea	Katkaisu, kuvauksen rajaus ¹ Keskiviivat Symmetriaviivat Moduuliviivat
	Pistekatkoviiva ³	leveä	Leikkaustason merkintä
	Pistekatkoviiva ³	erittäin leveä	Mitoituksen peruslinjan viiva
	Kaksipistekatkoviiva	kapea	Rajoittuvien rakennusosien reunat ja rajat (eivät kuulu kuvattavaan kohteeseen, mutta ovat tarpeen yhteyden ymmärtämiseksi) Vaihtoehtoinen sijainti tai liikkuvan osan ääriasennot Painopisteakseli
	Kaksipistekatkoviiva ³	leveä	Muutosalueen rajaus ¹
	Kolmipistekatkoviiva ³	leveä	Paloteknisten osastojen rajat ¹
	Siksak-murtoviiva	kapea	Kuvauksen rajaus, katkaisu ⁴

1) Ei sisälly standardiin SFS-EN ISO 128

2) CAD-piirustuksissa pisteosa tehdään lyhyinä viivanosina, jotta niiden leveys noudattaisi muiden viivajaksojen leveyttä

3) Pitkät viivaosat

4) Standardin SFS-EN ISO 128 mukainen merkintä, Suomessa käytetään yleensä pistekatkoviivaa.

Piirustuksissa käytettäviä viivapaksuuksia ovat yleensä:

- ohut viiva, paksuus 0,25 mm
- keskipaksu viiva, paksuus 0,35 mm
- paksu viiva, paksuus 0,5 mm

Mitta- ja apuviivat sekä toisarvoiset viivat esitetään ohuella viivalla. Keskipaksulla viivalla esitetään ääri viivat ja paksulla viivalla korostetut viivat. Vaihtoehtoisesti voidaan käyttää taulukon 6 viivapaksuuksia. Viivaryhmien paksuuden suhde on 1:2:4. [3, s. 105–106.]

Taulukko 6 Viivaryhmät ja viivaleveydet millimetreinä [2, s. 6]

I	II	III	IV
(0,13)	0,18	0,25	0,35
0,25	0,35	0,5	0,7
0,5	0,7	1,0	1,4

Kuvan 2 ainemerkinnöillä lisätään leikkauspintojen selvyyttä ja esitetään leikatun pinnan rakennekerrosten materiaali [2, s. 9]

	Täytemaa		Betoni
	Sora		Betoni (CAD-merkintä)
	Maanpinta		Kevytbetoni
	Maanpinta		Kevytsora ja kevytsorabetoni
	Kallio		Kevytsoraharkko
	Louhittu kallio		Muuraus tiilistä tai harkoista, erilaiset muurukset voidaan erottaa eri viivatiheyksillä
	Puu, yleinen		Tulenkestävä muuraus
	Puu, detaljipiirroksessa		Luonnonkivi
	CLT-levy, detaljipiirroksessa		Kova eriste, eriste yleensä
	LVL-levy, detaljipiirroksessa		Pehmeä eriste
	Puusoiron leikkaus, detaljipiirroksessa		Vedeneriste

Kuva 2 Ainemerkintöjä

Piirustusten tekstien tulee olla selkeää, helppolukuista ja yhtenäistä pienaakkostekstiä (gemena) ja otsikoissa suuraakkosia (versaalia). Tekstityyppinä suositellaan käytettäväksi päätteetöntä kirjasintyyppiä kuten Arial, Helvetica tai DIN. Tekstikoon ja -tyylin käyttö on vapaata, mutta piirustussarjassa saman tyyppinen sisältö tulee kirjoittaa yhteneväisesti ja liian pientä tekstikokoa tulee välttää. [2, s. 10.]

4 Autodesk Revit

Opinnäytetyön parametrinen detalji luodaan Autodesk Revit-ohjelmalla, joka on tietomallinnus- eli BIM-ohjelmisto (Building Information Modeling). Ohjelmalla voidaan luoda tietomalli eli todellisuutta vastaava 3D-malli rakennuksesta. Rakennusprojektissa samaa tietomallia on mahdollista käyttää sekä arkkitehti-, rakenne- että yhdyskuntatekniseen suunnitteluun koko rakennuksen elinkaaren ajan. Tietomallin etuja ovat parametrinen tarkkuus muodoissa, rakenteissa ja järjestelmässä. [15.]

4.1 Revitin parametrisuus

Parametri on lukuarvo, muuttuja tai määre, joka ohjaa ja informoi suunniteltavan algoritmisen prosessin osaa. Parametri voi määrittää esimerkiksi rakennuksen korkeutta tai ulkomittoja. Parametrin muutos päivittää koko prosessin ja sen pohjalta myös lopputuloksen. [16, s. 13–25.] Revitissä parametrit sisältävät tiedon objektin ulkomuodosta ja ominaisuuksista. Parametri tarkoittaa muuttujaa, joka määrittää objektin tietyn ominaisuuden tai ilmenemistavan. Revitissä parametri voi säädellä esimerkiksi objektin mittoja, materiaalia, ulkoasua tai näkyvyyttä. Lisäksi parametrien sisältämää tietoa voidaan käyttää objektien tunnisteissa ja luetteloissa. [15.]

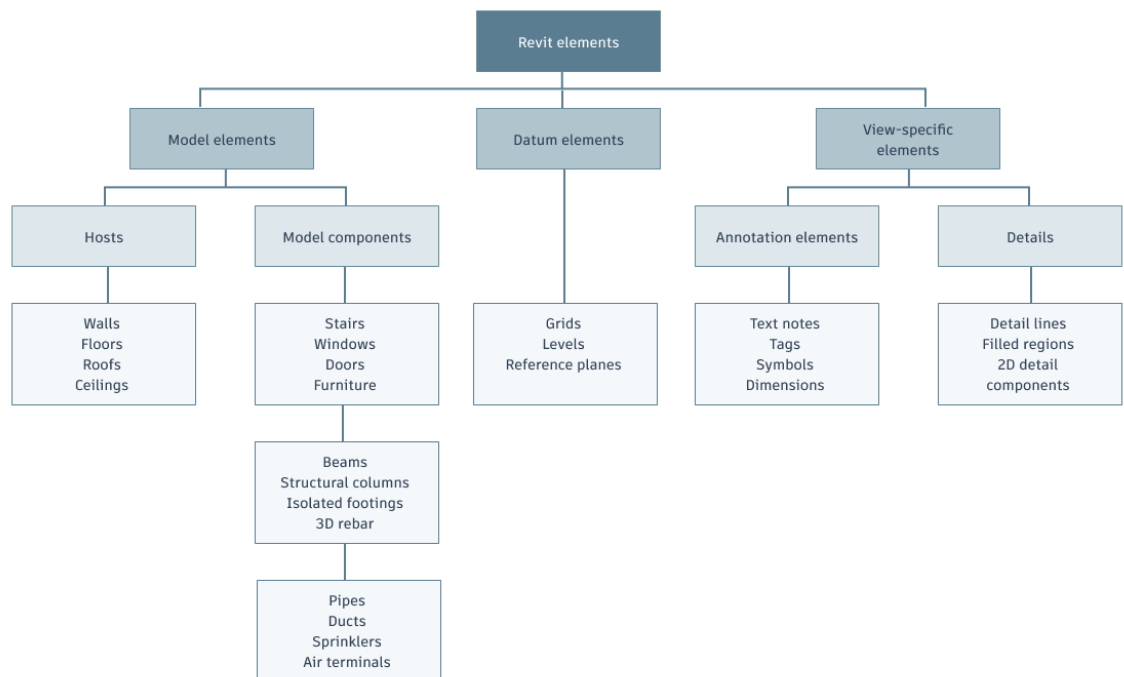
Parametrinen mallintaminen tarkoittaa riippuvuussuhteiden rakentamista suunnittelumallin eri geometrysten osien ja parametrien välille. Parametrin mallintamisen avulla suunnittelija voi rakentaa algoritmiseen prosessiin perustuvan kolmiulotteisen mallin, jossa tietyn prosessin osan tai parametrin muuttaminen muokkaa tietomallin geometriaa reaaliajassa. [16, s. 13.] Parametrinen malli rakentuu linkitettyjen komponenttien muodostavan algoritmisen logiikan ja sille syötettyjen parametrien avulla. Parametrin muutos päivittää koko prosessin ja sen pohjalta myös lopputuloksen. Parametrinen mallintaminen nopeuttaa huomattavasti suunnittelun variaatioiden ja vaihtoehtojen etsintää. [16, s. 25.]

Revitin toiminta on parametrista. Ohjelmassa rakennus mallinnetaan yhdistelemällä parametrisiä elementtejä, joista muodostuu tietomalli. Revitissä on elementtien välisiä riippuvuussuhteita, jotka sisältyvät automaattisesti ohjelmaan tai ovat ohjelman käyttäjän määrittämiä. Elementtien välisten riippuvuuksien avulla Revit pystyy reaaliaikaisesti koordinoimaan muutoksia ja ylläpitämään johdonmukaisuutta koko tietomallissa. Revit-tietomallissa jokainen 2D- ja 3D-näkymä,

luettelo ja määräluettelo on tietoesitys samasta virtuaalirakennusmallista. Kun tietomallia muokataan, päivittyvät muutokset kaikissa näkymissä. [15.]

4.2 Revitin rakenne

Revitin koko järjestelmä ja toiminta perustuvat ohjelman elementteihin, joita kutsutaan familyiksi. Ohjelman ymmärtäminen vaatii käsitystä ohjelman rakenteesta, toiminnoista ja familyistä. Tietomalli rakentuu yhdistelemällä eri familyitä yhdeksi kokonaisuudeksi. Revit sisältää kolmentyyppisiä elementtejä, joita ovat *Model elements*, *Datum elements* ja *View-specific elements*. Kuvassa 3 on esitetty kaavio Revitin elementeistä. [15.]



Kuva 3 Revitin elementit [15.]

Model elementit ovat 3D-familyitä, joista rakennuksen geometria muodostuu. Tähän ryhmään kuuluvat kaikki fyysiset rakenneosat, kuten esimerkiksi seinät, lattiat, katot, ikkunat, ovet, huonekalut, kiintokalusteet, rakenneosat ja talotekniset laitteet. *Model elements* -ryhmään kuuluvia familyitä on kahden tyyppisiä, *Hosts* ja *Model components*. *Host*-familyt ovat rakennusosia, joista muodostuu rakennuksen fyysinen ulkomuoto, kuten esimerkiksi seinät, katto ja välipohja. *Model components* -familyt ovat kaikki muut rakennukseen liitettävät rakennusosat tai muut elementit, kuten esimerkiksi ikkunat, ovet tai kalusteet. [15.]

Datum element -familyt luovat tietomallille rajat ja ympäristön, johon familyitä sijoitetaan. *Datum element* -familyitä ovat esimerkiksi mallinnustilassa olevat *Gridit* (ruudukot), *Levelit* (tasot) ja *Reference planet* (referenssitasot). [15.]

Tietomallista saadaan luotua useita eri näkymiä, kuten esimerkiksi pohjapiirros tai leikkauskuva. *View specific elementit* (näkökohtaiset elementit) ovat näkyvissä vain siinä näkymässä, johon ne on luotu. Nämä elementit ovat 2D-familyitä, joita on kahta eri tyyppiä, *Annotation elementit* (merkintäelementit) ja *Detailit* (yksityiskohtat). *Annotation elementeillä* dokumentoidaan näkymiin tietoja, joita tietomalli ja rakenneosat sisältävät. Näitä ovat esimerkiksi mitat, tunnisteet ja selitetekstit. *Annotation elementit* skaalautuvat näkymän mittakaavan mukaisesti. *Detailit* ovat 2D-familyitä, joita voidaan käyttää näkymissä yksityiskohtien esittämiseen ja detaljointiin. Näitä ovat esimerkiksi erilaiset *Detail linet* (viivat), *Filled regionit* (täyttökuviot) ja *Detail componentit* (yksityiskohtakomponentit). [15.] Opinnäytetyössä luotava parametrinen detalji kuuluu *Details*-ryhmään.

4.3 Family category

Family on ryhmä, jossa elementtien ominaisuudet ovat keskenään yhteneväisiä. Nämä ominaisuudet määritetään parametreillä, joita ovat esimerkiksi mitat, materiaalit ja *nested elements* (sisäkkäiset elementit). [15] [17, s. 634.] Samaan familyyn voi kuulua hyvinkin erilaisia elementtejä. Saman familyn elementeillä on kuitenkin samantyyppinen käyttötarkoitus ja toiminnallisuus. Esimerkiksi huonekalu-family voi sisältää hyvinkin erilaisia huonekaluja, mutta ne kuuluvat kaikki samaan huonekaluperheeseen. Familyn sisällä voi olla erilaisia *family typejä* (family-tyyppi). Esimerkiksi erilaiset kalusteet ovat huonekalu-familyn eri tyyppiä ja niiden parametrit voivat toimia eri tavoilla.

Ohjelmassa on kolmenlaisia familyitä: *system family* (järjestelmäperhe), *loadable family* (ladattava perhe) ja *In-Place family* (paikallinen perhe). Ohjelma sisältää mallintamiseen tarvittavia *system familyitä*. Rakennuksen ulkomuoto rakentuu tietomalliin *system familyistä*, jotka ovat rakennuksen peruselementtejä. Näitä ovat esimerkiksi seinät, lattiat, katto ja portaat. *System familyihin* kuuluvat myös tietomallille rajat luovat *gridit*, *levelit* ja *reference planet*. Uusia *system familyitä* ei voi luoda, mutta jo olemassa olevia voi kopioida ja muokata. *System familyn* elementit voivat toimia *host familynä* muille elementeille, yleensä *loadable familyille*. Esimerkiksi seinä-family toimii *hostina* ikkunoille ja oville. Tämä tarkoittaa sitä, että ikkunan tai oven voi lisätä vain seinään

ja ne tarvitsevat seinän ollakseen olemassa. Seinän poistaminen tietomallista poistaa myös siinä olevat ikkunat ja ovet. [15.]

Ohjelman mukana tulee kirjasto, joka sisältää *loadable familyitä*, joita voidaan lisäksi ladata ulkopuolisista kirjastoista tai lähteistä. Lisäksi verkossa on useita sivustoja, joissa on tarjolla ilmaisia, valmiita family-komponentteja. [15.] Prodlib-kirjasto on kotimainen komponenttikirjasto, josta voi ladata eri rakennusmateriaalien valmistajien tuotteita dwg- ja rfa-tiedostoina.

In-place element on tiettyyn projektiin kertaluonteisesti luotava 3D-objekti. *In-place element* on *familyn* kaltainen, mutta se on oma yksittäinen elementtinsä, josta ei voi tehdä useita *family* -tyyppejä kopiaimalla. *In-place elementtiä* voi kopioida, mutta kopioidut elementit toimivat omina erillisinä objekteinaan. *In-place elementin* perhetyyppiä ei voi myöhemmin muuttaa. Elementin luomiseen käytettävät työkalut ovat vastaavat, kuin muutkin family-piirtotyökalut. [15.]

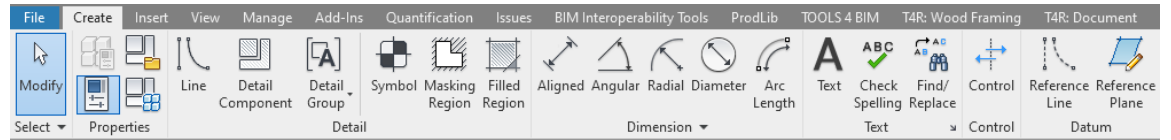
Opinnäytetyössä luotava parametrinen detalji on *loadable family*. Tarkemmin detalji kuuluu *Detail item* -famiylihin, jotka ovat viivapohjaisia 2D-famiyliä. *Detail itemejä* käytetään Revitissä eri näkymien detaljointiin. *Detail itemit* ovat näkymäkohtaisia famiylitä eli ne näkyvät vain näkymissä, joihin ne on lisätty. Niillä ei ole yhteyttä tietomalliin, vaan niiden tarkoitus on esittää yksityiskohtia rakenteista halutuissa näkymissä. Objektit skaalautuvat tietomallin mukaan, vaikka ne eivät ole yhteydessä siihen. [15.]

4.4 Revitin työskentelytilat

Revit sisältää kaksi erillistä työskentelytilaa: projektitila ja *Family Editor* (Family muokkaustila). Projektitila on mallinnustila, jossa luodaan tietomalli käyttäen Revitin sisältämiä family-objekteja. *Family Editor* on työskentelyalusta, jolla voidaan luoda uusia famiylitä tai muokata tietomallissa käytettäviä famiylitä. Ulkoasuiltaan työskentelytilat ovat samankaltaisia, mutta niissä käytettävissä työkaluissa on eroavaisuuksia. Projektitila ja *Family Editor* eivät ole yhteydessä toisiinsa, mutta käyttäjällä voi olla Revitissä käytössä molemmat työskentelytilat samanaikaisesti. Projektitilasta pääsee tarvittaessa suoraan *Family Editoriin* muokkaamaan komponentteja. Myös *Family Editorista* voi suoraan ladata muokkaamansa tai luomansa famiilyn samanaikaisesti käytössä olevaan projektiin. [15.]

Tässä opinnäytetyössä parametrisen detaljin kehitysprosessi tehdään *Family Editorilla*. Työskentelyalustan sisältämät työkalut vaihtelevat riippuen famiylstä, jota ollaan luomassa. Jokaiselle eri

ryhmään kuuluvalle familylle on oma familytemplate eli aloituspohja, joka määrittää, mitä työkaluja *Family Editorissa* on käytössä. 3D-familyille on käytössä työkaluja, joilla voidaan luoda 3D-geometriaa. Opinnäytetyössä tehtävä parametrinen detalji on *Detail item* -family, jonka työstämiseen on käytössä vain 2D-työkaluja (kuva 4). [15.]



Kuva 4 *Detail item* -familiarin työkalut *Family Editorissa*

Family Editorissa määritetään objektin parametrisuus, jonka mukaan family toimii projektitilassa. Erilaisia parametrejä on *Family Editorissa* valittavana kymmeniä. Opinnäytetyössä käytettävät parametrit rajautuvat niihin, jotka vaikuttavat familiarin mittoihin, näkyvyyteen ja objektien määrään. Parametrien toimintaa kontrolloimaan voidaan lisätä Revit-*formuloita*, jotka ovat joko numeerisia, matemaattisia laskutoimituksia tai riippuvuuksia määrittäviä ehtolausekkeita [15].

5 Parametrisen detaljin kehittämisprosessi

Toimeksiantajayrityksen halu kehittää työskentelytapojaan rakennesuunnitteluprosessissa oli lähtökohdana tässä opinnäytetyössä. Detaljipiirroksia lukuun ottamatta suunnitteluprosessissa dokumenttien työstäminen tehdään Revitillä. Tästä lähtökohdasta lähdettiin miettimään, miten Revitillä voitaisiin tehdä kaikki rakennesuunnittelun dokumentit. Revitin perinteinen detaljointimenetelmä ei suoraan sovi yrityksen rakennesuunnitteluprosessiin. Tässä menetelmässä jokainen detalji pitäisi tehdä erikseen, eikä se toisi ajankäytöllistä etua verrattuna AutoCAD-detaljeihin. Revit-detaljeista kootun detaljikirjaston kerääminenkin olisi aikaa vievä prosessi, koska jokainen detalji eri muuttujilla pitäisi tehdä erikseen. Tällaisen Revit-detaljikirjaston koko kasvaisi myös Autocad-detaljikirjastoa vastaavaksi. Yrityksessä toivottiin Revit-detaljointiin vastaavaa helpoutta kuin Autocad-detaljikirjaston valmiita detaljeja käytettäessä. Toiveena oli siis saada valmiita detaljeja, joiden muokkaaminen tarvittaessa olisi yksinkertaista ja nopeaa.

Toimeksiantajayrityksen suunnittelupäällikkö esitteli ajatuksen toiminnallisista detaljeista, joita Revitillä mahdollisesti pystyisi rakentamaan. Yrityksessä ei ollut tehty taustatutkimusta aiheeseen liittyen eikä ollut mietitty, millaista toiminnallisuutta detaljiin haluttaisiin ja voitaisiin rakentaa. Tiedonhakuvaiheessa kävi ilmi, ettei vastaavaa toiminnallista detaljia ollut tehty aikaisemmin. Aiheen käyttökelpoisuus yritykselle oli siis todennettava käymällä läpi toiminnallisen detaljin kehittämisprosessi. Kehittämisprosessin tulosten perusteella yritys pystyisi miettimään suuntaa, johon detaljipiirrosten ja detaljikirjaston kanssa edetään.

Opinnäytetyön alkuperäisenä tavoitteena oli luoda yritykselle parametrinen detaljikirjasto. Työn suunnitteluvaiheessa kuitenkin todettiin tämän olevan liian työläs toteutettavaksi yhdessä opinnäytetyössä. Pääasiallisena tavoitteena oli kuitenkin luoda perusta parametrille detaljikirjastolle. Tämä tavoite saavutettaisiin jo sillä, että käytäisiin läpi yhden parametrin detaljin kehittämisprosessi. Opinnäytetyöstä saatujen tietojen pohjalta voitaisiin tehdä päätös, ovatko parametriset detaljit ja detaljikirjasto se suunta, johon yritys haluaa jatkossa panostaa.

Opinnäytetyön aiheen tarkentumisen jälkeen tavoitteet määriteltiin uudelleen koskemaan yhtä parametristä esimerkkidetaljia. Tavoitteet määriteltiin kahdella tasolla. Ylemmällä tasolla tavoitteena oli saada tietoa toimeksiantajayritykselle parametristen detaljien ja detaljikirjaston jatkokehitykseen. Tavoitteena oli selvittää parametrin detaljin kehittämisprosessin vaiheet sekä ohjelman mahdollisuudet ja rajoitukset. Tavoitteena oli myös löytää ratkaisut ja kompromissit mahdollisiin ongelmakohtiin. Alemman tasot tavoitteet koskivat työssä kehitetyn parametrin

detaljin toiminnallisuutta, vaatimustenmukaisuutta ja ulkoasua. Tavoitteena oli saada aikaan rakenneteknisesti toimiva ja ulkoasultaan selkeä detalji, jonka toiminnallisuus sisältäisi yleisimmät muuttujat Autocad-detaljikirjaston kuvista. Toiminnallisuuden tulisi olla sujuvaa eri toimintojen välillä. Lisäksi tavoitteena oli, että detaljin käyttö olisi helppoa ja selkeää Revitin projekttillassa.

5.1 Suunnitteluvaihe

Suunnitteluprosessi aloitettiin listaamalla yrityksen käyttämästä detaljikirjastosta räystäsdetaljien yleisimmät muuttujat (Taulukko 7). Muuttujien perusteella tehtiin suunnitelma, mitkä muuttujat toimivat ns. familyä määrittävänä eli näihin muuttujiin ei tullut parametrisyyttä. Lähtökohteisesti tarkoituksena ei ollut saada kaikkia muuttujia yhteen detaljiin. Rajoittamalla muuttujien määrää, detaljin toiminnallisuuteen ei tule liikaa päällekkäisyyksiä eikä detaljista tule liian monimutkaista ja vaikeasti toteutettavaa. Kehittämisprosessin esimerkkidetaljiksi valittiin päätyräystäsdetalji umpiräystäsrakenteella. Nämä ovat ne familyä määrittävät muuttujat, joissa ei ole parametrisyyttä. Poissulkien edellä mainitut, katosrakenteen ja sivuräystäisiin liittyvät muuttujat, tavoitteena oli sisällyttää detaljiin kaikki taulukossa 7 olevat muuttujat.

Taulukko 7 Räystäsdetaljien muuttujat

Räystäät:	Pääty	Sivu		
Rakenne:	Umpi	Avo		
Pituus:	4 lautaa	2 lautaa	1 lauta	
Katemateriaali:	Pelti	Huopa	Tiili	
Kattokulma:	1:2,5/1:1,5	1:4	1:7/1:9	
Runkovahvuus:	122 mm	147 mm	197 mm	Vaakakoolaus +47 mm
Rakenne:	Eristetty	Kylmä	Katos	Muut
Palo-osastointi:	Osastoimaton	EI30	EI60	
Osastoitu rakenne	Sisäkatto	Päätyseinä	Sivuseinä +räystäs	

Suunnitteluvaiheessa parametriseen detaljiin haluttiin sisällyttää seuraavanlaisia muuttujia:

- 4 erilaista vesikatetta: rivipeltikate, tiilikuviopeltikate, tiilikate, huopakate
- Ruoteen dimension muokkausmahdollisuus
- Räystäslautojen lukumäärä ja muuttuva räystäsmitta
- Yläpohja: eristetty rakenne vaihtelevalla eristepaksuudella, eristämätön rakenne
- Ulkoseinä: seinän rakennekerrosten dimensioiden muokattavuus, eri seinärakennetyypit: eristetty, eristämätön, EI30- ja EI60-rakenteet.
- Sisäverhous: kipsilevy, kipsilevy x 2, vaakakoolaus ja kipsilevy, ulkoverhous (katkoviiva), koolaus ja ulkoverhous, ilman sisäverhousta
- Sisäkatto: koolaus ja kipsilevy, harvalaudoitus, ilman sisäkattomateriaalia, EI30 (eristetty), EI30 (eristämätön)

Alkuperäisenä suunnitelmana oli parametrisen detaljin piirtäminen viivapiirtona yksittäisenä familynä. Detaljia tehtiinkin näin tiettyyn pisteeseen asti. Toiminnallisuutta saatiin luotua esimerkiksi rakennosien dimensioihin. Varhaisessa vaiheessa kuitenkin huomattiin, että tällä menetelmällä rakennusosien nimeäminen *tageillä* eli tunnisteilla Revitin projekttilassa ei ollut mahdollista. Jokainen rakennusosa olisi pitänyt nimetä erikseen tekstillä projekttilassa, mikä taas ei täytännyt parametrille detaljille asetettuja vaatimuksia. Lisäksi monimutkaisempaa toiminnallisuutta olisi ollut mahdotonta luoda pelkkien viivojen välille.

Suunnitelmaa muutettiin niin, että kaikista rakennusosista tehtiin oma *detail item* -family, jotka yhdistettiin lopussa yhdeksi parametriseksi detaljiksi. Lopullinen detalji-family on siis *host family*, johon *detail item* -familyt yhdistettiin *nested family*nä. *Detail item* -objektien lisäksi *host family*ssä käytettiin viivapiirtoa ja *filled regionia* (täyttökuvia). Suunnitelman muutos mahdollisti osittain rakennusosien nimeämisen *tageillä* projekttilassa sekä monimutkaisemman toiminnallisuuden luomisen objektien välille. Tämä menetelmä on lähellä Revitin älykästä detaljointia, jossa detalji tehdään projekttilassa yhdistelemällä eri *detail items* -objekteja tietomallista otettuun leikkauskuvaan. Parametrisessä detaljissa on kuitenkin poikkeavaa se, että kuva tehdään *Family Editor* -tilassa eikä kuva ole mitenkään yhteydessä tietomalliin. Lisäksi objekteihin ja niiden välille pystytään lisäämään parametrejä.

Työvaiheen yksinkertaistamiseksi detaljin rakentaminen päätettiin tehdä rakenneosia kerrallaan. Detalji jaettiin eri rakenneosiin seuraavasti: Vesikatto- ja räystäsrakenne, yläpohja- ja sisäkatto-rakenne, seinärakenne ja sisäverhous. Tästä jaottelusta tehtiin vielä tarkempi jaottelu, jossa määritettiin, mitä erilaisia rakenneosien ratkaisuja detaljiin haluttiin sisällyttää (taulukko 8). Eri rakenneosien haluttiin toimivan itsenäisinä osinaan, jolloin detaljissa voisi vapaasti valita erilaisten rakenneosien yhdistelmiä. Rakenneosien tyypeiksi valittiin Autocad-detaljikirjaston detaljeissa esiintyvät rakenteet.

Taulukko 8 Rakenneosien eri tyypit detaljissa

Vesikatto	YP	Ulkoseinä	Sisäverhous
Rivipeltikate	Eristetty, <u>koolaus+kipsilevy</u>	Eristetty	Ei mitään
Tiilikuviopeltikate	Eristetty EI30	Eristämätön	Katkoviiva, (Ulkoverhous)
Betonitiilikate	Eristämätön, harvalaudoitus	EI30	Katkoviiva x 2, (<u>koolaus+ulkoverh.</u>)
Huopakate	Eristämätön, ilman harvalautaa	EI60	Kipsilevy
	Eristämätön EI30, harvalaudalla		Vaakakoolaus +kipsilevy

Kehittämisen alussa oli välttämätöntä opetella Revitin *Family Editorin* käytön perusteita. Ohjelman paremman ymmärtämisen perusteella voitiin detaljin rakennetta ja työvaiheita suunnitella tarkemmin. Detaljin toteuttaminen vaiheistettiin toisiaan loogisesti seuraavien vaiheiden mukaan seuraavasti:

1. Aloituspohjan luonti
2. *Detail item* -objektien luonti
3. *Host family reference planet* ja dimensioparametrit
4. *Nested familyiden* yhdistäminen *host familyyn*
5. *Visibility*-parametrien luonti
6. Eri osien toiminnan testaus ja muokkaaminen

Suunnittelu ja toteutusvaihe etenivät osittain päällekkäin. Opinnäytetyön tekijän kokemus familyn luomisesta oli suppea, joten ohjelman ja sen mahdollisuuksien opettelu ja testaus tapahtuivat kehittämisprosessin aikana. Myös detaljin toteutuksen suunnitelmat muokkautuivat työvaiheen aikana.

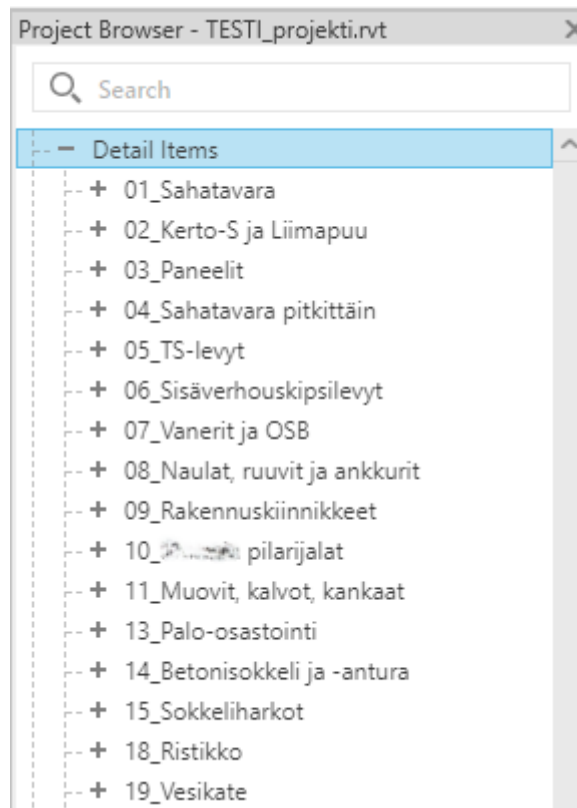
Parametristä detaljia käytetään mahdollisesti jatkossa osana yrityksen rakennesuunnitelmia. Tämän vuoksi opinnäytetyöntekijällä oli vastuu tehdä detalji niin, että se täyttää Maankäyttö- ja rakennuslaissa (132/1999) määritellyt rakentamista koskevat olennaiset tekniset vaatimukset. Detaljin kehittämisen jokaisessa vaiheessa oli otettava huomioon tekijät, jotka säätelevät rakenteiden lujuuutta ja vakautta, paloturvallisuutta, terveellisyttä, meluntorjuntaa sekä energiatehokkuutta. Rakennesuunnittelussa tavallisten detaljien rakenneratkaisut ovat muokattavissa vielä detaljiirrostosten teon hetkellä. Parametristä detaljia luotaessa rakenneratkaisut tulee olla etukäteen mietittynä, sillä niiden muokkaaminen jälkikäteen on hankalaa monimutkaisen toiminnallisuuden vuoksi. Parametrisen detaljin määräysten ja hyvän rakennustavan mukaisuus varmistettiin suunnittelemalla kuvan rakenneratkaisut teoriaosiossa käsiteltyjen vaatimusten mukaisesti.

5.2 Parametrisen päätyräystäsdetaljin toteutus

Parametrisen detalji-familyn rakentaminen aloitettiin muokkaamalla *Familytemplate* eli aloituspohja tarpeita vastaavaksi. Parametrinen detalji on *Detail item -family*, joka luodaan aina omalle *Detail item* -aloituspohjalle. Tämä aloituspohja sisältää sopivia työkaluja 2D-familyn luomiseen. Aloituspohja kustomoitiin lisäämällä siihen detaljin esitystapavaatimusten mukaiset viivatyyppit ja tarvittavat täyttökuviot. Samaa aloituspohjaa käytettiin sekä parametrinen detaljin että *nested familyiden* luomiseen.

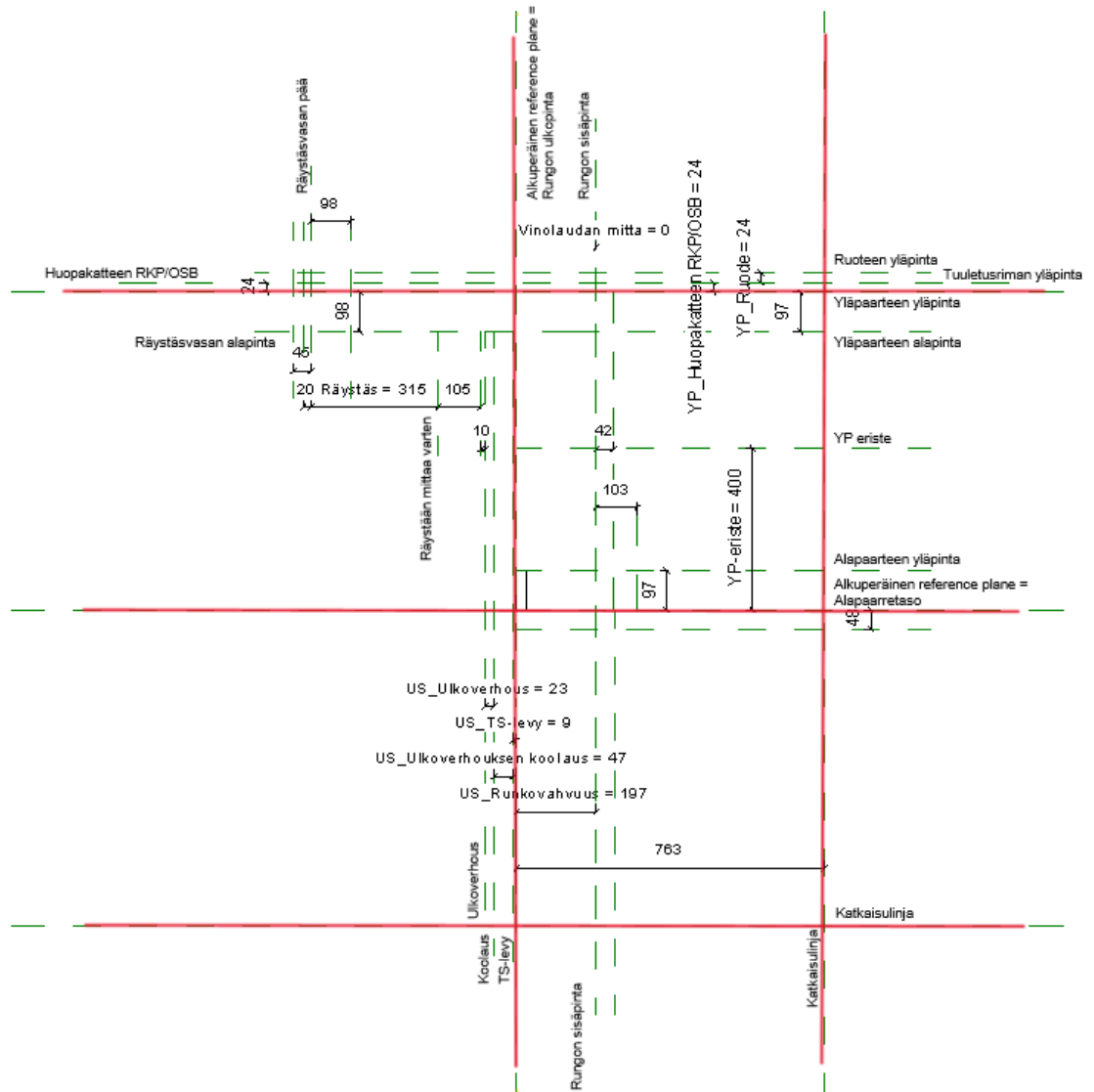
Parametrinen detalji toteutettiin *nesting family* -menetelmällä. *Nesting family* tarkoittaa toisen familyn lataamista ja sisällyttämistä toiseen *familyyn*. Ladattua *familyä* kutsutaan *nested familyksi*. Parametristä detalji-familyä, johon *nested family* ladataan, kutsutaan *host familyksi*. *Nested family* toimii itsenäisenä objektina *host familyn* sisällä. *Nested familyn* parametrit voidaan yhdistää *host familyn* parametreihin, mikä mahdollistaa *nested familyn* muokkaamisen *host familyn* parametreista projektitilassa. [17, s. 481.] Parametrisen detaljin rakentaminen aloitettiin tekemällä ensin sen rakennusosat eli *nested familyt*. Tässä vaiheessa *nested familyt* tehtiin sillä ajatuksella, että niitä voitaisiin käyttää myös erillisinä omina *Detail item* -objekteina projektitilassa. Myös viivana esiintyvät rakennusosat, kuten höyrynsulkumuovi tehtiin familynä. Tämä

mahdollistaa myös niiden objektien nimeämisen *tageillä* projekttilassa. *Detail item-familyt* jaettiin kategorioihin, jotka nimettiin ja numeroitiin kuvan 5 listauksen mukaisesti.



Kuva 5 *Detail item-familyt* projekttilassa

Family Editorissa 2D-familyn runko luodaan referenssitasoilla ja -viivoilla. Familyn geometriaa ohjaavat parametrit liitetään referenssitasoihin ja -viivoihin sujuvan toiminnan takaamiseksi. Referenssitasojen välille voidaan luoda ainoastaan lineaarisia parametrejä. Referenssiiviivaa käytetään luotaessa kulmiin ja suuntiin liittyviä parametrejä. [17, s. 463.] Parametriseen päätyräystäsdetaljiin luotiin runko referenssitasoilla, koska sen geometriaa ohjaavat parametrit olivat lineaarisia. Mallina taustalla käytettiin detaljikirjaston päätyräystäsdetaljia, jonka mukaisesti referenssitasot luotiin. Päätyräystäsdetaljiin ei sisältynyt kulmia tai suuntia määrittäviä parametrejä, joten referenssiivoja ei käytetty. Referenssitasojen välille määritettiin detaljin dimensioita ohjaavat parametrit. Kuvassa 6 on esitetty parametriseen detalji-familyn kaikki referenssitasot ja dimensioita ohjaavat parametrit. Punaisella merkityt referenssitasot ovat paikallaan pysyviä. Muut tasot liikkuvat määritettyjen dimensioparametrien mukaisesti. Parametrien toiminta testattiin jokaisen parametrin lisäämisen jälkeen.



Kuva 6 Parametrisen detaljin referenssitasot ja dimensioparametrit

Päätyräystäsdetalji-family rakennettiin Taulukko 8 jaottelun mukaisesti rakenneosia kerrallaan. Tarvittavat *nested family* ladattiin *host familyyn*. Työvaiheen alussa huomattiin, että *nested familyiden* kiinnittäminen toisiinsa tai tavalliseen viivaan toi esiin virhekommentin, jossa varoitettiin mahdollisista ongelmista familyn toiminnassa. Virheettömän toiminnan varmistamiseksi *nested family* kiinnitettiin ainoastaan referenssitasoihin, joiden toiminta oli varmistettu jo aikaisemmin. Päätyräystäsdetaljiin haluttiin sisällyttää useampia vaihtoehtoja esimerkiksi vesikatteelle, sisäverhoukselle ja sisäkatolle. Nämä rakenneosat lisättiin familyyyn samaan kohtaan päällekkäin, kiinnitettynä kuitenkin samoihin referenssitasoihin. Päällekkäisiin objekteihin lisättiin *visibility*-parametri säätämään niiden näkyvyyttä projekttilassa. Tämän vaiheen selkeyttämiseksi *visibility*-parametrit lisättiin objekteihin ennen niiden päällekkäin laittamista. Visibility-parametrillä ohjattavia päällekkäisiä objekteja olivat myös sisäverhouksen eri vaihtoehdot, ulkoseinän eristeet

muutos muokkaa kaikki tietomalliin lisätyt saman *family typen* objektit [17, s. 104]. *Type*-parametreiksi määritettiin sellaiset muuttujat, jotka ovat koko rakennuksessa muuttumattomat rakenneratkaisusta riippumatta, esimerkiksi seinän rakenteiden paksuus, vesikate ja räystäsmitta. Tarkoituksena on, että aluksi projekttilassa voitaisiin määrittää rakennuksen lähtötiedot *type*-parametreihin, minkä jälkeen siitä tehdään tarvittava määrä uusia *family typejä* (detaljeja), joihin muokataan rakenneratkaisut rakennuksen eri kohdissa *instance*-parametreillä. Tällöin rakenteiden lähtötiedot joudutaan lisäämään detaljiin vain yhdesti. Kuvassa 8 on esitetty päätyräystäsdetaljin *type*-parametrit. Kuvasta ilmenee myös, mitä toiminnallisuutta luotiin rakenteiden dimensioihin, vesikatemateriaalien vaihtoehtoihin ja räystään mitan määrittämiseksi. Räystäslaudat ovat vakiomittaisia ja räystään mitta määräytyy räystäslautojen lukumäärän mukaan.

Type Properties

Family: 1_Päätyräystäs Load...

Type: Classic-peltikate Duplicate... Rename...

Type Parameters

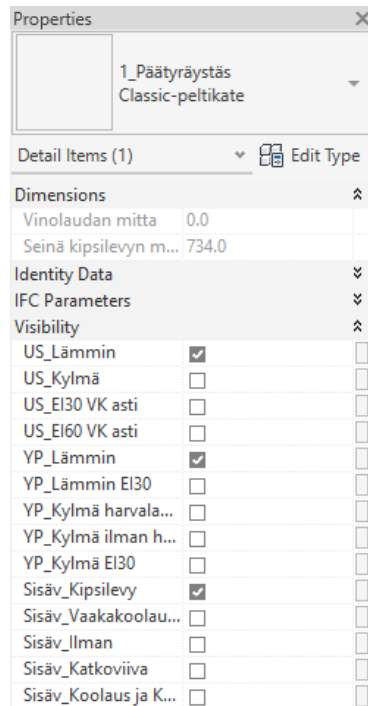
Parameter	Value
Graphics	
Sisäverhouskipsilevyt<Detail Items>	06_Sisäverhouskipsilevyt : Kipsilevy N
Structural	
Dimensions	
US_Ulkoverhous	23.0
US_Ulkoverhouksen koolaus	47.0
US_TS-levy	9.0
US_Runkovahvuus	197.0
YP-eriste	400.0
YP_Ruode	24.0
YP_Huopakatteen RKP/OSB	24.0
Räystäs	315.0
Identity Data	
IFC Parameters	
Visibility	
Vasan tukilauta	<input checked="" type="checkbox"/>
Classic	<input checked="" type="checkbox"/>
Tiilikainen	<input type="checkbox"/>
Tiilikate	<input type="checkbox"/>
Huopakate	<input type="checkbox"/>
Vesikattorakenteet Classic, Tiilikainen, Ti	<input checked="" type="checkbox"/>
Other	
Räystäslautojen lkm	4
Räystäsarrray	<input checked="" type="checkbox"/>
Räystäslaudat	4

Sort by: [Icons]

<< Preview OK Cancel Apply

Kuva 8 Päätyräystäs-familyn *type*-parametrit projekttilassa

Instance-parametrit muokkaavat ainoastaan sitä objektia, joka on valittuna tietomallissa [17, s. 104]. *Instance*-parametreiksi määritettiin erilaiset ulkoseinän, yläpohjan ja sisäverhouksen rakenneosien ratkaisut, jotka voivat vaihdella rakennuksen eri kohdissa. Kuvassa 9 on päätyräystäsdetaljin *instance*-parametrit. Kuvasta ilmenevät myös eri vaihtoehdot, joita rakenneosista on valittavissa.



Kuva 9 Päätyräystäsfamilyn *instance*-parametrit projekttilassa

Dimensioparametreilla luotiin toiminnallisuutta detaljin rakennekerrosten paksuuksiin. Mitat ovat muokattavissa *edit type* -valikon *dimensions*-osiossa (kuva 8). Dimensioparametrit ohjailevat familyn referenssitason etäisyyksiä suhteessa toisiinsa. Referenssitason kiinnitetty *nested family* tai viivapiirrolla tehdyt ääriviivat liikkuvat määritettyjen dimensioparametrien mukaan. Referenssitason kiinnitetystä *nested family*stä esimerkiksi seinärungon ja yläpohjan eristeet ja tuulensuojalevy olivat sellaisia, joiden haluttiin toimivan *host familyn* dimensioparametrien mukaan. *Host familyn* parametreihin linkitettävistä *nested family*istä tehtiin omat, erilliset familyt, jotta ne voitiin linkittää eri dimensioparametriin. Esimerkiksi *nested family*ksi luotiin US-eriste *family*, joka linkitettiin seinärungon dimensioparametriin. Vastaavasti tehtiin yläpohjan eristeelle ja tuulensuojalevylle. Parametrien linkittämisen myötä *nested family* toimii kiinteänä osana *host family*ä projekttilassa. *Nested familyiden* nimeäminen *tageilla* ei ollut enää mahdollista linkittämisen jälkeen.

Räystäälle saatiin rakennettua toiminnallisuutta, jolla pystyttiin määrittämään detaljiin räystäslautojen lukumääräksi 1 kpl tai enemmän. Räystään mitta määrittyy automaattisesti räystäslautojen lukumäärän perusteella. Räystäslaudat tehtiin *array*-työkalulla, jolla voidaan luoda yhdestä objektista lineaarisia jonoja. Räystäslauta-*arraysta* luotiin parametri, joka kontrolloi räystäslautojen lukumäärää. Ohjelma asettaa *Array*-jonossa olevien objektien sallituksi minimilukumääräksi 2 kpl, minkä vuoksi yhden räystäslaudan määrittäminen *arraylla* ei ollut mahdollista. Yhden laudan räystäas saatiin kuitenkin tehtyä laittamalla ylimääräinen räystäslautaobjekti *arrayn* ensimmäisen laudan kanssa päällekkäin. *Arrayn* objektien lukumäärää ja näkyvyyttä kontrolloimaan luotiin parametrin, joiden toimintaa ohjaamaan lisättiin Revit-*formulat*. Käytännössä räystäslaudat toimivat seuraavasti. Mikäli räystäslautojen määrä on 2 tai enemmän, on Räystäslauta-*array* näkyvässä ja lautojen lukumäärä määritetyn mukainen. Valittaessa räystäslautojen lukumääräksi 1, räystäslauta-*array* piiloutuu, mutta näkyville jää kuitenkin ylimääräinen räystäslautaobjekti, joka laitettiin *arrayn* ensimmäisen laudan kanssa päällekkäin. Räystään mittaa määrittää dimensioparametri, jonka toimintaa ohjaa räystäslautojen lukumäärän huomioon ottava Revit-*formula*. Vesikatto-rakenteet ja räystään otsalaudat on kiinnitetty referenssitasoniin, jotka liikkuvat räystään mittaa ohjaavan parametrin mukaisesti.

Detaljiin saatiin luotua yleisimmät detaljikirjaston ratkaisut eri rakenneosille eli vesikatolle, ulkoseinälle, yläpohjalle ja sisäverhoukselle. Kuvissa 8 ja 9 on esitetty nämä rakenteiden tyypit. Detaljissa on mahdollista tehdä erilaisia yhdistelmiä näistä rakenteista. Osat rakennettiin niin, että ne toimivat itsenäisesti eikä niillä ole riippuvuuksia toistensa kanssa. Detaljiin valitaan yksi jokaisesta rakenneosasta (vesikatto, ulkoseinä, yläpohja ja sisäverhous). Rakenneosien toiminnallisuus toteutettiin käyttämällä *visibility*-parametrejä, jotka luotiin kahdessa vaiheessa. Kaikille objekteille tehtiin oma *visibility*-parametri. Seuraavaksi luotiin *visibility*-parametrit kuvassa 9 näkyville eri rakenneratkaisuille. Objektien näkyvyyttä kontrolloimaan lisättiin Revitin *and*- ja *or*-formuloita (kuva 10). Tämän menetelmän etuna oli, että yhtä objektia oli mahdollista käyttää useampaan eri rakenneratkaisuun.

VK kivivilla (default)	<input type="checkbox"/>	= and(or(Classic, Tiilikainen, Tiilikate), or(US_EI30 VK asti, US_EI60 VK asti))
Vesikattorakenteet Classic, Tiilikainen, Tiili	<input checked="" type="checkbox"/>	= or(Classic, Tiilikainen, Tiilikate)
Palo-osastointi (default)	<input type="checkbox"/>	= or(US_EI30 VK asti, US_EI60 VK asti)
48x97 ristikon kyljessä (default)	<input type="checkbox"/>	= and(or(US_EI30 VK asti, US_EI60 VK asti), or(YP_Kylmä harvalaudoitus, YP_Kylmä ilman harvalaudoitusta))
HS-muovi (default)	<input checked="" type="checkbox"/>	= or(YP_Lämmin, YP_Lämmin EI30)
Kipsilevy seinä (default)	<input checked="" type="checkbox"/>	= or(Sisäv_Kipsilevy, Sisäv_Koolaus ja Katkoviiva)
Kipsilevy seinä 2 (default)	<input type="checkbox"/>	= and(Sisäv_Kipsilevy, YP_Lämmin, YP_Lämmin EI30)
Jäykistävä vinolaudoitus (default)	<input type="checkbox"/>	= or(YP_Kylmä harvalaudoitus, YP_Kylmä ilman harvalaudoitusta)
YP_eriste (default)	<input checked="" type="checkbox"/>	= or(YP_Lämmin, YP_Lämmin EI30)
Kattokoolaus 47x47 (default)	<input checked="" type="checkbox"/>	= or(YP_Lämmin, YP_Lämmin EI30)
Kattokoolauksen välikapula 47x47 (default)	<input checked="" type="checkbox"/>	= and(or(YP_Lämmin, YP_Lämmin EI30), Sisäv_Kipsilevy)
Kattokoolauksen välikapula 48x97 (default)	<input type="checkbox"/>	= and(or(YP_Lämmin, YP_Lämmin EI30), Sisäv_Vaakakoolaus ja Kipsilevy)
Kipsilevy_katto (default)	<input checked="" type="checkbox"/>	= or(YP_Lämmin, YP_Lämmin EI30)
Kipsilevy_katto 2 (default)	<input checked="" type="checkbox"/>	= and(or(YP_Lämmin, YP_Lämmin EI30), Sisäv_Kipsilevy)
Kipsilevy_katto 3 (default)	<input type="checkbox"/>	= and(YP_Lämmin EI30, Sisäv_Kipsilevy)

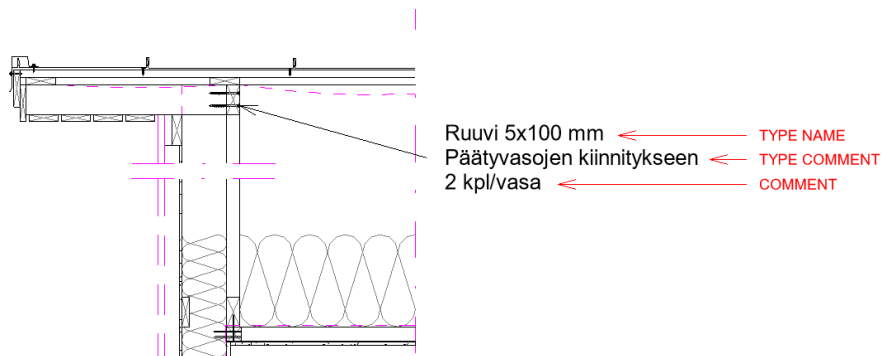
Kuva 10 Objektien *visibility*-parametrejä kontrolloivat Revit-*formulat*

Rakenneosien eri yhdistelmien toimivuus testattiin. Erityisesti kiinnitettiin huomiota rakenneosien liitoksiin ja rakennetekniseen toimintaan. Rakennepiirustuksissa esitetään mm. kiinnitykset, liitokset, reiät ja heikennykset, rakenteiden ja niiden materiaalien ominaisuudet kantavuuden, äänen-, lämmön-, kosteuden- ja vedeneristykseen, paloturvallisuuden ja säilyvyyden suhteen [6]. Eri yhdistelmien liitosten piti olla rakenneteknisesti oikein toteutettu, jotta kuvien sisältö olisi määräysten mukainen ja rakenneteknisesti toimiva. Palo-osastoiduissa rakenteissa huomioitiin liitoksissa katkeamaton rakenne sekä oikeanlainen palo-osastoiva rakenne. Rakenteiden lujuus ja vakaus huomioitiin alapaarretason jäykistykseen yhtenäisyydessä sekä kiinnikkeiden sijoittelussa. Seinärakenteessa huomioitiin myös höyrynsulkumuovin oikeanlainen sijoittaminen niin, ettei ole riskiä kosteuden tiivistymisestä seinärakenteen sisään. Ulkoseinärakenteen tulisi harventua sisältä ulospäin niin, että rakennekerrosten vesihöyrynläpäisevyys kasvaa ulospäin mentäessä [13, s. 399.]. Vaakakoolauksen ja lisäeristeen sisältämässä seinärakenteessa höyrynsulkumuovi on sijoitettu 97–122 mm:n rungolla vaakakoolauksen ja sisäverhouskipsilevyn väliin. 147 mm:n tai paksummalla rungolla höyrynsulkumuovi on sijoitettu rungon ja vaakakoolauksen väliin. Liitokset tehtiin käyttämällä dimensioparametrejä, *formuloita* sekä *visibility*-parametrejä.

5.4 Rakenteiden nimeäminen ja selitetekstit

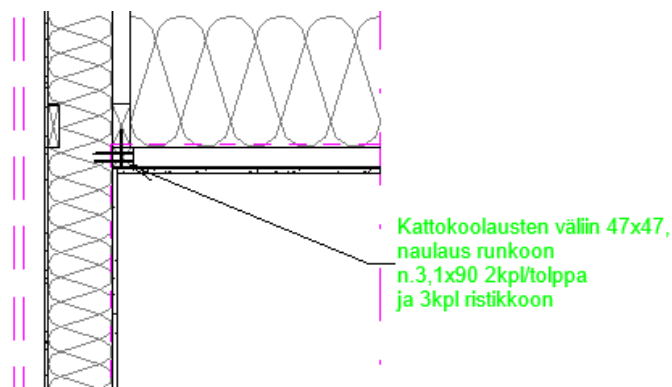
Projekttillassa detaljin rakenneosien nimeäminen voidaan tehdä *tageilla*, *keynoteilla* tai tekstillä. *Tag* on objektien nimeämiseen tarkoitettu merkintätyökalu. Jokaiselle Revitin objektille on olemassa oma *tag*. Projekttillassa *tagin* esittämä seliteteksti on määritetty valmiiksi *tag-familjissa*, jota voidaan muokata tarpeita vastaavaksi. Parametrinen detaljin *Detail item* -familjille tehtiin oma *tag*, jossa tekstikenttään nousee familjin *type name*, *type comments* ja *comments* (kuva 11). *Tagissa* *type name* on aina esillä. Tämän vuoksi objekteja luotaessa oli tärkeää nimetä *family*

type sellaiseksi, kuin sen halutaan detaljin *tag*issä esiintyvän. *Type comment* ja *comment*-kenttiin voidaan tarvittaessa kirjoittaa lisätietoja, jotka objektista halutaan esittää.



Kuva 11 *Detail item* -tag projekttilassa

Detaljikirjaston kuvissa on useita tarkentavia selitetekstejä rakenneosien teknisestä toteuttamistavasta. Pelkkiä *tagejä* käyttämällä tarkentavat tiedot pitäisi kirjoittaa aina erikseen *type comment*- tai *comment*-kenttään. Tämä taas ei ole tarkoituksenmukaista parametrisessä detaljissa, jonka käyttö pitäisi olla nopeaa ja vaivatonta. Parametrisen detaljin pidemmistä seliteteksteistä tehtiin tekstitiedosto, joka ladattiin Revitin *User keynoteseihin*. Tämä voidaan liittää mihin tahansa objektiin ja listalta valita haluttu seliteteksti (kuva 12).



Kuva 12 *User keynote* liitettynä objektiin

User keynote -listalla kaikki selitetekstit löytyvät valmiina oikein laadittuina, eikä detaljien seliteteksteihin tule tekijöiden välisiä eroavaisuuksia. Oikean selitetekstin löytämisen helpottamiseksi *keynotesit* voidaan jaotella listalle detaljikirjaston pääotsikointien mukaan. Esimerkiksi kaikki päätyräystään selitetekstit löytyisivät Päätyräystäs-otsikon alta (kuva 13).

Key Value	Keynote Text
PAÄTYRÄYSTÄS	PAÄTYRÄYSTÄSDETALJIN SELITETEKSTIT
... PÄÄTY-1	Päätyrunko ylös asti
... PÄÄTY-2	Päätyvasojen kiinnitys ruuvilla 5x100 mm 2 kpl/vasa
... PÄÄTY-3	32x100 päätyvasojen tueksi, kiinnitetään runkoon n. 3, 1x90...
... PÄÄTY-4	24x98 TS-levyn jatkoskohtaan
... PÄÄTY-5	Kattokoolausten väliin 47x47, naulaus runkoon n.3, 1x90 2k...
... PÄÄTY-6	48x97, n. 3, 1x90 2kpl/k200
... PÄÄTY-7	24x98 vinojäykiste alapaarteen yläpintaan ristikkojaolla, n. 2...

Kuva 13 Päätyräystä *User keynotes*

Parametrisen detaljin objektien nimeämisessä ja seliteteksteissä voidaan joissain tapauksissa käyttää myös tekstityökalua. Tavallisen tekstin käyttö tulee kysymykseen erikoistilanteissa, kun *tagissä* tai *keynoteseissa* ei ole sopivaa selitettä. Toimeksiantajayrityksen mahdollinen tuleva Revit-detaljikirjasto voisi sisältää detaljeja, joiden selitteet on tehty tagejä, *keynoteseja* ja tekstiä yhdistelemällä.

5.5 Ohjelman asettamat rajoitukset

Kehittämisen eri vaiheissa tuli vastaan ohjelman asettamia rajoituksia ja haasteita, jotka koskivat pääasiassa parametrisen detaljin ulkoasua ja käyttöä projekttilassa. Haasteita aiheutui joidenkin objektien esiintymisestä projekttilassa, *tagien* ja *keynotesien* ulkoasussa, objektien *tageissä* sekä *visibility*-parametrien valikon toiminnassa. Osaltaan haasteet olivat ratkaistavissa, mutta kompromissejakin jouduttiin tekemään.

Tagin lisääminen detaljin objektiin edellyttää, että se toimii omana erillisenä objektina projekttilassa. *Host familyn* parametreihin yhdistetty *nested family* toimii *host familyn* kiinteänä osana. Tämän vuoksi yhdistettyä *nested familyä* ei pysty nimeämään *tagillä* projekttilassa. Parametrisessä detaljissa esimerkiksi tuulensuojalevy, vesikatteet ja eristeet ovat objekteja, joita ei pysty nimeämään *tagillä*. Ongelmaan ei löytynyt muuta ratkaisua, kuin *keynotesin* tai tekstin käyttäminen tarvittaessa. Näiden objektien nimeämättä jättäminen on myös hyväksyttävää, koska näille rakenteille ei detaljikirjaston kuvissakaan ole selitetekstiä.

Toinen ongelma koski selitetekstien ulkoasua projekttilassa. Selitetekstit tehdään *tagiin type commentina* tai *commentina* taikka erillisenä *keynotesina*. Molemmissa vaihtoehdoissa tekstin rivittäminen oli ongelmana. Kuvassa 11 esitetyt *type comment*- ja *comment*-tekstit tulivat kaikki yhdelle riville, jolloin detaljin vaatima tila kasvaa liian isoksi. *Type commentin* ja *commentin* osalta

ongelmaan ei löytynyt ratkaisua. *Keynotes*-tekstien rivitys (kuva 12) on kuitenkin mahdollista muokkaamalla tekstitiedosto Excel-ohjelmalla, jossa teksti voidaan rivittää näppäinyhdistelmällä Alt+Enter. Muokattu Excel-tiedosto tallennetaan *Keynotes*-tekstitiedostoksi muotoon 'Muotoiltu teksti (välilyöntierotin) (*.prn)' ja ladataan Revitin *User keynotesiksi*.

Vesikatteelle, ulkoseinälle, yläpohjalle ja sisäverhoukselle on kullekin neljä erilaista vaihtoehtoa, joista täytyy aina olla yksi valittuna. Eri rakenneosien *visibility*-parametreihin yritettiin Revit-*formuloilla* saada aikaan käskyä, jolla saman ryhmän rakenneosien valintaruuduissa voisi olla vain yksi valinta kerrallaan. Esimerkiksi valittaessa Classic-peltikate, muiden vesikatteiden valintaruudut eivät voisi olla valittuna. Tällä olisi pystytty estämään kahden objektin päällekkäinen valinta. *Visibility*-parametrin kontrollointi kaavalla ei kuitenkaan onnistunut, koska samat parametrit olivat jo yhdistettynä toiseen kaavaan. Tämän ongelman kanssa jouduttiin tekemään kompromissi. Valintaruutujen oikeanlainen käyttö jätettiin jatkokäyttäjän vastuulle. Jatkokäyttäjän vastuulla on myös oikeanlaisten rakenneyhdistelmien valinta palo-osastoidulla EI30 tai EI60 seinällä. Detaljiin on rakennettumahdollisuus yhdistellä US-rakenteesta ja sisäverhouksesta palo-osastoiva rakenne. Suunnittelijan vastuulle kuitenkin jää oikeanlaisen yhdistelmän valitseminen.

Kehittämisen prosessin aikana huomattiin, että detaljin toiminta projektitilassa hidastui toiminnallisuuden lisääntymisen myötä. Monimutkaisen toiminnallisuuden vuoksi muutosten latausaika kestää useita sekunteja, varsinkin jos tehdään useita muutoksia kerralla. Tämän ei koettu olevan ongelma, sillä latausajasta huolimatta detalji muokkautui syötettyjen parametrien mukaisesti.

5.6 Toimeksiantajayrityksen rooli kehitystyössä

Kehitystyössä mukana oli toimeksiantajayrityksen suunnittelupäällikkö, jolta idea projektiin oli lähtenyt. Opinnäytetyön suunnitteluvaiheessa käytiin läpi ohjelman eri toimintoja ja mahdollisuuksia. Suunnittelupäällikkö esitteli ajatuksen parametrisestä detaljista ja esimerkkejä sen mahdollisista toiminnoista. Toimeksiantajayrityksen puolelta ei annettu valmiita vaatimuksia detaljin sisällöstä eikä työskentelymenetelmistä. Opinnäytetyöntekijällä oli prosessin aikana vapaat kädet viedä kehitystyötä eteenpäin haluamaansa suuntaan detaljin sisällön sekä työskentelymenetelmien suhteen. Toimeksiantajayrityksen rooli prosessissa oli toimia teknisenä tukena sekä antaa erilaisia näkökulmia ohjelman käytöstä sekä parametrisestä detaljista. Prosessin aikana käytiin pariin otteeseen läpi kehitystyön etenemistä ja haettiin yhdessä ratkaisuja ilmenneisiin ongelmiin.

6 Lopputulokset ja johtopäätökset

Opinnäytetyön tavoitteina oli luoda parametrinen detalji käyttäen Autodeskin Revit-ohjelmaa sekä selvittää kehitysprosessin vaiheet jatkokäyttö varten. Työn lopputuloksena saatiin aikaan parametrinen päätyräystäsdetalji, johon onnistuttiin sisällyttämään monipuolisesti suunniteltua toiminnallisuutta. Työn aikana kertyi tietoa parametrisen detaljin kehitysprosessin vaiheista ja sopivista menetelmistä. Saadun tiedon perusteella voidaan parametrisen detaljin kehitysprosessi toistaa tarvittaessa muidenkin detaljien kanssa.

6.1 Kehittämisprosessi

Parametrisen detaljin toteutus on mahdollista. Avainasemassa on detaljin hyvä suunnittelu etukäteen, muuttujien järkevä määrä sekä tarpeeksi yksinkertainen toiminnallisuus. Detaljin suunnitteluvaiheessa kannattaa miettiä mahdollisimman pitkälle lopullisen detaljin toiminta, rakentamisen vaiheet, käytettävät työkalut, parametrit ja *formulat*.

Yhteen parametriseen detaljiin on järkevää yhdistää vain saman tyyppisiä rakenneosia, kuten esimerkiksi opinnäytetyössä käytetty ulkoseinärakenne. Vastaavasti yhteen parametriseen detaljiin voitaisiin yhdistää esimerkiksi erilaisia päätyräystäsdetaljeja katosrakenteella. Toimintojen ja vaihtoehtojen määrä kannattaa pitää maltillisena. Opinnäytetyössä rakenneosien (vesikate, yläpohja, ulkoseinä ja sisäverhoukset) erilaisten variaatioiden määräksi tuli 4–5 kpl. Tällä määrällä parametrit ja niiden koordinointi oli vielä työn tekijän hallittavissa. Lisäksi ongelmatilanteissa oli vielä mahdollista jäljittää virheen alkuperä. Detaljissa muutosten latausaika pitkittyi toiminnallisuuden lisääntyessä. Myös tämä seikka tukee toimintojen määrän pitämistä maltillisena. Detaljin muutosten latausajan piteneminen oli kuitenkin työn tekijän kokemus, mutta sitä ei todennettu mitenkään. Työssä ei kokeiltu miten detalji käyttäytyy, jos siihen lisätään suurempi määrä toiminnallisuutta.

Detaljin rakentaminen onnistui hyvin *Family Editorin* perustyökaluilla, joita oli tarjolla *Detail item*-aloituspohjassa. Toiminnallisuus rakentui kolmella erilaisella parametrilla: *dimension*, *visibility* ja *array*. Parametrien toimintaa kontrolloitiin *if*-, *and*- ja *or*-*formuloilla*. Lisäksi joitakin parametrejä kontrolloitiin yksinkertaisilla matemaattisilla laskukaavoilla. Detaljin kehitysprosessi oli selkeästi jaettavissa työvaiheisiin:

1. *Detail itemsien/nested familyjen luominen*
2. Parametrisen detaljin/*host familyn* referenssitason luominen
3. Dimensioparametrien lisääminen
4. *Nested familyjen* lisääminen *host familyyn*
5. *Array-* ja *visibility-*parametrien lisääminen
6. *Revit-formuloiden* lisääminen parametreihin
7. *Tagien* ja *keynotesien* luominen

6.2 Revitin ja käyttäjän ominaisuudet prosessissa

Revitin toiminnassa detaljin luojaan kannattaa pitää mielessään, että ohjelman toiminta on loogista ja johdonmukaista. Poikkeuksia toiminnassa ei tapahdu, vaan ohjelman toiminta on aina samanlaista. Kehitysprosessin aikana ohjelma antoi virhe- ja varoitusilmoituksia. Ohjelman tehtävä on koordinoita tietomallissa tapahtuvia muutoksia. Kehitysprosessin aikana tulleiden virheilmoitusten tarkoitus on suojata tietomallin parametristä toimintaa. Ilmoituksilla Revit ohjaa käyttäjää lainalaisuuksiensa mukaiseen työskentelyyn. *Family Editorilla* työskennellessä on siis huomioitava, että virheilmoitukset johtuvat käyttäjästä eikä ohjelmasta. Ohjelma toimii käyttäjän ohjelmoimalla tavalla. Virheilmoituksia kohdatessaan käyttäjän tulee perehtyä aiheeseen ja etsiä vaihtoehtoja lähestymistapaa ongelmaan.

Ohjelman käyttäjän kokemuksella voi olla vaikutusta myös parametrisen detaljin luomisessa ja lopputuloksessa. Kokeneella ohjelman käyttäjällä on monipuolisemmin ratkaisuja ongelmiin, minkä myötä myös detaljissa käytetyt ratkaisut ja toiminnot voivat olla kehittyneemmät. Laadukkaan parametrisen detaljin aikaansaaminen vaatii tekijältä perehtyneisyyttä ohjelmaan, *Family Editoriin* ja familyiden kanssa työskentelyyn. Toistuva kehitysprosessin läpikäynti auttaa tekijää löytämään uusia ratkaisuja toiminnallisuuteen. Opinnäytetyön tekijän kokemus familyiden ja *Family Editorin* kanssa työskentelystä oli suppea. Tämän vuoksi parametrisen detaljin luomiseen käytetyt menetelmät olivat yksinkertaisia eikä moninaisempaa lopputulosta lähdetty tavoittelemaan.

6.3 Käyttömahdollisuudet

Parametrisen detaljin etuja ovat nopea ja helppo muokattavuus sekä parametrinen tarkkuus. Parametrinen detalji sopii parhaiten käyttöön vakioitujen rakenneratkaisujen kanssa, joissa samantyyppiset ratkaisut toistuvat projektista toiseen. Esimerkkinä voidaan käyttää toimeksiantajayrityksen asiakasyrityksen vakioituja rakenteita, joihin harvoin tehdään muutoksia. Parametrinen detalji siis kohdennetaan johonkin tietynlaiseen käyttöön. Sitä voidaan toki käyttää yleisdetaljinakin moninaisemmissa kohteissa, mutta siinä tapauksessa toiminnallisuutta todennäköisesti pitää lisätä verrattuna opinnäytetyössä tehtyyn detaljiin.

Detaljia voidaan käyttää projekteissa yksitellen muokattuna tai detaljin pohjalta voidaan rakentaa Revit-detaljikirjasto, joihin muokataan erilaiset detaljit valmiiksi. Jälkimmäinen vaihtoehto ei juurikaan eroaisi AutoCAD-detaljikirjastosta. Etuna kuitenkin olisi se, että kaikki materiaali löytyisi valmiiksi Revitistä, jolla toimeksiantajayrityksen muut rakennekuvat tehdään. Parametrisen detaljin nopean ja helpon muokattavuuden vuoksi, kirjaston rakentaminen olisi nopeampaa kuin jokaisen detaljin piirtäminen erikseen. Detaljikirjastosta voitaisiin myös tehdä hieman suppeampi versio kuin AutoCAD-kirjastosta. Revit-kirjastoon voitaisiin esimerkiksi lisätä detaljipiirroksia, joihin olisi määritetty valmiiksi rakennepaksuudet, mutta käyttäjän valittavaksi jätettäisiin esimerkiksi vesikate ja räystäspituus. Tällä menetelmällä detaljikirjaston kuvien määrää saataisiin vähennettyä huomattavasti, mutta kaikkia detaljin parametreja ei tarvitsisi joka kerta muokata. Tällöin myöskään detaljien muokkaus esimerkiksi räystäsaluslautojen muuttuessa ei olisi niin työlästä, kuin AutoCAD-detaljikirjastossa.

Parametrisen detaljin käytössä ongelmana on rakenteiden ja eri variaatioiden rajallinen määrä. Toiminnallisuus ja rakenneratkaisut on rajoitettu tietynlaisiksi jo familyn tekovaiheessa. Detaljilla ei pysty tekemään poikkeavia rakenneratkaisuja, vaan se toimii vain familyn ohjelmoidulla tavalla. Muokkaaminen family-tasolla ei ole kannattavaa, koska monimutkainen toiminnallisuus saattaa häiriintyä. Parametrisistä detaljeista poikkeavat detaljit joudutaan siis edelleen tekemään AutoCADilla tai Revitin muilla detaljointimenetelmillä. Jonkinlaista poikkeavuutta parametriseen detaljiin saadaan tehtyä yhdistämällä parametriseen detaljiin *detail itemseja* projektitilassa.

Detaljin kehittäminen on aikaa vievä ja monimutkainen prosessi, joka kuitenkin helpottuu työn tekijän kokemuksen karttuessa. Toistamalla prosessia detaljin luomiseen käytettävistä menetelmistä tulee rutiinia ja tiedon karttuessa löydetään uusia tapoja rakentaa toiminnallisuutta. Kokemuksen karttuessa myös detaljin luomiseen käytetty aika lyhenee. Kuitenkaan parametrisen detaljin ajankäyttöä etuja verrattuna AutoCAD-detaljeihin ei pystytä tämän opinnäytetyön

perusteella todentamaan. Prosessia pitäisi tarkastella laajemmassa mittakaavassa eikä vain detaljin luomiseen käytetyn ajan perusteella.

Parametrinen detalji on tekijänsä näköinen ja ilmentää tekijänsä kokemusta ja ajatusmaailmaa. Detaljin lopputulos voi vaihdella tekijän mukaan, jolloin eri detaljien välillä voi olla laadullista vaihtelua. Tasalaatuisuuden vuoksi parametristen detaljien kehittäminen yrityksessä olisi järkevää osoittaa samalle henkilölle, joka myös tekisi tarvittaessa muokkaukset detaljeihin.

6.4 Jatkokehitysmahdollisuudet

Looginen jatkokehityskohde ovat muiden AutoCAD-detaljikirjaston rakenneosien parametrinen detaljien rakentaminen. Opinnäytetyössä tehtyä umpiräystäällistä päätyräystäsdetaljia voidaan käyttää pohjana muille päätyräystäsdetaljeille. Muiden rakennuksen osien detaljipiirustukset voidaan kehittää opinnäytetyössä käytetyillä menetelmillä. Jatkossa detaljia rakentaessa ei enää tarvitse määrittää kehitysprosessia, joten detaljin rakentamisessa voidaan keskittyä enemmän toiminnallisuuden kehittämiseen monipuolisemmaksi ja sujuvammaksi. Kokemuksen myötä voidaan kokeilla monimutkaisemman toiminnallisuuden lisäämistä detaljiin.

Opinnäytetyön kehitysprosessin aikana heräsi ajatuksia muidenkin detaljien toiminnallisuuden luomisesta. Esimerkiksi parametrinen sivuräystäsdetaljin kehittäminen vaatii vielä lisätutkimuksia säädettävän kattokulman rakentamisen osalta. Kattokulman lisäksi detaljiin pitäisi saada myös mahdollisuus räystäsmitan säätöön. Molempien parametrien luominen yksinään onnistuu teoriassa, mutta näiden yhdistäminen voi tuottaa hankaluuksia käytännössä. Perustusleikkauksien osalta ajatuksia heräsi korkotasojen ja -merkintöjen automatisoinnista. Perustusleikkaukseen voitaisiin yrittää rakentaa lattian sekä sokkelin ylä- ja alapinnan korkoa määrittävä parametri sekä parametrin mukana muokkautuva korkomerkintä.

Detaljikirjaston muotoa miettiessä heräsi myös ajatus *Project parametersien* (Projektiparametrit) käytöstä familyissä. Ajatuksena olisi tehdä projektitemplateen projektiparametrit, jotka määritettäisiin aina projektin alussa. Näihin parametreihin voisivat kuulua esimerkiksi kaikki rakenteiden mittoja määrittävät parametrit. Projektiparametrit voitaisiin linkittää detalji-familyn parametrien kanssa, jolloin projektin alussa määritetyt parametrit muokkaisivat valmiiksi detaljeihin oikeat rakennepaksuudet. Tämä idea on vain teoria, jonka toteuttamiskelpoisuus vaatisi lisäselvittelyitä. Tässä työssä ei tutkittu onko Revitissä ylipäättään mahdollisuutta työstää projektiparametrejä *Family Editorissa*.

7 Pohdinta

Opinnäytetyö osoittaa, että ideoiden pohjalta voi syntyä jotain uutta ja toimivaa. Omia rutinoituneita työskentelytapojaan kannattaa tarkastella, jolloin vaihtoehtoisia toimintatapoja voi löytyä myös jokapäiväisistä työmenetelmistä.

Opinnäytetyön tuloksena saatiin aikaan uusi työskentelymenetelmä, jota ei aikaisemmin ole käytetty. Aiheeseen tutustumisen yhteydessä ei löytynyt tietoa vastaavasta kehitystyöstä. Aiheen tietopohja jouduttiin rakentamaan yhdistelemällä detaljeihin ja käytettyyn ohjelmaan liittyvää teoriatietoa. Tiedonhaun yhteydessä ei löytynyt myöskään suoraan kehitysprosessia tukevaa tietoa, vaan ohjelman käyttöominaisuuksia jouduttiin soveltamaan sillä ajatuksella, mitä parametri- seltä detaljilta tavoiteltiin. Mielestäni tiedon soveltaminen käytäntöön kehitysprosessissa onnistui hyvin, sillä detalji toimii lähes tavoitellulla tavalla.

Työn kehitysprosessin vaiheiden kuvauksessa ja lopputuloksissa ei ole itsessään tulkinnan varaa. Lopputulokset ja johtopäätökset ovat tämän opinnäytetyön kehitysprosessin mukaiset. Olen kuitenkin miettinyt, olisiko työ toisen tekemänä antanut erilaiset lopputulokset. Olisiko kehitysprosessi ollut erilainen jonkun muun tekemänä? Olisivatko käytetyt menetelmät ja työvaiheet olleet erilaisia? Työn tekijän kokemus ohjelman käytöstä voi vaikuttaa työn lopputulokseen. Kokeneemman Revitin käyttäjän luoma detalji ja kehitysprosessi olisi voinut olla erilainen, kuin nyt saavutettu. Itse työn tekijänä en luultavasti olisi esimerkiksi vuosi sitten olisi saavuttanut samaa lopputulosta kuin nyt.

Opinnäytetyön prosessi kokonaisuudessaan oli minulle haastava. Aiheen rajaaminen suunnitteluvaiheesta lähtien tuotti vaikeuksia. Lähtökohtana oli kokonaisen Revit-detaljikirjaston rakentaminen. Työn tekijänä minulla ei alkuvaiheessa ollut käsitystä aiheen laajuudesta ja siihen sisältyvästä työmäärästä. Opinnäytetyön edetessä aihe rajautui koskemaan pelkästään parametrisen detaljin kehitysprosessia, joka oli laajuudeltaan sopiva opinnäytetyön aiheeksi. Opinnäytetyöhön käytetty aika oli normaalia pidempi, koska työvaiheen onnistunut suorittaminen vaati ohjelman opettelua. Prosessin alussa kokemuksen puutteen vuoksi kehitysprosessissa käytetyt menetelmät eivät johdaneet toivottuun lopputulokseen. Yrityksen ja erehdyksen kautta aloin ymmärtää ohjelmaa paremmin, jolloin myös käytetyt menetelmät tarkentuivat. Kehitysprosessi sisälsi *Family Editorilla* työskentelyä, josta minulla ei ollut aiempaa kokemusta tässä laajuudessa. Työn aikana taitoni ohjelman käyttäjänä kehittyivät ja opin ymmärtämään ohjelman ajatusmaailmaa paremmin, joten kehitysprosessin läpikäyminen kehitti työskentelytaitojani myös ammatillisesti.

Alkuperäisten tavoitteiden pienentymisen vuoksi työn hyöty toimeksiantajayritykselle on pienempi kuin alussa suunniteltiin. Vaikka alun perin tavoiteltiin kokonaisen detaljikirjaston rakentamista, ovat työllä saavutetut tulokset kuitenkin askel kohti oikeaa suuntaa yrityksen suunnitelmien eteenpäin viemisessä. Revit-detaljikirjaston prosessi on vielä erittäin keskeneräinen. Täydellisen parametrinen detaljikirjaston rakentaminen vaatii aikaa ja resursseja yritykseltä. Työn anti yritykselle on kuitenkin todennettu tieto siitä, että detaljien kehittäminen on mahdollista ja resurssien kohdentaminen tähän voi tuottaa tulosta tulevaisuudessa.

Lähteet

1. Insinööritoimisto Raksainssit Oy. [Internet]. [viitattu 15.1.2024]. Saatavilla: <https://www.raksainssit.fi/>
2. RT103396. Rakennuspiirustukset. Esitystapaohjeita. RT-kortisto. Rakennustieto Oy. 2021.
3. RIL 229-1-2020. Rakennesuunnittelun asiakirjaohje. Tekstiosa. Helsinki. Suomen Rakennusinsinöörien Liitto RIL ry. 2020.
4. L 132/1999. Maankäyttö- ja rakennuslaki. 1999. Saatavilla: <https://www.finlex.fi/fi/laki/ajantasa/1999/19990132#L17-2>
5. A 216/2015. Ympäristöministeriön asetus rakentamista koskevista suunnitelmista ja selvityksistä. 2015. Saatavilla: <https://www.finlex.fi/fi/laki/ajantasa/2015/20150216>
6. Ympäristöministeriön ohje rakentamista koskevista suunnitelmista ja selvityksistä. YM3/601/2015. [Internet]. [viitattu 21.1.2024]. Saatavilla: https://www.edilex.fi/data/rakentamismaaraykset/Ymparistoministerion_ohje_rakentamista_koskevista_suunnitelmista_ja_selvityksista.pdf
7. Rakennustuotteet. Ympäristöministeriö. [Internet]. [viitattu 15.2.2024]. Saatavilla: <https://ym.fi/rakennustuotteet>
8. Sahatavaran lujuuslajittelu ja CE-merkintä. Puuinfo. [Internet]. 2020. [viitattu 2.3.2024]. Saatavilla: <https://puuinfo.fi/puutieto/sahatavara-ja-sen-jalosteet/sahatavaran-lujuuslajittelu-2/>
9. Paloturvallinen puutalo. Asuin- ja toimitilarakentaminen. Puuinfo. [Internet]. 2011. [viitattu 26.3.2024]. Saatavilla: https://puuinfo.fi/wp-content/uploads/2021/05/Palo_kirja_netki_kokonainen.pdf
10. A 848/2017. Ympäristöministeriön asetus rakennusten paloturvallisuudesta. 2017. Saatavilla: <https://finlex.fi/fi/laki/ajantasa/2017/20170848>
11. Paloturvallisuus. Puuinfo. [Internet]. 2020. [viitattu 27.3.2024]. Saatavilla: <https://puuinfo.fi/suunnittelu/maaraykset/paloturvallisuus/>
12. A 782/2017. Ympäristöministeriön asetus rakennusten kosteusteknisestä toimivuudesta. Saatavilla: <https://www.finlex.fi/fi/laki/alkup/2017/20170782>
13. Vinha J. Rakennusten rakennusfysikaalisen suunnittelun ja rakentamisen periaatteet. Rakentajan kalenteri 2009. Rakennustieto Oy. Saatavilla: https://tiedostot.rakennustieto.fi/rakentajain-kalenteri/RK090302.pdf?_gl=1*1106cca*_ga_QJFJQSBJM0*MTcyNjUwOTk1Ni4xLjEuMTcyNjUwMDAzNC42MC4wLjA.
14. RIL 250-2020. Kosteudenhallinta ja homevaurioiden estäminen. Helsinki. Suomen rakennusinsinöörien liitto RIL ry. 2020.

15. Autodesk Revit 2024. Autodesk. [Internet]. 2024. [viitattu 28.4.2024]. Saatavilla: <https://help.autodesk.com/view/RVT/2024/ENU/>
16. Tanska T, Österlund T. Algoritmit puurakenteissa: menetelmät, mahdollisuudet ja tuotanto. Oulu. Digiwoodlab; Oulun yliopisto, arkkitehtuurin tiedekunta. 2014. Saatavilla: <https://urn.fi/URN:ISBN:9789526204567>
17. Kirby L, Krygiel E, Kim M. 1. painos. Mastering Autodesk Revit 2018. Indianapolis, Indiana. Sybex. 2017.