Calculation and evaluation of U-value of construction by Add-in in Autodesk Revit

CZECH TECHNICAL UNIVERSITY IN PRAGUE
Faculty of civil engineering
Department of architecture engineering – K124

Bc. Horacio Kašpar

Master Thesis
Supervisors: Ing. Jakub Diviš, Ph.D., Ing. Jakub Veselka, Ph.D.

Acknowledgement

First, I want to thank my supervisors Ing. Jakub Diviš, Ph.D. and the biggest help Ing. Jakub Veselka, Ph.D., also I need to thank my friend Petr Sysel, for great coding advices. And for all my other friends who were helping me psychically through the process. Also, the book how to write scientific paper workingly called Jonah´s Milk. The last, but not least to my boss in work for giving me day off every time when needed.

Declaration

I honestly declare that the presented thesis is my own work and that I have cited every resource of information. And all used programmes were legal for my kind of use.

In Prague 8th January 2023: ___________________________

Annotation

The aim of this diploma thesis was to make Add-in into Autodesk Revit. U-Value in Revit is not calculated according to ČSN, add-in will improve accuracy of Revit´s calculations of U-value. Diploma thesis is divided into six parts. First part is about introduction of the plugin and about searching for if there is not a plugin that already can improve calculation quality into required accuracy. Second part compares calculations on the sample project according to ČSN with the Revit calculations. Third part is devoted to design architecture of the system of the plugin in visual programming environment Autodesk Dynamo and comparison of add-in calculations with ČSN. Fourth part contains creating add-in in programming language C# and comparison of ČSN and add-in results. Fifth part is summary of the project, results, benefits and outlooks how the plugin can be expanded in the future. In the attachment, there is my personal way to BIM and described how have I worked with artificial intelligence.

Keywords

Autodesk Revit, Thermal resistance during the heat transfer, thermal conductivity coefficient, Programming, Plugin, BIM

Anotace

Cílem této diplomové práce bylo vytvořit doplněk pro Autodesk Revit. Revit při výpočtech U-Value nezahrnuje vše, co ČSN vyžaduje. Cílem této práce bylo vytvořit doplněk, který bude tyto výpočty počítat přesně. Diplomová práce je rozdělena do šesti částí. První část obsahuje představení problematiky a rešerši, zdali již neexistuje doplněk, který by toto uměl. Druhá část srovnává na ukázkové studii výsledky z Revitu a manuální výpočty podle ČSN. Ve třetí části je představena architektura systému a popsáno tvoření pluginu ve vizuálním programovacím prostředí Autodesk Dynamo, následné porovnání výsledků s ČSN. Čtvrtá část obsahuje popis architektury systému, proces programování v klasickém programovacím jazyce C# a srovnání výsledků hotového pluginu s ČSN. Pátá část obsahuje shrnutí vytvořeného pluginu, jeho možných přínosů pro běžnou praxi a nápady, kam by se plugin mohl rozvíjet dále. V příloze je popsána má osobní motivace k této tématice a jak byla při programování využita umělá inteligence.

Klíčová slova

Autodesk Revit, Tepelný odpor při přestupu tepla, Součinitel tepelné vodivosti, Programování, Plugin, BIM

List of Abbreviations

EPS

– Expanded polystyrene – Thermal insulation, mostly used in the sandwich walls, floors and roofs.

R

– Thermal resistance – declares how effective is the material in resisting to transfer heat through conduction, radiation, and convention. The unit is (m^2*K)/W.

λ

– Thermal conductivity – The opposite to thermal resistance, declares how effective is the material in transferring heat through conduction, radiation, and convention. The unit is W/(m^2*K).

U-Value

– Declares how effective is the construction in transferring heat through conduction, radiation, and convention. The unit is W/(m^2*K).

R_{se}

– Thermal resistance of interior surface.

R_{si}

– Thermal resistance of exterior surface.

Integer

– Variable in C# stores integers.

Double

– Variable in C# stores numbers with 15 – 17 decimal places.

String

– Variable in C# stores text.

String builder

– Variable in C# stores multiple text lines

Foreach

– Function in C#, reaches each item in the list, do every desired action written in the curly brackets and then repeats for every item in the list.

Contents

1) Introduction

This part will be introduction of the solved part of this topic with context of standards. Second parts will be about aims of this thesis and third part will be research of current plugins.

1.1) Motivation

BIM = Building Information Modelling is basically making 3D models of the buildings instead of just drawing tons of lines in AutoCAD, which is similarly ineffective as if you were using the drawing-board.

Not just because it is faster, every change, like different size of the window is immediately in all drawings – floor plans, sections, views, so simply it is just far more effective.

But the BIM is not just about the drawings by itself. Important is also the “Information” part. Models also contains a lot of information, like materials from which are the constructions made and all their important properties like λ, or density etc.

All this information can be used in any way capable user wants to.

Why develop new add-in

When the model contains so much info, why not use them to calculate U-Values correctly.

How useful the BIM can be is limited only by developers and the things they made. With every app and add-in is the BIM more powerful and effective, which will attract more people to use it, it means more developers interested in developing BIM and so on.

1.2) Thesis statement.

In the Czech state standard ČSN 73 0540 [1] there is defined a property of a construction called heat transfer coefficient with more used equivalent technical term U-Value.

This parameter evaluates influence of whole construction, adjacent air layers on heat transfer trough construction.

Calculation formula for the U-Value.

$$U = \frac{1}{R_{si} + R + R_{se}}$$

Topic of how we get the R_{si} and R_{se} is complicated. For our needs all we need to know is that in ČSN 73 0540 [1] there is a table that includes exact values.

U – U-Value

R – Thermal resistance of construction.

R_{se} – Thermal resistance of interior surface.

R_{si} – Thermal resistance of exterior surface.

Θ_i – Temperature of interior environment.

Θ_{si} – Temperature of interior surface.

Θ_{se} – Temperature of exterior surface.

Θ_e – Temperature of exterior environment.

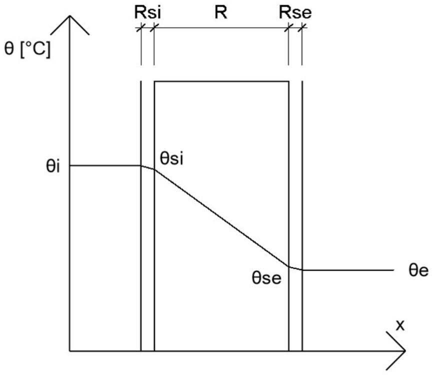
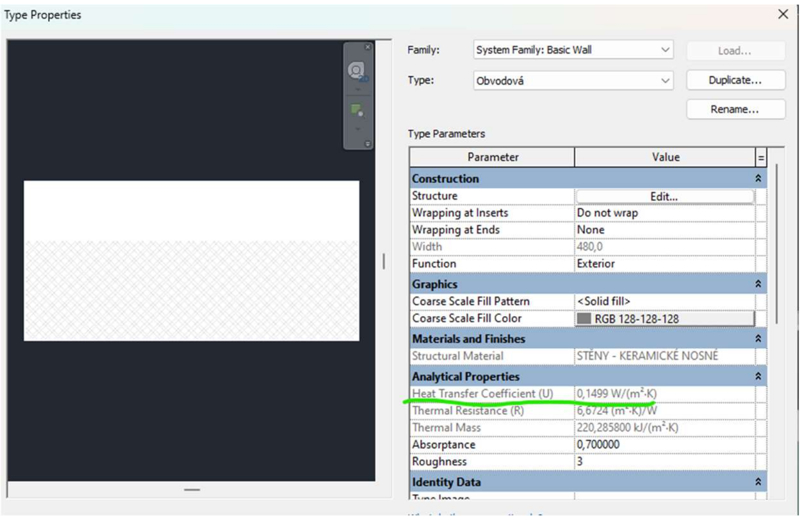


Figure 1 - Temperature course through the construction.



Family: System Family: Basic Wall

Type: Obvodová

Type Parameters

Parameter	Value
Construction	
Structure	Edit...
Wrapping at Inserts	Do not wrap
Wrapping at Ends	None
Width	480,0
Function	Exterior
Graphics	
Coarse Scale Fill Pattern	<Solid fill>
Coarse Scale Fill Color	RGB 128-128-128
Materials and Finishes	
Structural Material	STĚNY - KERAMICKÉ NOSNÉ
Analytical Properties	
Heat Transfer Coefficient (U)	0,1499 W/(m²·K)
Thermal Resistance (R)	6,6724 (m²·K)/W
Thermal Mass	220,285800 kJ/(m²·K)
Absorptance	0,700000
Roughness	3
Identity Data	
Type Image	

Figure 2 - U-Value in Revit.

In Revit, probably the most used BIM software there is a U-Value calculation, it calculates values only from material resistances and ignores R_{si} and R_{se}, which means that the U-Value is only indicative.

The calculations are more complex, but according to ČSN calculations should be made this way.

Calculations and comparison of exact values will be in the chapter 2, but Revit calculates with mistake, so when an engineer want to compare the U-Value to ČSN, Revit is unreliable.

1.3) Aims of the work

Aim of the coding was to make Revit add-in that will help engineers while designing compositions and make available for them to easily see U-Values calculated according to ČSN and compared to the standard.

Direct aims of the add-in:

- Add parameters R_{si} and R_{se} into the Revit.
- Insert values of R_{si}, R_{se} according to ČSN.
- Calculate U-Values of the elements.
- Compare U-Values to ČSN.
- Make weighted averages for the elements and for whole building cover.

1.4) Methods of the work

- Create exemplary model of fictive building with constructions.
- Calculate all constructions according to ČSN and compare them with Revit results.
- Make working script in visual programming tool Autodesk Dynamo, to make the system architecture and find potential problems.
- Make add-in to Revit in classic programming language C#.
- Test the add-in and declare how successful the work was.
- Find the possible ways to expand the add-in in the future.

1.5) Research – Tested plugins.

This part is dedicated to research of already existing plugins to find if there is not a plugin, that already can calculate what I have intended to make by myself. – Spoiler there isn´t, so if you don´t want to read about each plugin you can skip to overview at the end of chapter 1.

1.5.1) Methodology of chosen plugins.

First factor how I have chosen plugins to test was version compatibility, all of plugins were tested on Revit 2023 (EDU version, but this part is not important) so plugins had to be compatible with this version.

Also, I have tried to find plugins at least a bit connected to my topic so there are some eco-friendly calculators, or compositions and material libraries. But I didn´t want to test only the few ideas done other way, so I have tested a few different plugins.

The last factor was also that I needed the plugins to be free or have free trial.

1.5.2) Tally

Download source:

Official Autodesk plugin library:

https://apps.autodesk.com/RVT/en/Detail/Index?id=3841858388457011756&appLang=en&os=Win64&autostart=true

Or from a website of the plugin:

https://choosetally.com

Pricing:

Standard license $695.00/ year.

Premium license $995.00/ year.

On demand you can have free demo for non-commercial purposes.

Tested on 10 days long free trial.

Compatibility:

Revit 2016 and newer versions.

Tested version of the plugin: 2022.04.08.01

Description:

This plugin is used to see how the designed building stands in many different ecologic views. The plugin can consider the whole life of the building. Building phase – how ecologically objectional the materials are considering building phase and demolition phase in the future. Maintenance – for example the buildings are mostly designed to be used for 50 years, but the plasterboard is designed to be used just for 15 years. So, this app will consider, that this part must be three times changed during the life cycle of the building. Every part of the building has few graphs divided by on which aspect of environment is the calculation aimed. The plugin creates reports in formats PDF and XLSX (MS Excel). You can also make two variants and compare them.

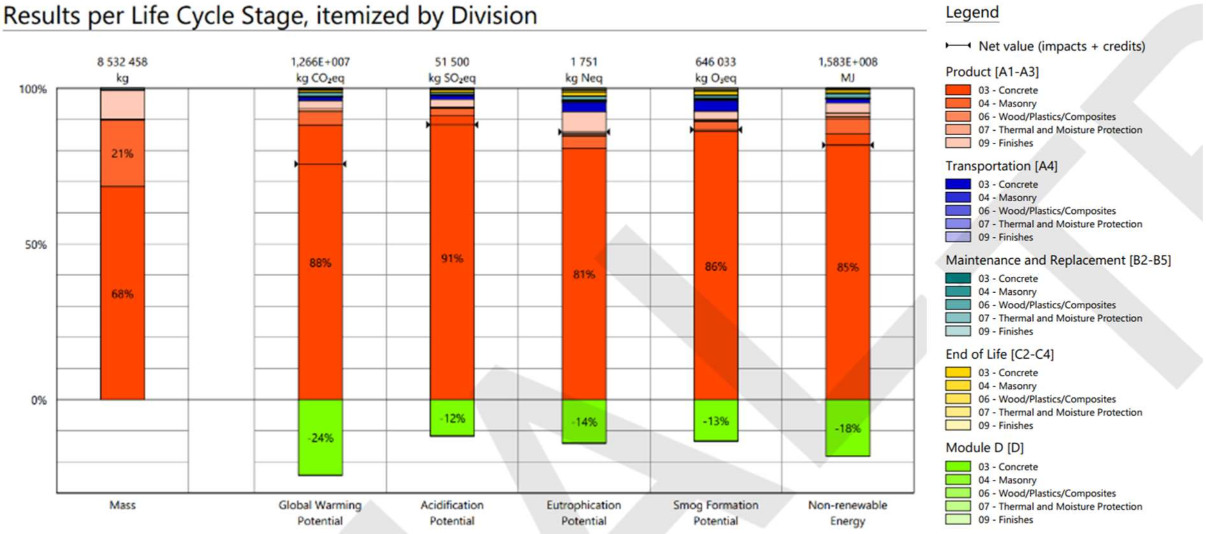
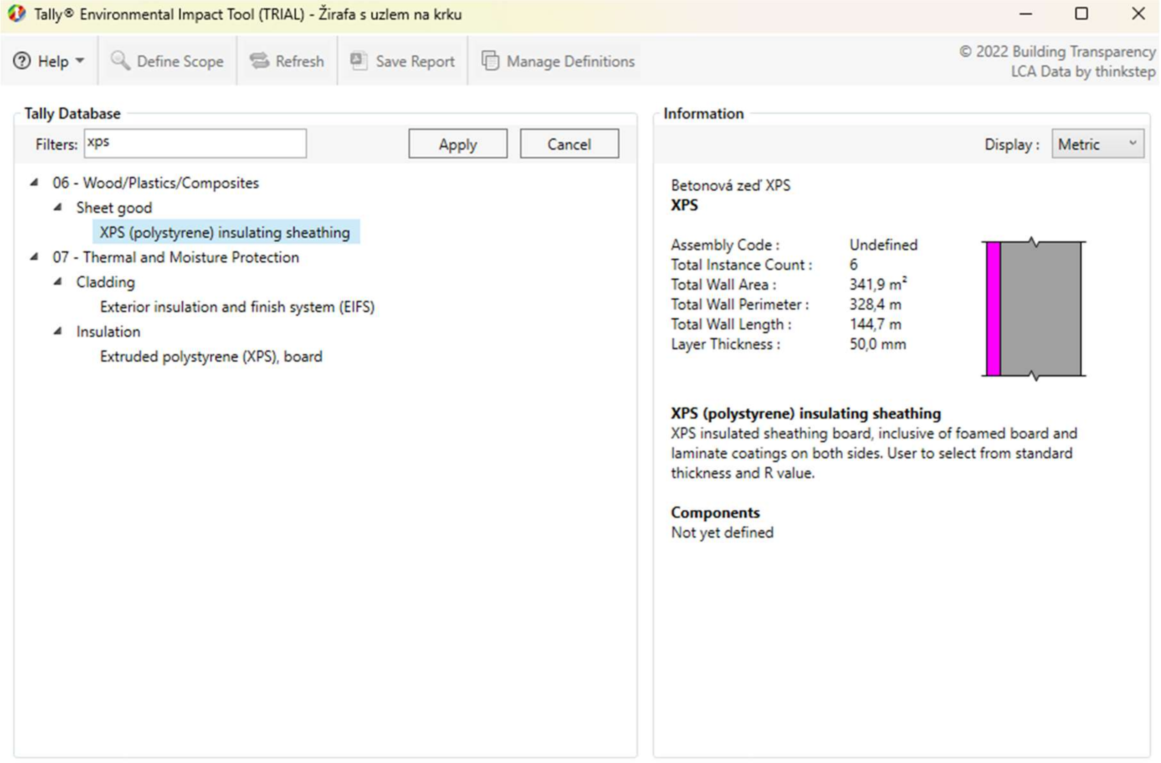


Figure 3 - Calculated graphs in PDF report and Legend.

Advantages or disadvantages:

+ Plugin is very complex, and it can help a lot with sustainability of construction.

Few other screenshots:



Tally® Environmental Impact Tool (TRIAL) - Žirafa s uzlem na krku

Help | Define Scope | Refresh | Save Report | Manage Definitions

© 2022 Building Transparency
LCA Data by thinkstep

Tally Database

Filters: xps | Apply | Cancel

- 06 - Wood/Plastics/Composites
  - Sheet good
    - XPS (polystyrene) insulating sheathing
- 07 - Thermal and Moisture Protection
  - Cladding
    - Exterior insulation and finish system (EIFS)
  - Insulation
    - Extruded polystyrene (XPS), board

Information

Display : Metric

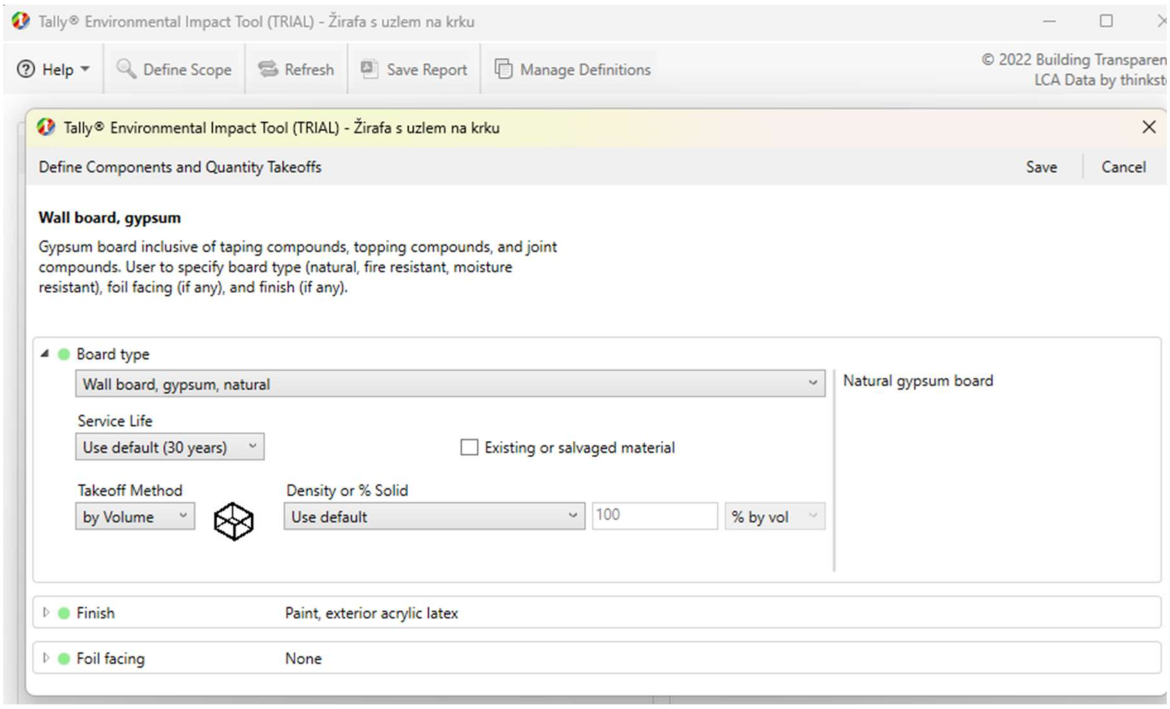
Betonová zeď XPS
XPS

Assembly Code :	Undefined
Total Instance Count :	6
Total Wall Area :	341,9 m^2
Total Wall Perimeter :	328,4 m
Total Wall Length :	144,7 m
Layer Thickness :	50,0 mm

XPS (polystyrene) insulating sheathing
XPS insulated sheathing board, inclusive of foamed board and laminate coatings on both sides. User to select from standard thickness and R value.

Components
Not yet defined

Figure 4 – Browsing the materials.



Tally® Environmental Impact Tool (TRIAL) - Žirafa s uzlem na krku

Help | Define Scope | Refresh | Save Report | Manage Definitions

© 2022 Building Transparen
LCA Data by thinkst

Tally® Environmental Impact Tool (TRIAL) - Žirafa s uzlem na krku

Define Components and Quantity Takeoffs | Save | Cancel

Wall board, gypsum

Gypsum board inclusive of taping compounds, topping compounds, and joint compounds. User to specify board type (natural, fire resistant, moisture resistant), foil facing (if any), and finish (if any).

● Board type

Wall board, gypsum, natural | Natural gypsum board

Service Life: Use default (30 years)

□ Existing or salvaged material

Takeoff Method: by Volume

Density or % Solid: Use default | 100 | % by vol

● Finish: Paint, exterior acrylic latex

● Foil facing: None

Figure 5 – Detailed options for the material.

<u>1.5.3) Carbo Life Calculator</u>

<u>Download source:</u>

Official Autodesk plugin library:

https://apps.autodesk.com/RVT/en/Detail/Index?id=5134144936403749174&appLang=en&os=Win64

Official website of the plugin:

http://www.davidveld.nl/carbocalc.php

<u>Pricing:</u>

Free

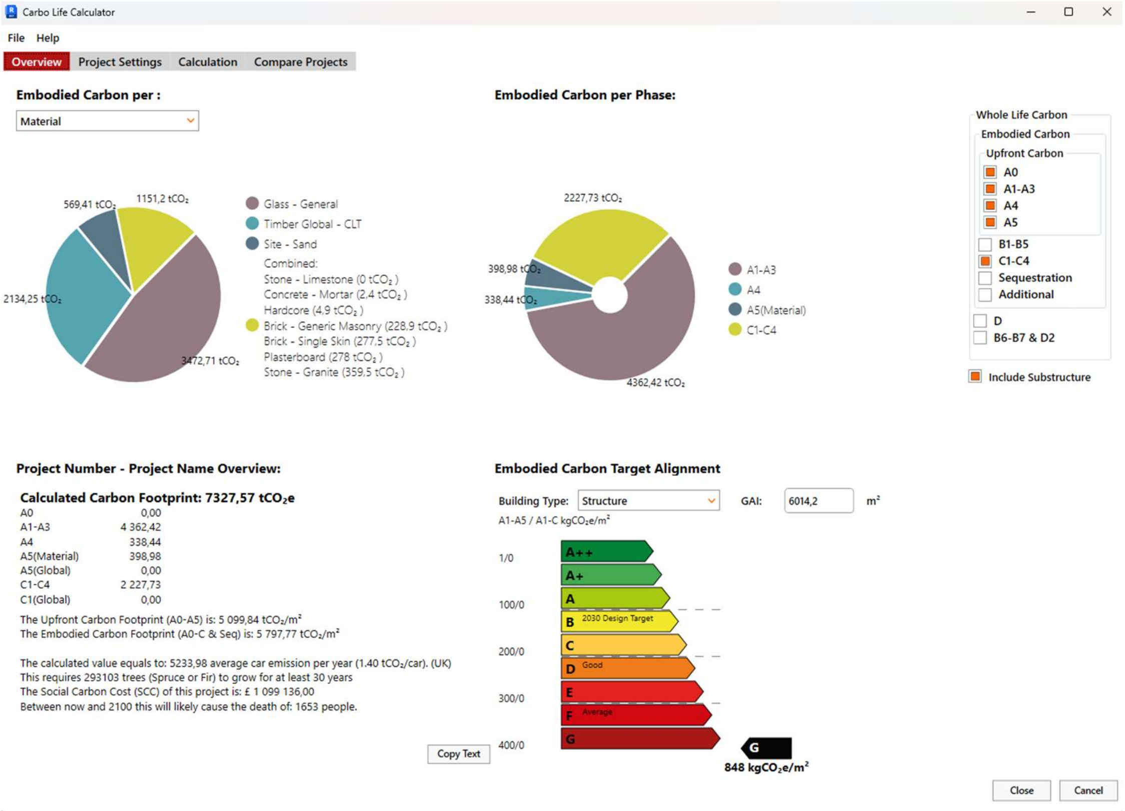
<u>Compatibility:</u>

Revit 2019 - 2024.

Tested version of the plugin: 1.5.0

<u>Description:</u>

Calculates embodied carbon in the building. You don´t need to assign any materials, just click to import settings, but if you want, you can adjust settings of materials as you want.



Embodied Carbon per : Material

Combined:
Stone - Limestone (0 tCO₂)
Concrete - Mortar (2,4 tCO₂)
Hardcore (4,9 tCO₂)
Brick - Single Skin (277,5 tCO₂)
Plasterboard (278 tCO₂)
Stone - Granite (359,5 tCO₂)

Whole Life Carbon
Embodied Carbon
Upfront Carbon
☒ A0
☒ A1-A3
☒ A4
☒ A5
☐ B1-B5
☒ C1-C4
☐ Sequestration
☐ Additional
☐ D
☐ B6-B7 & D2
☒ Include Substructure

Project Number - Project Name Overview:

Calculated Carbon Footprint: 7327,57 tCO₂e

A0	0,00
A1-A3	4 362,42
A4	338,44
A5(Material)	398,98
A5(Global)	0,00
C1-C4	2 227,73
C1(Global)	0,00

The Upfront Carbon Footprint (A0-A5) is: 5 099,84 tCO₂/m²
The Embodied Carbon Footprint (A0-C & Seq) is: 5 797,77 tCO₂/m²

The calculated value equals to: 5233,98 average car emission per year (1.40 tCO₂/car). (UK)
This requires 293103 trees (Spruce or Fir) to grow for at least 30 years
The Social Carbon Cost (SCC) of this project is: £ 1 099 136,00
Between now and 2100 this will likely cause the death of: 1653 people.

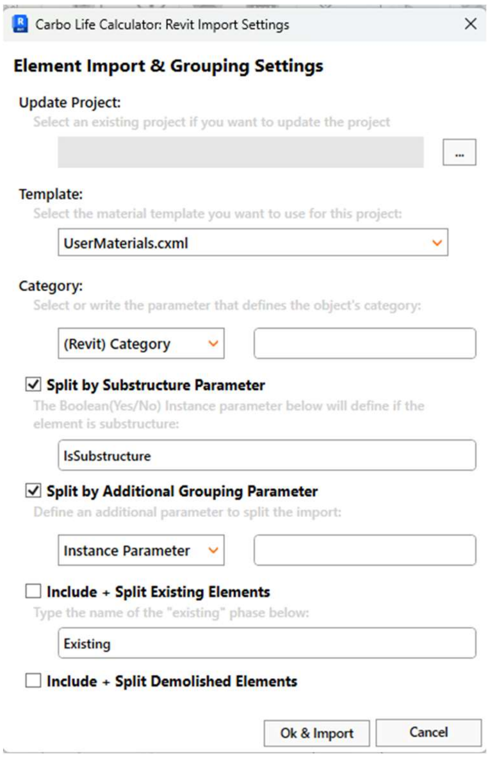
Building Type: Structure GAI: 6014,2 m²

Figure 6 – Embodied carbon by the building.

<u>Advantages or disadvantages:</u>

+ Simple and working.

Few screenshots:



Carbo Life Calculator: Revit Import Settings

Element Import & Grouping Settings

Update Project:
Select an existing project if you want to update the project

Template:
Select the material template you want to use for this project:
UserMaterials.cxml

Category:
Select or write the parameter that defines the object's category:
(Revit) Category

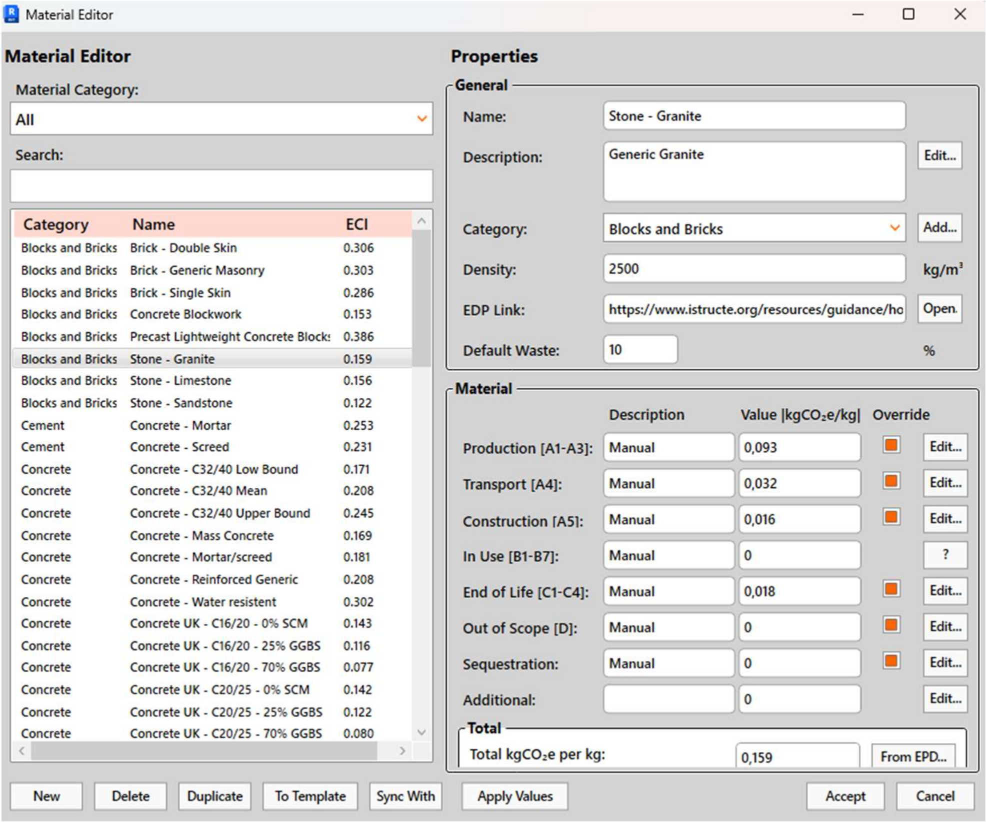
☑ Split by Substructure Parameter
The Boolean(Yes/No) Instance parameter below will define if the element is substructure:
IsSubstructure

☑ Split by Additional Grouping Parameter
Define an additional parameter to split the import:
Instance Parameter

□ Include + Split Existing Elements
Type the name of the "existing" phase below:
Existing

□ Include + Split Demolished Elements

Figure 7 - Importing settings.



Material Editor

Material Editor

Material Category: All

Search:

Category	Name	ECI
Blocks and Bricks	Brick - Double Skin	0.306
Blocks and Bricks	Brick - Generic Masonry	0.303
Blocks and Bricks	Brick - Single Skin	0.286
Blocks and Bricks	Concrete Blockwork	0.153
Blocks and Bricks	Precast Lightweight Concrete Block:	0.386
Blocks and Bricks	Stone - Granite	0.159
Blocks and Bricks	Stone - Limestone	0.156
Blocks and Bricks	Stone - Sandstone	0.122
Cement	Concrete - Mortar	0.253
Cement	Concrete - Screed	0.231
Concrete	Concrete - C32/40 Low Bound	0.171
Concrete	Concrete - C32/40 Mean	0.208
Concrete	Concrete - C32/40 Upper Bound	0.245
Concrete	Concrete - Mass Concrete	0.169
Concrete	Concrete - Mortar/screed	0.181
Concrete	Concrete - Reinforced Generic	0.208
Concrete	Concrete - Water resistent	0.302
Concrete	Concrete UK - C16/20 - 0% SCM	0.143
Concrete	Concrete UK - C16/20 - 25% GGBS	0.116
Concrete	Concrete UK - C16/20 - 70% GGBS	0.077
Concrete	Concrete UK - C20/25 - 0% SCM	0.142
Concrete	Concrete UK - C20/25 - 25% GGBS	0.122
Concrete	Concrete UK - C20/25 - 70% GGBS	0.080

Properties

General

Name: Stone - Granite
Description: Generic Granite
Category: Blocks and Bricks
Density: 2500 kg/m³
EDP Link: https://www.istructe.org/resources/guidance/ho
Default Waste: 10 %

Material

	Description	Value [kgCO₂e/kg]	Override	
Production [A1-A3]:	Manual	0,093	🟧	Edit...
Transport [A4]:	Manual	0,032	🟧	Edit...
Construction [A5]:	Manual	0,016	🟧	Edit...
In Use [B1-B7]:	Manual	0		?
End of Life [C1-C4]:	Manual	0,018	🟧	Edit...
Out of Scope [D]:	Manual	0	🟧	Edit...
Sequestration:	Manual	0	🟧	Edit...
Additional:		0		Edit...

Total

Total kgCO₂e per kg: 0,159

Figure 8 - Material editor.

1.5.4) BIMTech

Download source:

Official website of the plugin:

https://bimtech.cz/en/bimtech-tools-for-architects/

Pricing:

Plugin is for free.

Compatibility:

Revit 2017 - 2024.

Tested version of the plugin: 2.5.3.

Description:

As you can see in picture, there is many things that this plugin can do.

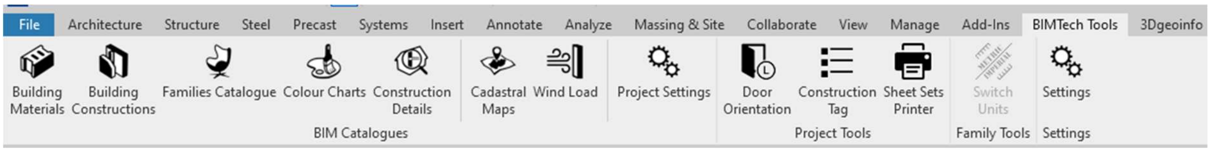
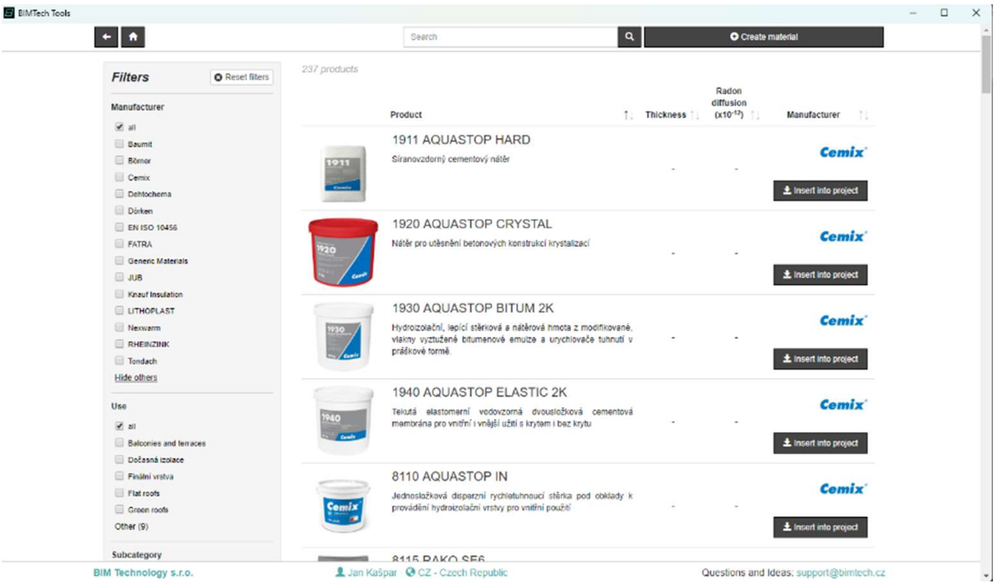


Figure 9 - BIMTech tools.

Part building materials is a huge library where you can pick a category of material you need, and then browse for the exact material you need, with all its parameters.



Manufacturer

☑ all
☐ Baumit
☐ Börner
☐ Cemix
☐ Dehtochema
☐ Dörken
☐ EN ISO 10456
☐ FATRA
☐ Generic Materials
☐ JUB
☐ Knauf Insulation
☐ LITHOPLAST
☐ Nexwarm
☐ RHEINZINK
☐ Tondach

Use

☑ all
☐ Balconies and terraces
☐ Dočasná izolace
☐ Finální vrstva
☐ Flat roofs
☐ Green roofs

Other (9)

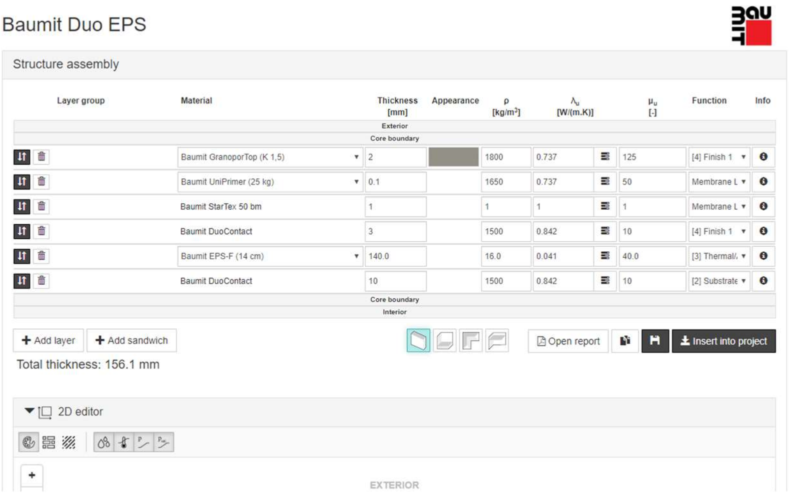
Subcategory

237 products

Product	Thickness	Radon diffusion ($x10^{-12}$)	Manufacturer
1911 AQUASTOP HARD Síranovzdorný cementový nátěr	-	-	Cemix
1920 AQUASTOP CRYSTAL Nátěr pro utěsnění betonových konstrukcí krystalizací	-	-	Cemix
1930 AQUASTOP BITUM 2K Hydroizolační, lepící stěrková a nátěrová hmota z modifikované, vlákny vyztužené bitumenové emulze a urychlovače tuhnutí v práškové formě.	-	-	Cemix
1940 AQUASTOP ELASTIC 2K Tekutá elastomerní vodovzorná dvousložková cementová membrána pro vnitřní i vnější užití s krytem i bez krytu	-	-	Cemix
8110 AQUASTOP IN Jednosložková disperzní rychletuhnoucí stěrka pod obklady k provádění hydroizolační vrstvy pro vnitřní použití	-	-	Cemix
8115 RAKO SE6			

Figure 10 - Browsing waterproofing materials.

In the tab building constructions, you can browse construction sandwiches for ETICS system, roofs, walls etc. When you pick a construction you need, it automatically opens them in editor.



Baumit Duo EPS

Structure assembly

Layer group	Material	Thickness [mm]	Appearance	ρ [kg/m³]	λ_u [W/(m.K)]	μ_u [-]	Function	Info
Exterior								
Core boundary								
	Baumit GranoporTop (K 1,5)	2	▨	1800	0.737	125	[4] Finish 1	ℹ
	Baumit UniPrimer (25 kg)	0.1		1650	0.737	50	Membrane L	ℹ
	Baumit StarTex 50 bm	1		1	1	1	Membrane L	ℹ
	Baumit DuoContact	3		1500	0.842	10	[4] Finish 1	ℹ
	Baumit EPS-F (14 cm)	140.0		16.0	0.041	40.0	[3] Thermal/	ℹ
	Baumit DuoContact	10		1500	0.842	10	[2] Substrate	ℹ
Core boundary								
Interior								

Total thickness: 156.1 mm

Figure 11 - Sandwich editor.

Another very useful part is library of families. You just pick the manufacturer and browse families like windows, walls, lintels etc.

Figure 12 - Available Manufacturers.

I think the most useful part of BIMTech is cadastral maps. There is map and 3D terrain database for whole Europe and America, but the cadastral maps are just for about the half of Europe.

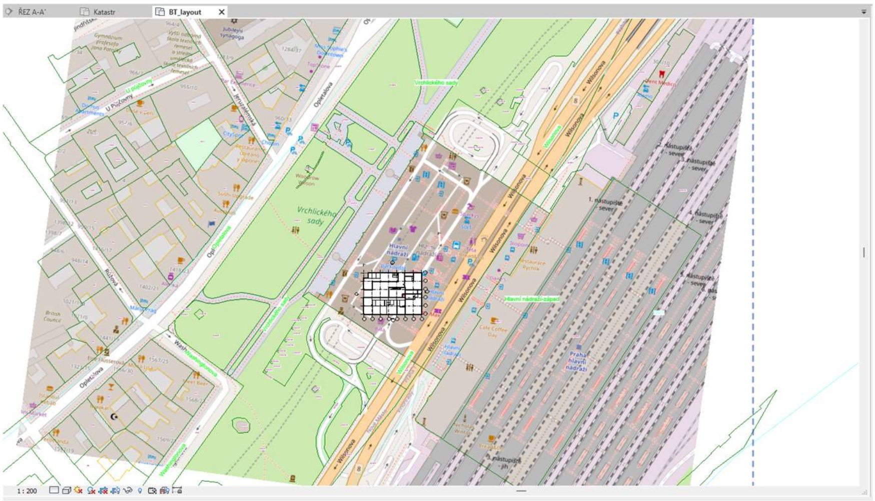


Figure 13 – Generated cadastral map in the project, the map is imported in new layout.

Advantages or disadvantages:

+ Very useful and complex plugin, that can import many important things in your project.

+ All of the data are added into database directly from the manufacturers.

1.5.5) Planary

Download source:

Official Autodesk plugin library:

https://apps.autodesk.com/RVT/en/Detail/Index?id=1230915056448788893&appLang=en&os=Win64

Official website of the plugin:

https://agileform.ca

Pricing:

Basic $79/ month, billed monthly, one fixed seat.

Floating $255/ month, billed monthly, unlimited users/ installations.

Student license.

Tested on 28 days free trial.

Compatibility:

Revit 2019 – 2023.

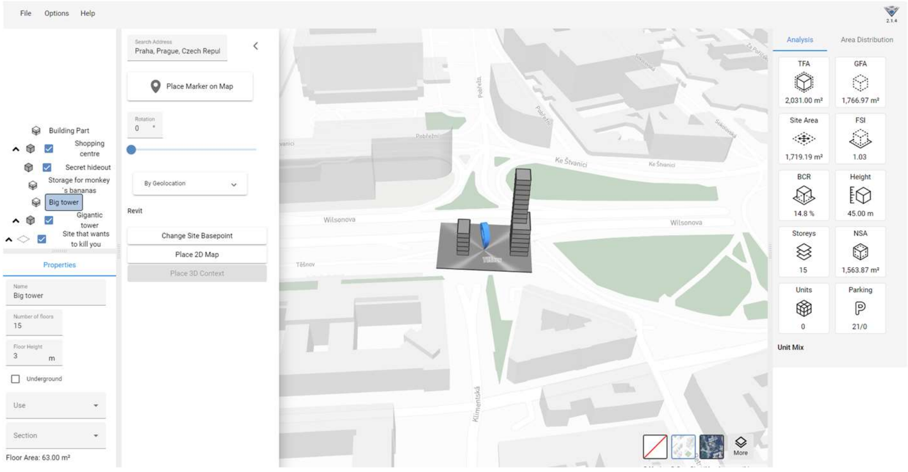
Tested version of the plugin: 2.1.4.

Description:

Planary is simple to use, but effective plugin which can help with massing study for sites.

Planary have a map with a model of buildings in cities, but you can´t import it into your model. First, you must create site. Buildings should be modelled as filled regions. Then, you can assign regions into planary. Then, you must assign parameters like number of floors, name of building etc. Also, you instantly see statistics like total floor area of all buildings, area of the site, height of the tallest building in the site, number of parking spots you need (calculation is based on total floor area), distribution of building space etc. It also has live connection to Revit, so you can draw site in Revit and every building instantly add into Planary. You can also edit floor plan dimensions and every part of analysis is changing in real time.

Exporting data can be into a picture or MS Excel format. Exporting modelled buildings can be done directly into the Revit, or into SketchUP.



Search Address: Praha, Prague, Czech Repul

Rotation: 0 °

Name: Big tower

Number of floors: 15

Floor Height: 3 m

□ Underground

Use

Section

Floor Area: 63.00 m²

TFA: 2,031.00 m²

GFA: 1,766.97 m²

Site Area: 1,719.19 m²

FSI: 1.03

BCR: 14.8 %

Height: 45.00 m

Storeys: 15

NSA: 1,563.87 m²

Units: 0

Parking: 21/0

Figure 14 - Site with my own 2 buildings shown in model of the city.

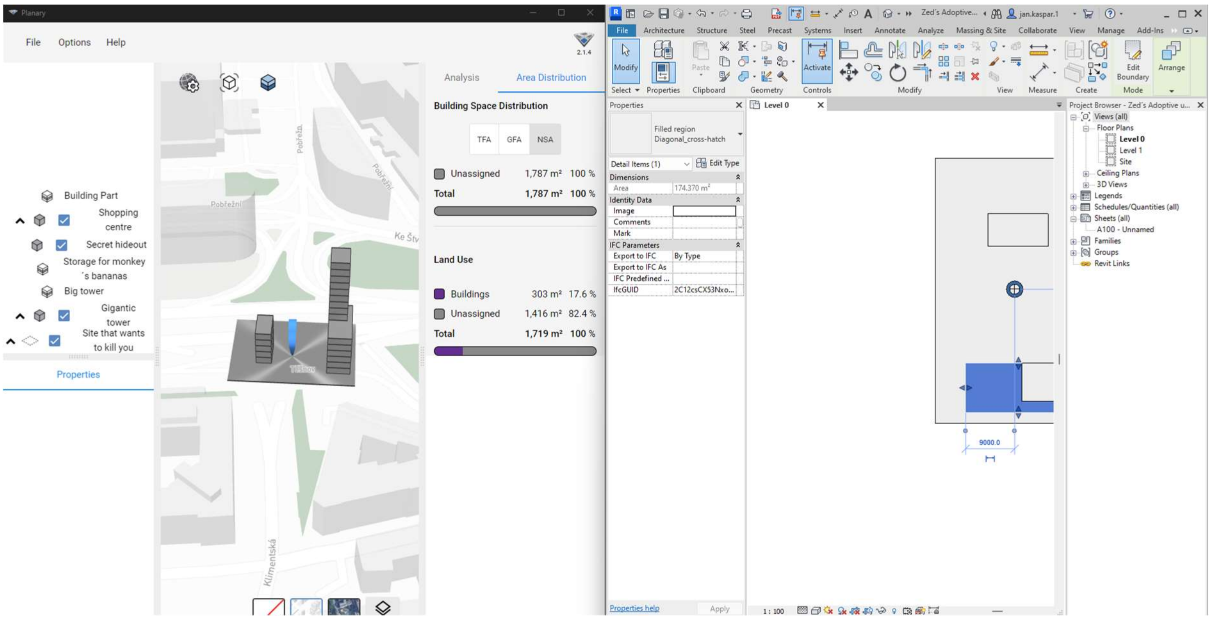
Advantages or disadvantages:

+ Simply and effective.

+ Live connection.

- Starting was a bit complicated, because it was hard to find in which format you should draw and insert site.

Few screenshots:



Building Space Distribution

▨ Unassigned	1,787 m²	100 %
Total	1,787 m²	100 %

Land Use

🟪 Buildings	303 m²	17.6 %
▨ Unassigned	1,416 m²	82.4 %
Total	1,719 m²	100 %

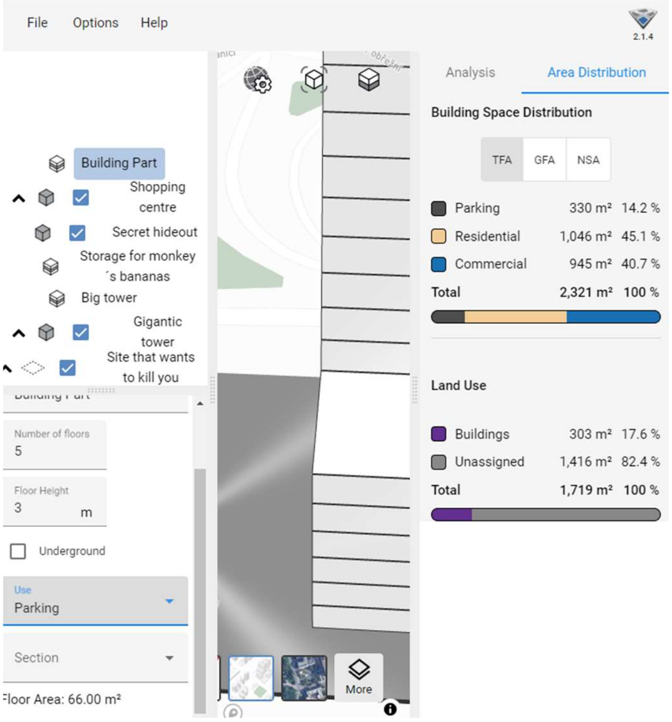
Properties

Filled region
Diagonal_cross-hatch

Detail Items (1)

Dimensions	
Area	174.370 m²
Identity Data	
Image	
Comments	
Mark	
IFC Parameters	
Export to IFC	By Type
Export to IFC As	
IFC Predefined ...	
IfcGUID	2C12csCX53Nxo...

Figure 15 – Live connection to Revit.



Building Space Distribution

■ Parking	330 m²	14.2 %
🟧 Residential	1,046 m²	45.1 %
🟦 Commercial	945 m²	40.7 %
Total	2,321 m²	100 %

Land Use

🟪 Buildings	303 m²	17.6 %
▨ Unassigned	1,416 m²	82.4 %
Total	1,719 m²	100 %

Number of floors
5

Floor Height
3 m

☐ Underground

Use
Parking

Section

Floor Area: 66.00 m²

Figure 16 - Analysis of building space distribution.

1.5.6) ID:Legend / doors & windows schedule

Download source:

Official Autodesk plugin library:

https://apps.autodesk.com/RVT/en/Detail/Index?id=9123792650018338024&appLang=en&os=Win64&autostart=true

Official website of the plugin:

https://idarqtools.com

Pricing:

Paid version $10.00/ 7 year.

Free trial – limits the number of legends that can be generate.

Compatibility:

Revit 2021 – 2024.

Tested version of the plugin: 2.1.1

Description:

Creates legends for doors or windows. Using plugin is very intuitive, just by few clicks you can create multiple legends for each type/ dimension of doors/ windows. There is also option to save your favourite settings.

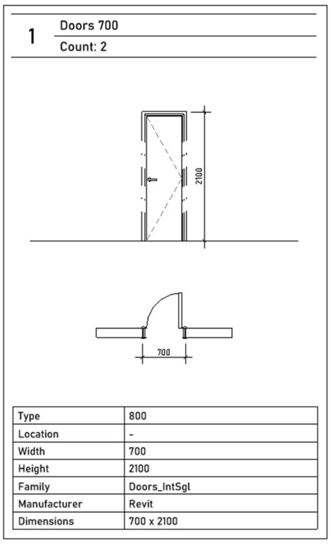
Advantages or disadvantages:

+ Legend looks pretty good.

+ Easy to use.

- Creates separate legends for each door type/ dimensions.

Few screenshots:

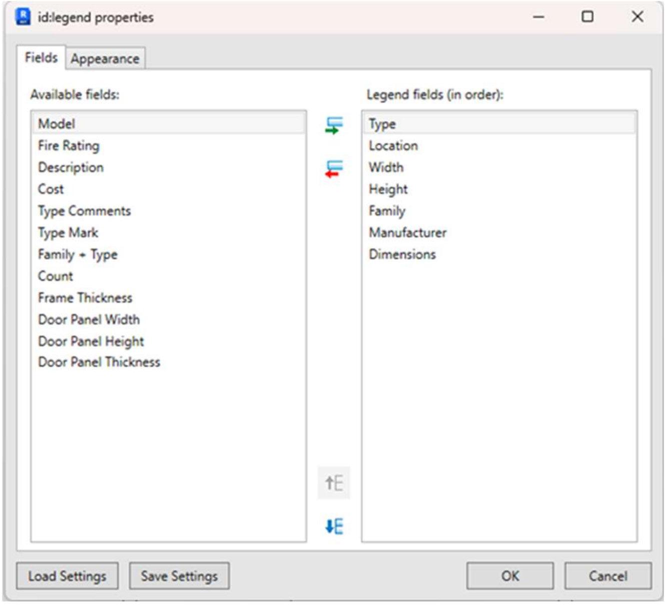


1 Doors 700

Count: 2

Type	800
Location	-
Width	700
Height	2100
Family	Doors_IntSgl
Manufacturer	Revit
Dimensions	700 x 2100

Figure 17 - Door legend.



Available fields:

- Model
- Fire Rating
- Description
- Cost
- Type Comments
- Type Mark
- Family + Type
- Count
- Frame Thickness
- Door Panel Width
- Door Panel Height
- Door Panel Thickness

Legend fields (in order):

- Type
- Location
- Width
- Height
- Family
- Manufacturer
- Dimensions

Figure 18 - Creating a legend.

1.5.7) Tables

Download source:

Official Autodesk plugin library:

https://apps.autodesk.com/RVT/en/Detail/Index?id=2349381363455444680&appLang=en&os=Win64

Official website of the plugin:

https://www.planworks.de

Pricing:

250.00 € (+ VAT)/ year.

600.00 € (+VAT)/ year.

Tested on 30 days free trial.

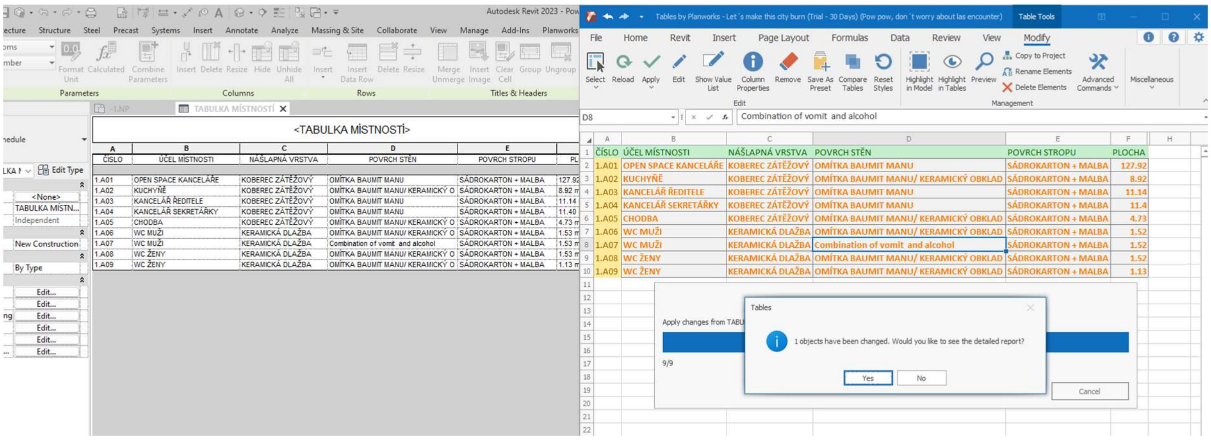
Compatibility:

Revit 2018 and newer versions.

Tested version of the app: 2023.1.0

Description:

Simple program to maintain tables in Revit. When you open the plugin, you just chose table that you want to edit, then just edit it as if you were in some table editor like MS Excel. It don´t have live connection to Revit, but after clicking on apply, it just take few seconds to make data in Revit up to date. There is also possibility to save data as .xlsx – Excel formate. Also.you can export the table into PDF or print directly on your printer.



D8 · Combination of vomit and alcohol

ČÍSLO	ÚČEL MÍSTNOSTI	NÁŠLAPNÁ VRSTVA	POVRCH STĚN	POVRCH STROPU	PLOCHA
1.A01	OPEN SPACE KANCELÁŘE	KOBEREC ZÁTĚŽOVÝ	OMÍTKA BAUMIT MANU	SÁDROKARTON + MALBA	127.92
1.A02	KUCHYŇĚ	KOBEREC ZÁTĚŽOVÝ	OMÍTKA BAUMIT MANU/ KERAMICKÝ OBKLAD	SÁDROKARTON + MALBA	8.92
1.A03	KANCELÁŘ ŘEDITELE	KOBEREC ZÁTĚŽOVÝ	OMÍTKA BAUMIT MANU	SÁDROKARTON + MALBA	11.14
1.A04	KANCELÁŘ SEKRETÁŘKY	KOBEREC ZÁTĚŽOVÝ	OMÍTKA BAUMIT MANU	SÁDROKARTON + MALBA	11.4
1.A05	CHODBA	KOBEREC ZÁTĚŽOVÝ	OMÍTKA BAUMIT MANU/ KERAMICKÝ OBKLAD	SÁDROKARTON + MALBA	4.73
1.A06	WC MUŽI	KERAMICKÁ DLAŽBA	OMÍTKA BAUMIT MANU/ KERAMICKÝ OBKLAD	SÁDROKARTON + MALBA	1.52
1.A07	WC MUŽI	KERAMICKÁ DLAŽBA	Combination of vomit and alcohol	SÁDROKARTON + MALBA	1.52
1.A08	WC ŽENY	KERAMICKÁ DLAŽBA	OMÍTKA BAUMIT MANU/ KERAMICKÝ OBKLAD	SÁDROKARTON + MALBA	1.52
1.A09	WC ŽENY	KERAMICKÁ DLAŽBA	OMÍTKA BAUMIT MANU/ KERAMICKÝ OBKLAD	SÁDROKARTON + MALBA	1.13

Apply changes from TABU

9/9

Tables

1 objects have been changed. Would you like to see the detailed report?

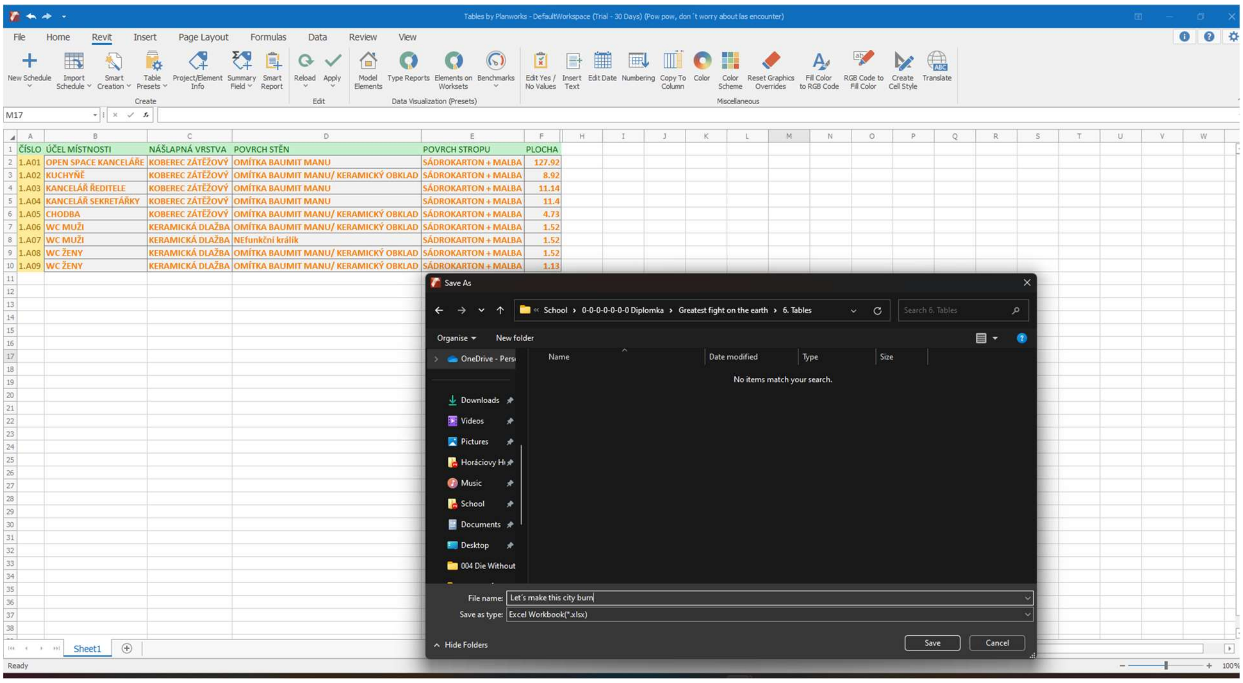
Figure 19 - Updated data in Revit.

Advantages and disadvantages:

+ Very intuitive, everything is where you expect it.

+ Works fast.

Few screenshots:



Tables by Planworks - DefaultWorkspace (Trial - 30 Days) (Pow pow, don´t worry about las encounter)

ČÍSLO	ÚČEL MÍSTNOSTI	NÁŠLAPNÁ VRSTVA	POVRCH STĚN	POVRCH STROPU	PLOCHA
1.A01	OPEN SPACE KANCELÁŘE	KOBEREC ZÁTĚŽOVÝ	OMÍTKA BAUMIT MANU	SÁDROKARTON + MALBA	127.92
1.A02	KUCHYŇĚ	KOBEREC ZÁTĚŽOVÝ	OMÍTKA BAUMIT MANU/ KERAMICKÝ OBKLAD	SÁDROKARTON + MALBA	8.92
1.A03	KANCELÁŘ ŘEDITELE	KOBEREC ZÁTĚŽOVÝ	OMÍTKA BAUMIT MANU	SÁDROKARTON + MALBA	11.14
1.A04	KANCELÁŘ SEKRETÁŘKY	KOBEREC ZÁTĚŽOVÝ	OMÍTKA BAUMIT MANU	SÁDROKARTON + MALBA	11.4
1.A05	CHODBA	KOBEREC ZÁTĚŽOVÝ	OMÍTKA BAUMIT MANU/ KERAMICKÝ OBKLAD	SÁDROKARTON + MALBA	4.73
1.A06	WC MUŽI	KERAMICKÁ DLAŽBA	OMÍTKA BAUMIT MANU/ KERAMICKÝ OBKLAD	SÁDROKARTON + MALBA	1.52
1.A07	WC MUŽI	KERAMICKÁ DLAŽBA	NEfunkční králík	SÁDROKARTON + MALBA	1.52
1.A08	WC ŽENY	KERAMICKÁ DLAŽBA	OMÍTKA BAUMIT MANU/ KERAMICKÝ OBKLAD	SÁDROKARTON + MALBA	1.52
1.A09	WC ŽENY	KERAMICKÁ DLAŽBA	OMÍTKA BAUMIT MANU/ KERAMICKÝ OBKLAD	SÁDROKARTON + MALBA	1.13

Save As

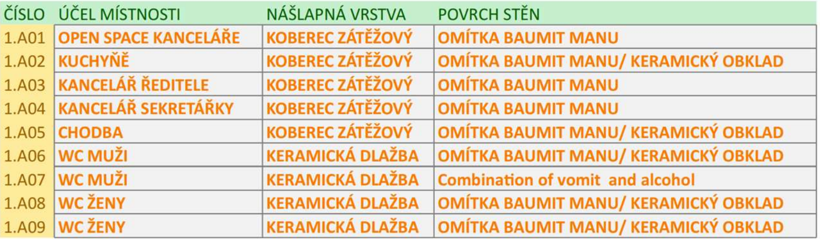
School › 0-0-0-0-0-0 Diplomka › Greatest fight on the earth › 6. Tables

No items match your search.

File name: Let´s make this city burn

Save as type: Excel Workbook(*.xlsx)

Figure 20 - Exporting to .XLSX.



ČÍSLO	ÚČEL MÍSTNOSTI	NÁŠLAPNÁ VRSTVA	POVRCH STĚN
1.A01	OPEN SPACE KANCELÁŘE	KOBEREC ZÁTĚŽOVÝ	OMÍTKA BAUMIT MANU
1.A02	KUCHYŇĚ	KOBEREC ZÁTĚŽOVÝ	OMÍTKA BAUMIT MANU/ KERAMICKÝ OBKLAD
1.A03	KANCELÁŘ ŘEDITELE	KOBEREC ZÁTĚŽOVÝ	OMÍTKA BAUMIT MANU
1.A04	KANCELÁŘ SEKRETÁŘKY	KOBEREC ZÁTĚŽOVÝ	OMÍTKA BAUMIT MANU
1.A05	CHODBA	KOBEREC ZÁTĚŽOVÝ	OMÍTKA BAUMIT MANU/ KERAMICKÝ OBKLAD
1.A06	WC MUŽI	KERAMICKÁ DLAŽBA	OMÍTKA BAUMIT MANU/ KERAMICKÝ OBKLAD
1.A07	WC MUŽI	KERAMICKÁ DLAŽBA	Combination of vomit and alcohol
1.A08	WC ŽENY	KERAMICKÁ DLAŽBA	OMÍTKA BAUMIT MANU/ KERAMICKÝ OBKLAD
1.A09	WC ŽENY	KERAMICKÁ DLAŽBA	OMÍTKA BAUMIT MANU/ KERAMICKÝ OBKLAD

Figure 21 - Table in PDF.

1.5.8) Ideate software bundle.

Download source:

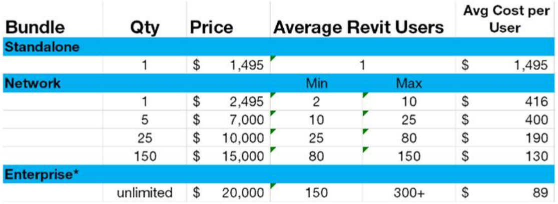
Official Autodesk plugin library:

https://apps.autodesk.com/RVT/en/Detail/Index?id=6300849540418062587&appLang=en&os=Win64

Official website of the plugin:

https://ideatesoftware.com

Pricing:



Bundle	Qty	Price	Average Revit Users		Avg Cost per User
Standalone					
	1	$ 1,495	1		$ 1,495
Network			Min	Max	
	1	$ 2,495	2	10	$ 416
	5	$ 7,000	10	25	$ 400
	25	$ 10,000	25	80	$ 190
	150	$ 15,000	80	150	$ 130
Enterprise*					
	unlimited	$ 20,000	150	300+	$ 89

Figure 22 - Pricing of ideate software bundle.

Tested on time limited free trial.

Compatibility:

Revit 2020 and newer versions.

Tested version of the app: 2024.0

Description:

Bundle of 5 plugins, you can activate just the needed ones.

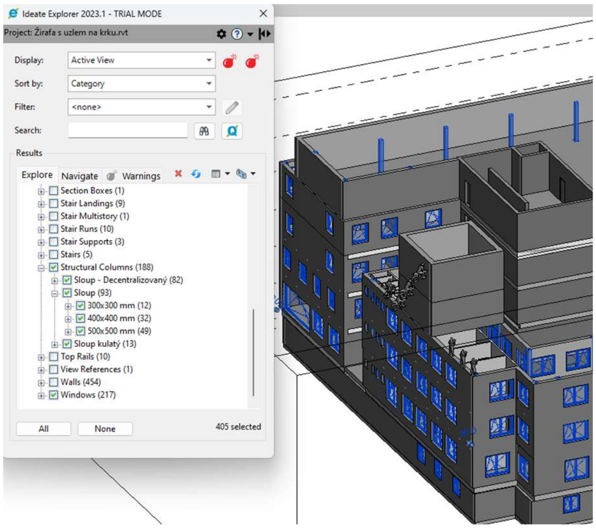
Ideate BIMLink – Revit management data, connects to excel.

Ideate Explorer – Sorts particles of model and highlights them, you can also filter warnings and highlight them.

Ideate StyleManager – Analyses how effectively model uses the template. It shows how often is material (or line pattern, etc) used in model. It allows you to delete or merge materials.

Ideate Sticky – link PDFs or excel files to Revit.

IdeateApps – Bunch of apps to manage project more effectively, like XRay – finds you desired object in project and show in selected active view.



Ideate Explorer 2023.1 - TRIAL MODE

Project: Žirafa s uzlem na krku.rvt

Display: Active View

Sort by: Category

Filter: <none>

Search:

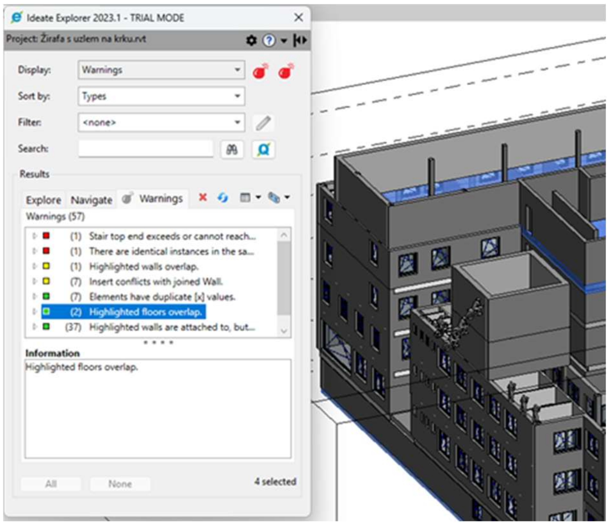
Results

Explore | Navigate | Warnings

- ☐ Section Boxes (1)
- ☐ Stair Landings (9)
- ☐ Stair Multistory (1)
- ☐ Stair Runs (10)
- ☐ Stair Supports (3)
- ☐ Stairs (5)
- ☑ Structural Columns (188)
  - ☑ Sloup - Decentralizovaný (82)
  - ☑ Sloup (93)
    - ☑ 300x300 mm (12)
    - ☑ 400x400 mm (32)
    - ☑ 500x500 mm (49)
  - ☑ Sloup kulatý (13)
- ☐ Top Rails (10)
- ☐ View References (1)
- ☐ Walls (454)
- ☑ Windows (217)

All | None | 405 selected

Figure 23 - Exploring the model.



Ideate Explorer 2023.1 - TRIAL MODE

Project: Žirafa s uzlem na krku.rvt

Display: Warnings

Sort by: Types

Filter: <none>

Search:

Results

Explore | Navigate | Warnings

Warnings (57)

- (1) Stair top end exceeds or cannot reach...
- (1) There are identical instances in the sa...
- (1) Highlighted walls overlap.
- (7) Insert conflicts with joined Wall.
- (7) Elements have duplicate [x] values.
- (2) Highlighted floors overlap.
- (37) Highlighted walls are attached to, but...

Information

Highlighted floors overlap.

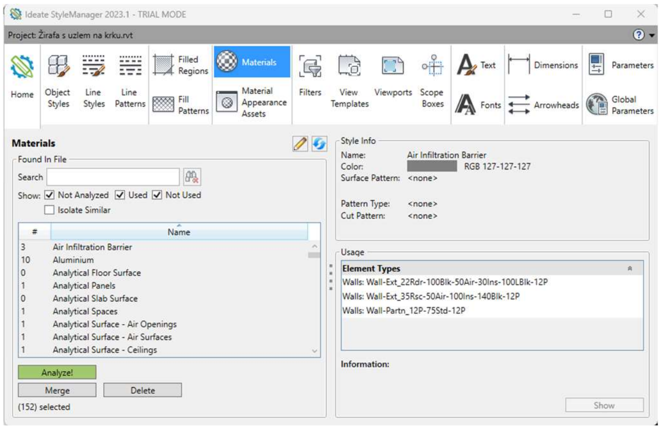
All | None | 4 selected

Figure 24 - Highlighting warning errors in the model.

Advantages or disadvantages:

+ Lot of useful plugins and you chose to pay and use only the parts you need.

Few screenshots:



Ideate StyleManager 2023.1 - TRIAL MODE

Project: Žirafa s uzlem na krku.rvt

Materials

Found In File

Search

Show: ☑ Not Analyzed ☑ Used ☑ Not Used

☐ Isolate Similar

#	Name
3	Air Infiltration Barrier
10	Aluminium
0	Analytical Floor Surface
1	Analytical Panels
0	Analytical Slab Surface
1	Analytical Spaces
1	Analytical Surface - Air Openings
1	Analytical Surface - Air Surfaces
1	Analytical Surface - Ceilings

Analyze! | Merge | Delete

(152) selected

Style Info

Name: Air Infiltration Barrier

Color: RGB 127-127-127

Surface Pattern: <none>

Pattern Type: <none>

Cut Pattern: <none>

Usage

Element Types

Walls: Wall-Ext_22Rdr-100Blk-50Air-30Ins-100LBlk-12P

Walls: Wall-Ext_35Rsc-50Air-100Ins-140Blk-12P

Walls: Wall-Partn_12P-75Std-12P

Information:

Show

Figure 25 - Analysed materials.

1.5.9) Enscape

Download source:

Official Autodesk plugin library:

https://apps.autodesk.com/RVT/en/Detail/Index?id=2629595860167800202&appLang=en&os=Win64

Official website of the plugin:

https://www.enscape3d.com/

Pricing:

Fixed seat license 42.90 €/ month.

Floating license 76.90 €/ month.

Tested on 14 days free trial.

Compatibility:

Revit 2019 and newer versions.

Tested version of the app: 3.5

Description:

Enscape is real time model viewer with many possibilities. It creates nice visualizations by just few clicks. You can render picture, panorama where you can look around in 3D, or animation with flying camera around, or through the building. There is also asset library with many models of people, cars, bikes, etc. You can also add city context. The plugin works also as project viewer – shows you objects in projects sorted by type of object and families.

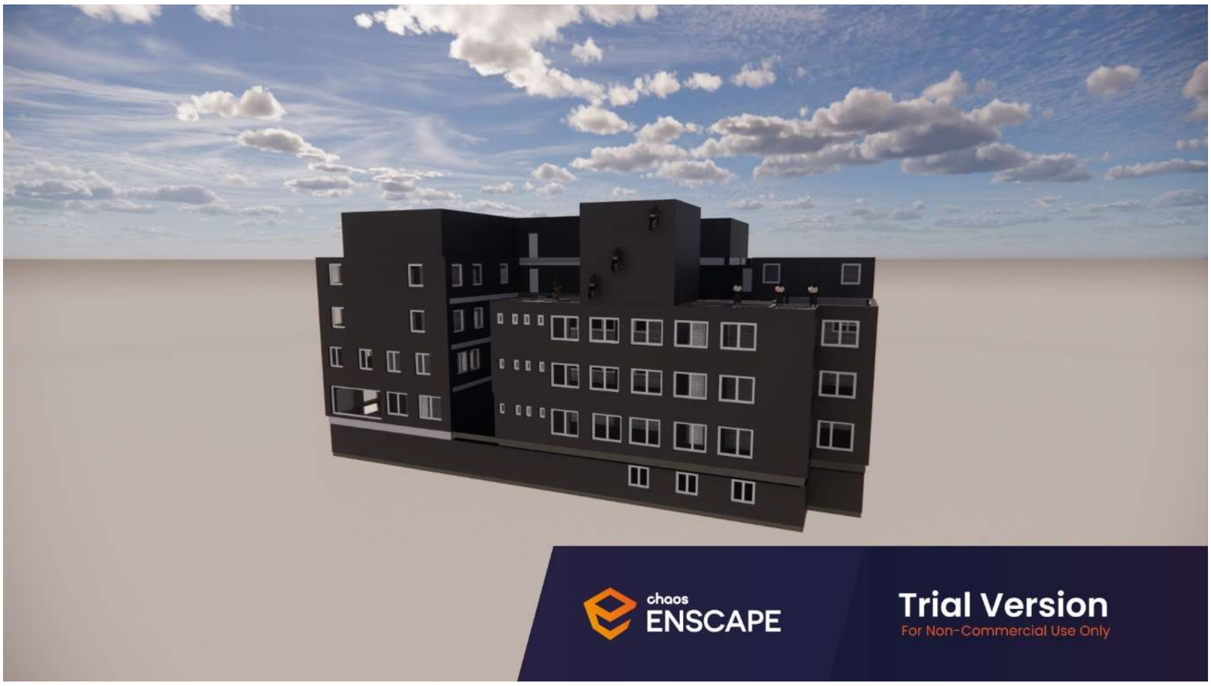


Figure 26 - Visualisation created by Enscape.

Advantages and disadvantages:

+ Easy to use.

+ Large Asset library

Few screenshots:

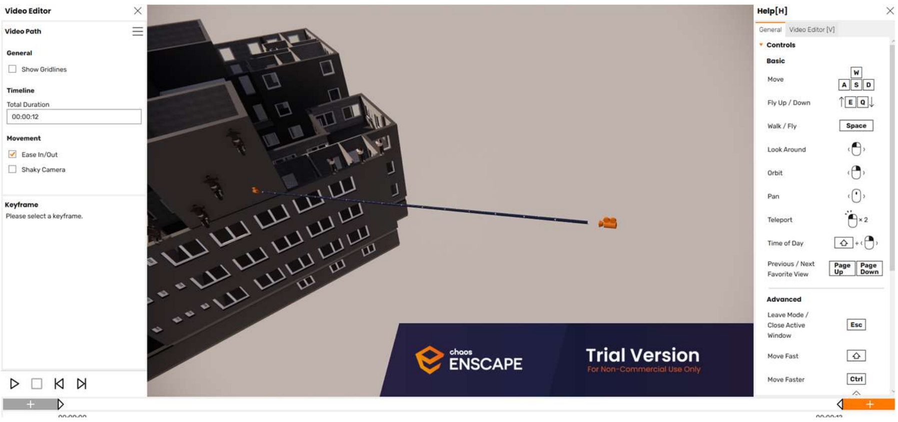


Figure 27 - Animation editor.

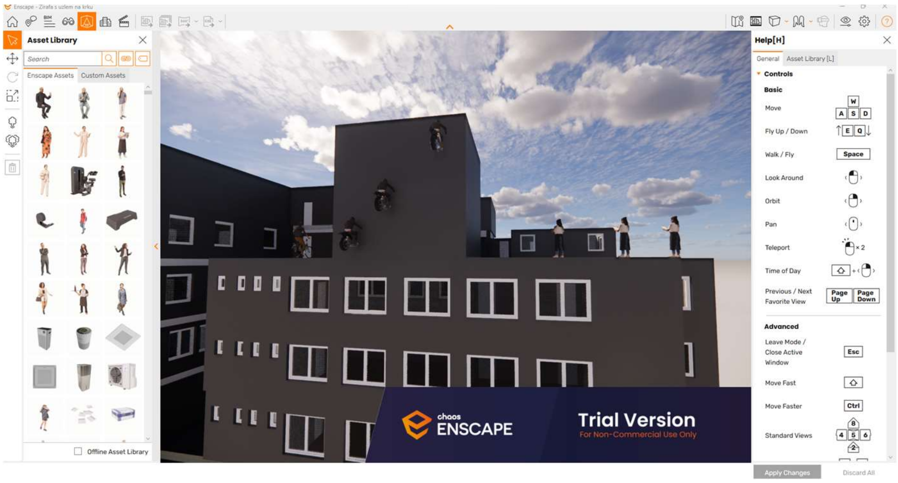
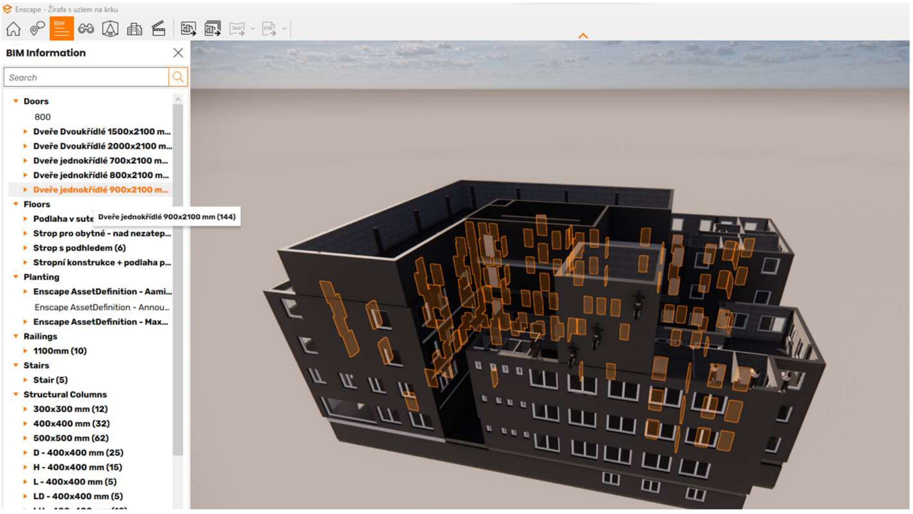


Figure 28 - Asset Library.



- Doors
  - 800
  - Dveře Dvoukřídlé 1500x2100 m...
  - Dveře Dvoukřídlé 2000x2100 m...
  - Dveře jednokřídlé 700x2100 m...
  - Dveře jednokřídlé 800x2100 m...
  - Dveře jednokřídlé 900x2100 m...
- Floors
  - Podlaha v sute
  - Strop pro obytné - nad nezatep...
  - Strop s podhledem (6)
  - Stropní konstrukce + podlaha p...
- Planting
  - Enscape AssetDefinition - Aami...
  - Enscape AssetDefinition - Annou...
  - Enscape AssetDefinition - Max...
- Railings
  - 1100mm (10)
- Stairs
  - Stair (5)
- Structural Columns
  - 300x300 mm (12)
  - 400x400 mm (32)
  - 500x500 mm (62)
  - D - 400x400 mm (25)
  - H - 400x400 mm (15)
  - L - 400x400 mm (5)
  - LD - 400x400 mm (5)

Dveře jednokřídlé 900x2100 mm (144)

Figure 29 - Project explorer

1.5.10) 3DPDF Converter for Autodesk Revit

Download source:

Official Autodesk plugin library:

https://apps.autodesk.com/RVT/en/Detail/Index?id=4991490546054443185&appLang=en&os=Win64

Official website of the plugin:

http://visionworkplace.com/

Pricing:

One seat $100.00.

Two seats $180.00.

Tested on 14 days free trial.

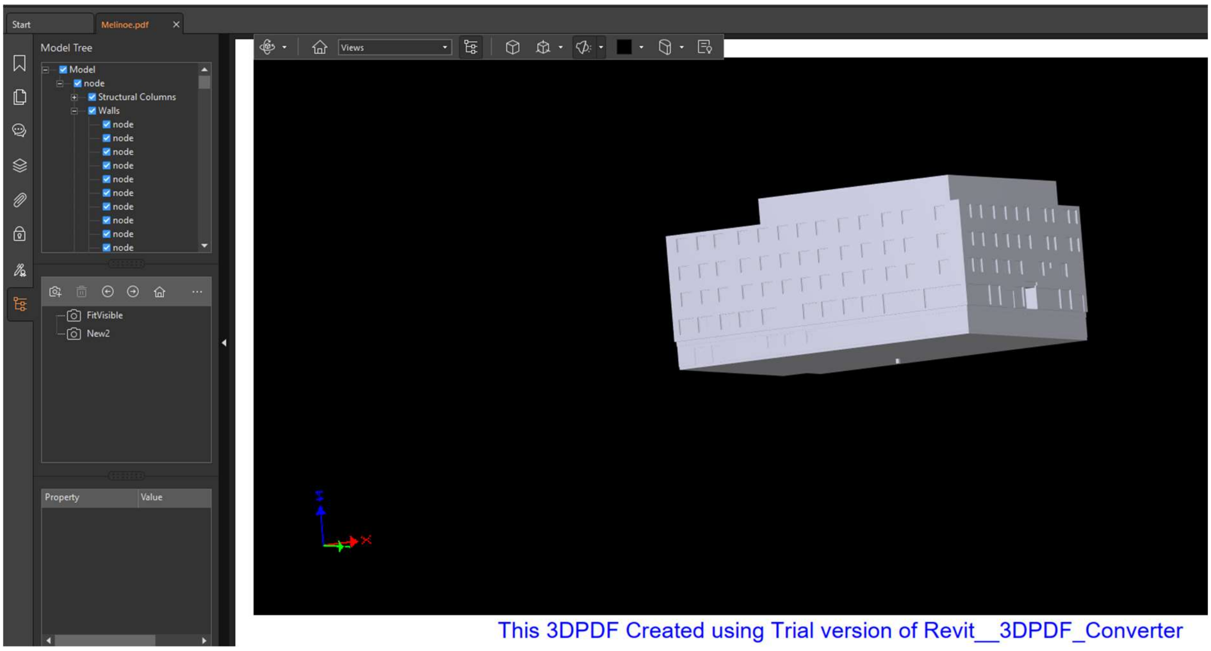
Compatibility:

Revit 2019 - 2023.

Tested version of the app: 23.22.12.0

Description:

Allows you to export your model as 3D PDF. PDF is generated as model with option to hide any part of model you want to hide. You also can set light, background, and some other features for your PDF, most of them can be changed later in PDF viewer.



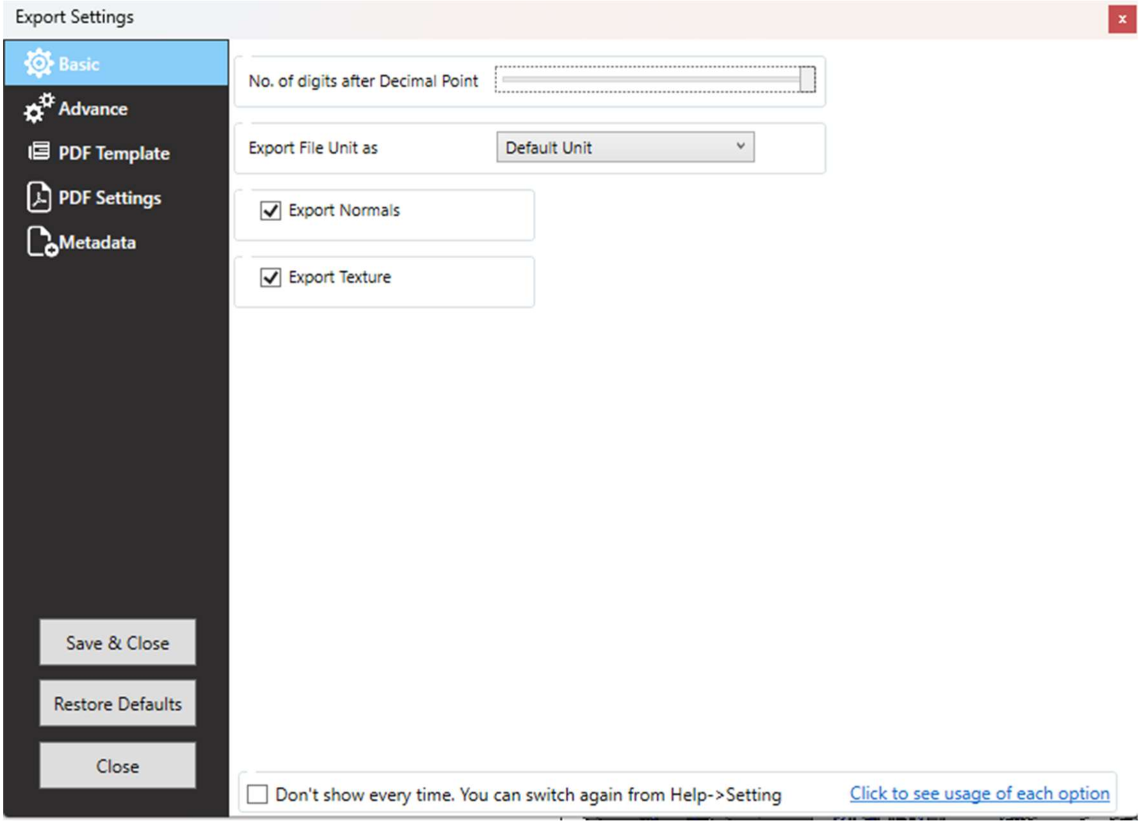
This 3DPDF Created using Trial version of Revit__3DPDF_Converter

Figure 30 - Generated 3D PDF.

Advantages or disadvantages:

+ Simple and working.

Few screenshots:



No. of digits after Decimal Point

Export File Unit as: Default Unit

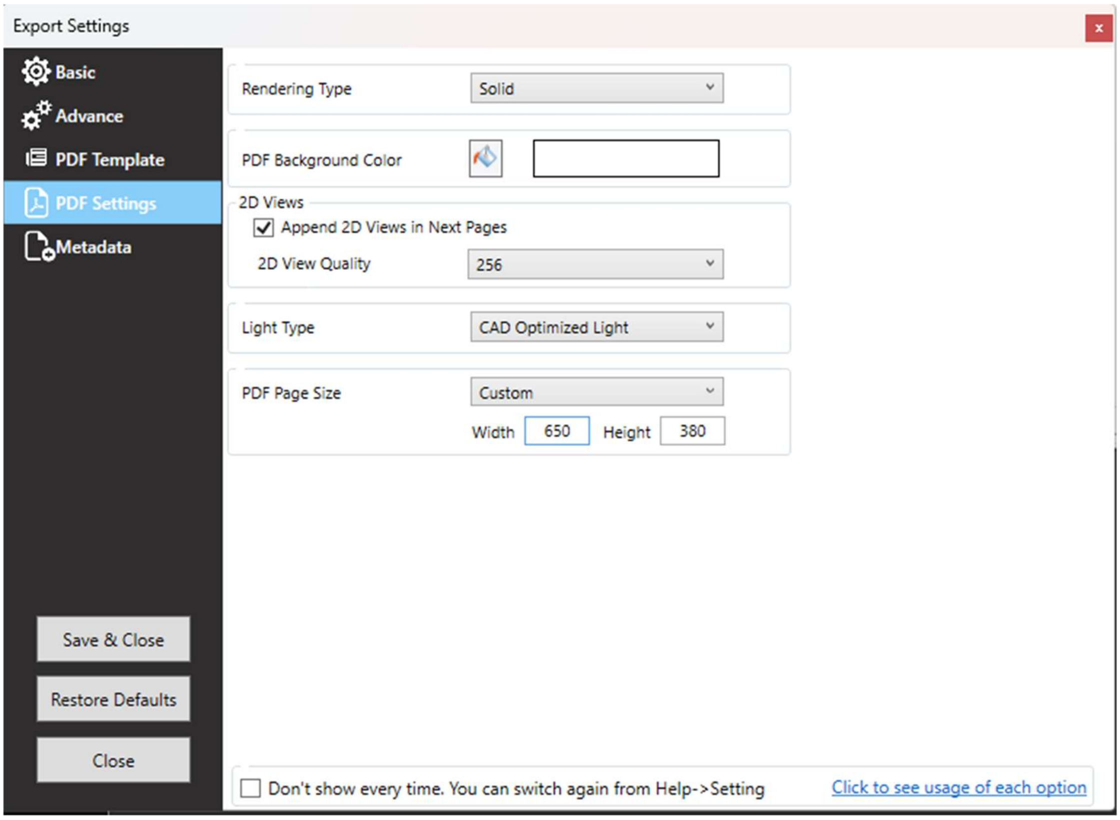
☑ Export Normals

☑ Export Texture

□ Don't show every time. You can switch again from Help->Setting

Click to see usage of each option

Figure 31 - Basic settings of exported PDF.



Rendering Type: Solid

PDF Background Color

2D Views

☑ Append 2D Views in Next Pages

2D View Quality: 256

Light Type: CAD Optimized Light

PDF Page Size: Custom

Width 650 Height 380

□ Don't show every time. You can switch again from Help->Setting

Click to see usage of each option

Figure 32 - Settings of the future PDF.

1.5.11) PCL PartsLab

Download source:

Official Autodesk plugin library:

https://apps.autodesk.com/RVT/en/Detail/Index?id=8116491063486271272&appLang=en&os=Win64

Official website of the plugin:

http://partslab.pcl.com

Pricing:

Not to be found on their website.

Tested on 30 days free trial.

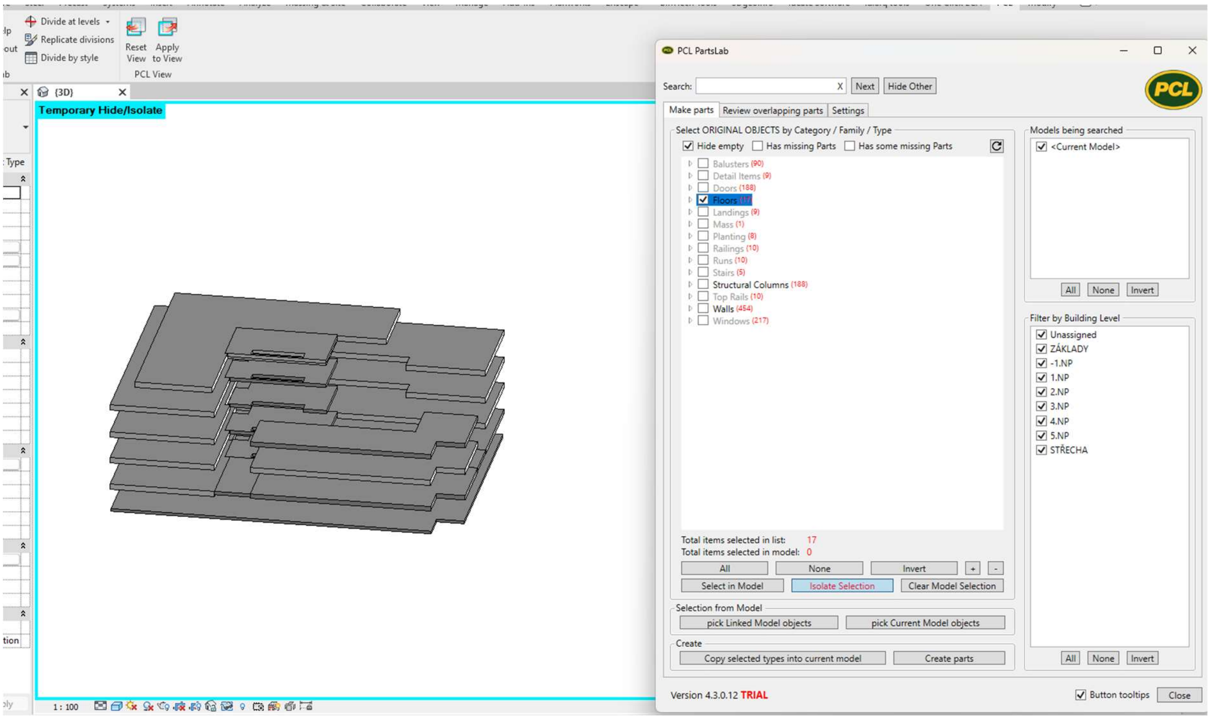
Compatibility:

Revit 2021 and newer versions.

Tested version of the app: 4.3.0.12

Description:

PCL PartsLab is model browser, which allows you to separate the part of model more effectively than in Revit. If you have more models linked into one, plugin allows you to make parts of inserted model native.



Select ORIGINAL OBJECTS by Category / Family / Type

☑ Hide empty ☐ Has missing Parts ☐ Has some missing Parts

- ☐ Balusters (90)
- ☐ Detail Items (9)
- ☐ Doors (188)
- ☑ Floors (17)
- ☐ Landings (9)
- ☐ Mass (1)
- ☐ Planting (8)
- ☐ Railings (10)
- ☐ Runs (10)
- ☐ Stairs (5)
- ☐ Structural Columns (188)
- ☐ Top Rails (10)
- ☐ Walls (454)
- ☐ Windows (217)

Total items selected in list: 17
Total items selected in model: 0

Selection from Model

Create

Models being searched

☑ <Current Model>

Filter by Building Level

- ☑ Unassigned
- ☑ ZÁKLADY
- ☑ -1.NP
- ☑ 1.NP
- ☑ 2.NP
- ☑ 3.NP
- ☑ 4.NP
- ☑ 5.NP
- ☑ STŘECHA

Version 4.3.0.12 TRIAL

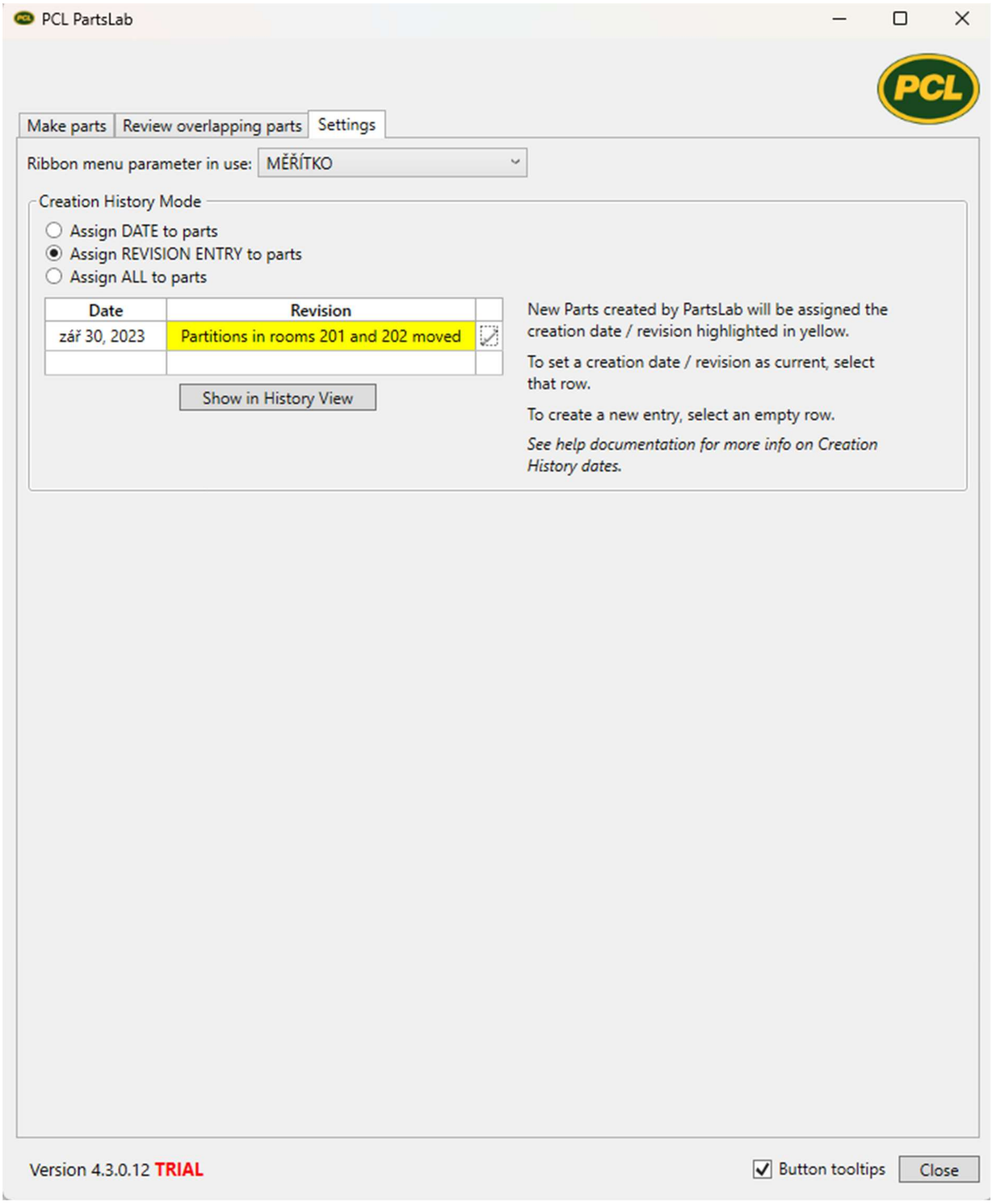
☑ Button tooltips

Figure 33 - Browsing all floors.

Advantages or disadvantages:

+ Simple and useful.

Few screenshots:



Ribbon menu parameter in use: MĚŘÍTKO

Creation History Mode

○ Assign DATE to parts
● Assign REVISION ENTRY to parts
○ Assign ALL to parts

Date	Revision	
zář 30, 2023	Partitions in rooms 201 and 202 moved	

New Parts created by PartsLab will be assigned the creation date / revision highlighted in yellow.

To set a creation date / revision as current, select that row.

To create a new entry, select an empty row.

See help documentation for more info on Creation History dates.

Version 4.3.0.12 TRIAL

☑ Button tooltips

Figure 34 - History of revisions.

1.5.12) Overwiew

Tab. 1 – Summary of tested Add-ins

Num.	Name of the plugin	Use of plugin	Calculates thermal resistance?
1	Tally	Ecological cycle of building	No
2	Carbo Life Calculator	Calculates embodied carbon in buildings	No
3	BIMTech	Cadastral maps, materials and constructions database	No
4	Planary	City + project context	No
5	ID:Legend / doors & windows shedule	Doors and windows legends	No
6	Tables	Table editor like MS Excel.	No
7	Ideate software bundle	Project browsing, analytics, connection to MS Excel.	No
8	Enscape	Project browser, visualizations	No
9	3DPDF converter for Autodesk Revit	3D PDF export	No
10	PCL PartsLab	Project browser	No

Conclusion:

There are plenty of plugins, that can improve quality of BIM engineering in Revit in the many different ways. Calculating ecological impact on environment by according to used materials, making light calculations, creating legends, exporting into 3D pdf, or just browsing the project.

However, I have not found a plugin that calculates thermal heat resistance while including thermal transfer surface resistance. So, creating a plugin like that seems to be a worthy work.

2) Sample project

In this chapter I will show you drawings of small, simplified house. The object has 1 zone and was created for the testing purposes. Aim of this object was to have all constructions which the add-in can calculate. I will write all marked compositions that will be included in the plugin calculations, and I will compare Revit´s U-Values with manual calculations according to ČSN.

2.1) Sketches

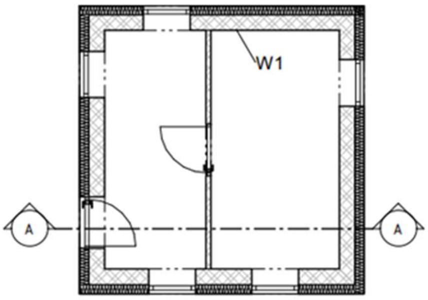


Figure 35 - Sketch of 1.NP floor plan.

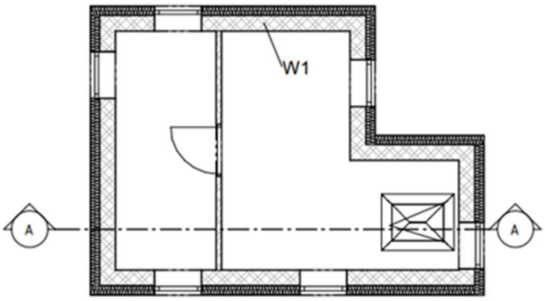


Figure 36 - Sketch of 2.NP floor plan.

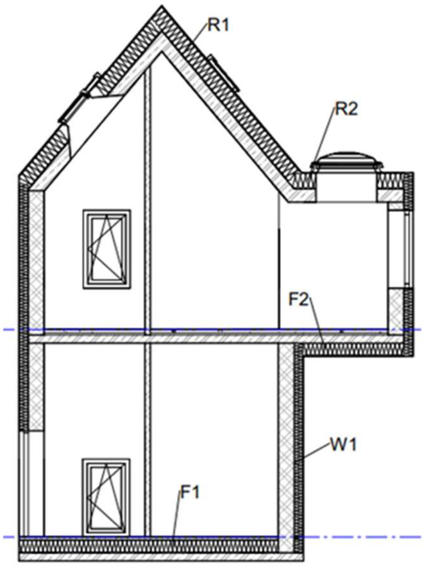


Figure 37 - Sketch of the section.

Compositions

W1 – Wall

EPS,	d = 180 mm,	λ = 0.0348 W/(m*K)
Bricks,	d = 300 mm,	λ = 0.2000 W/(m*K)

F1 – Floor on the ground

PVC,	d = 2 mm,	λ = 0.1670 W/(m*K)
Adhesive,	d = 2 mm,	λ = 1.2000 W/(m*K)
Screed,	d = 51 mm,	λ = 1.0460 W/(m*K)
Separation layer		
EPS,	d = 250 mm,	λ = 0.0348 W/(m*K)
Asphalt sheet,	d = 5 mm,	λ = 1.2000 W/(m*K)
Concrete,	d = 150 mm,	λ = 2.0000 W/(m*K)

F2 – Bay floor

PVC,	d = 2 mm,	λ = 0.1670 W/(m*K)
Adhesive,	d = 2 mm,	λ = 1.2000 W/(m*K)
Screed,	d = 56 mm,	λ = 1.0460 W/(m*K)
Separation layer		
EPS,	d = 50 mm,	λ = 0.0348 W/(m*K)
Concrete,	d = 150 mm,	λ = 2.000 W/(m*K)
EPS,	d = 250 mm,	λ = 0.0348 W/(m*K)

R1 – Sloped roof

Ceramic roof tiles + air layer not counting to U-Value		
Hydro insulation,	d = 10 mm,	λ = 1.2000 W/(m*K)
EPS,	d = 300 mm,	λ = 0.0348 W/(m*K)
Hydro insulation,	d = 5 mm,	λ = 1.2000 W/(m*K)
Ytong comfort,	d = 250 mm,	λ = 2.0000 W/(m*K)

R2 – Flat roof

Hydro insulation,	d = 10 mm,	λ = 1.2000 W/(m*K)
EPS,	d = 300 mm,	λ = 0.0348 W/(m*K)
Hydro insulation,	d = 5 mm,	λ = 1.2000 W/(m*K)
Ytong comfort,	d = 250 mm,	λ = 2.0000 W/(m*K)

2.2) Manual U-Value calculations X default Revit U-Values

In my manual calculations I will need values of the R_{si} and R_{se} parameters, as I have said before there is a table in ČSN 73 0540 that declares the values [1].

Table. 2 – R_{si}, R_{se} values

Surface	Construction/ surface	R_{si}/ R_{se} [m^2*K/W]
Exterrior	Single shell	0.04
	Double shelled	Same as R_{si}
Ground	Contact with ground	0
Interrior	Wall	0.13
	Roof	0.10
	Floor	0.17

<u>Variables for the calculations</u>

U_{Re} = U-Value calculated by Revit.

$U_{ČSN}$ = U-Value calculated manually according to ČSN.

<u>W1 – Wall</u>

ČSN U-Value calculation

$$R_{eps} = \frac{d\ [m]}{\lambda\ [(m*K)/W]} = \frac{0.18}{0.0348} = 5.1724\ (m^2*K)/W$$

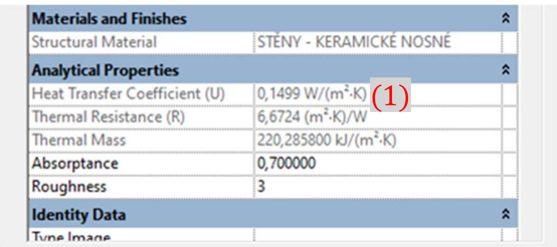
$$R_b = \frac{d\ [m]}{\lambda\ [(m*K)/W]} = \frac{0.3}{0.2} = 1.5\ (m^2*K)/W$$

$$R = R_{eps} + R_b = 1.5 + 5.1724 = 6.6724\ (m^2*K)/W$$

$$R_{si} = 0.13\ (m^2*K)/W$$

$$R_{se} = 0.04\ (m^2*K)/W$$

$$U_{ČSN} = \frac{1}{R_{si}+R+R_{se}} = \frac{1}{0.13+6.6724+0.04} = \mathbf{0.1461\ W/(m^2*K)}$$



Materials and Finishes	
Structural Material	STĚNY - KERAMICKÉ NOSNÉ
Analytical Properties	
Heat Transfer Coefficient (U)	0,1499 W/(m²·K) (1)
Thermal Resistance (R)	6,6724 (m²·K)/W
Thermal Mass	220,285800 kJ/(m²·K)
Absorptance	0,700000
Roughness	3
Identity Data	
Type Image	

Figure 38 - Thermal resistance of the wall.

$$1)\ \text{-}\ U_{Re} = \mathbf{0.1499\ W/(m^2*K)}$$

$$\frac{U_{Re}}{U_{ČSN}} = \frac{0.1499}{0.1461} = 1.026$$

$$\rightarrow 2.6\%\ difference$$

F1 – Floor on the ground

ČSN U-Value calculation

$$R_{PVC} = \frac{d\ [m]}{\lambda\ \left[\frac{m*K}{W}\right]} = \frac{0.002}{0.1670} = 0.0120 \frac{m^2*K}{W}$$

$$R_{adh} = \frac{d\ [m]}{\lambda\ \left[\frac{m*K}{W}\right]} = \frac{0.002}{1.2} = 0.0017 \frac{m^2*K}{W}$$

$$R_{Scr} = \frac{d\ [m]}{\lambda\ \left[\frac{m*K}{W}\right]} = \frac{0.051}{1.046} = 0.0488 \frac{m^2*K}{W}$$

$$R_{eps} = \frac{d\ [m]}{\lambda\ [(m*K)/W]} = \frac{0.250}{0.0348} = 7.1839\ (m^2*K)/W$$

$$R_{Asp} = \frac{d\ [m]}{\lambda\ [(m*K)/W]} = \frac{0.005}{1.2} = 0.0049\ (m^2*K)/W$$

$$R_C = \frac{d\ [m]}{\lambda\ [(m*K)/W]} = \frac{0.15}{2.0} = 0.075\ (m^2*K)/W$$

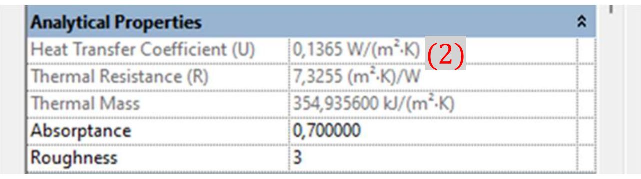
$$R = R_{PVC} + R_{adh} + R_{scr} + R_{eps} + R_{Asp} + R_C =$$

$$= 0.012 + 0.0017 + 0.0488 + 7.1839 + 0.0049 + 0.75 = 7.3255\ (m^2*K)/W$$

$$R_{si} = 0.17\ (m^2*K)/W$$

$$R_{se} = 0.0\ (m^2*K)/W$$

$$U_{ČSN} = \frac{1}{R_{si} + R + R_{se}} = \frac{1}{0.17 + 7.3255 + 0} = \mathbf{0.1334\ W/(m^2*K)}$$



Analytical Properties	
Heat Transfer Coefficient (U)	0,1365 W/(m²·K) (2)
Thermal Resistance (R)	7,3255 (m²·K)/W
Thermal Mass	354,935600 kJ/(m²·K)
Absorptance	0,700000
Roughness	3

Figure 39- Thermal resistance of the floor on the ground.

2) - $U_{Re} = \mathbf{0.1365\ W/(m^2*K)}$

F2 – Bay floor

ČSN U-Value calculation

$$R_{PVC} = \frac{d\ [m]}{\lambda\ \left[\frac{m*K}{W}\right]} = \frac{0.002}{0.1670} = 0.0120 \frac{m^2*K}{W}$$

$$R_{adh} = \frac{d\ [m]}{\lambda\ \left[\frac{m*K}{W}\right]} = \frac{0.002}{1.2} = 0.0017 \frac{m^2*K}{W}$$

$$R_{Scr} = \frac{d\ [m]}{\lambda\ \left[\frac{m*K}{W}\right]} = \frac{0.056}{1.046} = 0.0535 \frac{m^2*K}{W}$$

$$R_{eps} = \frac{d\ [m]}{\lambda\ [(m*K)/W]} = \frac{0.050}{0.0348} = 1.4368\ (m^2*K)/W$$

$$R_C = \frac{d\ [m]}{\lambda\ [(m*K)/W]} = \frac{0.15}{2.0} = 0.075\ (m^2*K)/W$$

$$R_{eps} = \frac{d\ [m]}{\lambda\ [(m*K)/W]} = \frac{0.250}{0.0348} = 7.1839\ (m^2*K)/W$$

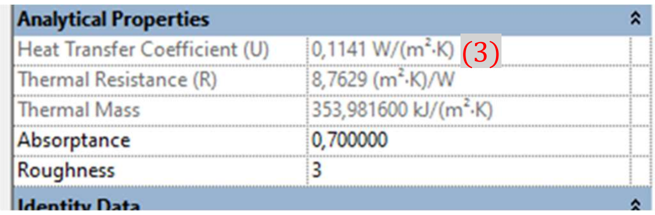
$$R = R_{PVC} + Padh + R_{Scr} + R_{eps} + R_C + R_{eps} =$$

$$= 0.012 + 0.0017 + 0.0535 + 1.4368 + 0.75 + 7.1839 = 8.7629\ (m^2*K)/W$$

$$R_{si} = 0.17\ (m^2*K)/W$$

$$R_{se} = 0.04\ (m^2*K)/W$$

$$U_{ČSN} = \frac{1}{R_{si} + R + R_{se}} = \frac{1}{0.17 + 8.7629 + 0.04} = \mathbf{0.1114\ W/(m^2*K)}$$



Analytical Properties	
Heat Transfer Coefficient (U)	0,1141 W/(m²·K) (3)
Thermal Resistance (R)	8,7629 (m²·K)/W
Thermal Mass	353,981600 kJ/(m²·K)
Absorptance	0,700000
Roughness	3
Identity Data	

Figure 40 - Thermal resistance of the bay floor.

3) - $U_{Re} = \mathbf{0.1141\ W/(m^2*K)}$

R1 – Sloped roof

ČSN U-Value calculation

$$R_{Hy} = \frac{d\ [m]}{\lambda\ \left[\frac{m*K}{W}\right]} = \frac{0.01}{1.2} = 0.0120\frac{m^2*K}{W}$$

$$R_{mia} = \frac{d\ [m]}{\lambda\ [(m*K)/W]} = \frac{0.300}{0.0348} = 1.4368\ (m^2*K)/W$$

$$R_{Hy} = \frac{d\ [m]}{\lambda\ [(m*K)/W]} = \frac{0.005}{1.2} = 0.075\ (m^2*K)/W$$

$$R_{eps} = \frac{d\ [m]}{\lambda\ [(m*K)/W]} = \frac{0.250}{2.000} = 7.1839\ (m^2*K)/W$$

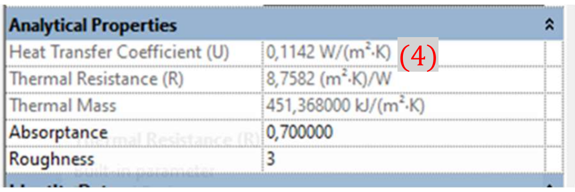
$$R = R_{Hy} + R_{mia} + R_{Hy} + H_{eps} =$$

$$= 0.0083 + 8.6207 + 0.0042 + 0.1250 = 8.7582\ (m^2*K)/W$$

$$R_{si} = 0.10\ (m^2*K)/W$$

$$R_{se} = 0.10\ (m^2*K)/W$$

$$U_{ČSN} = \frac{1}{R_{si} + R + R_{se}} = \frac{1}{0.10 + 8.7629 + 0.10} = \mathbf{0.1116\ W/(m^2*K)}$$



Analytical Properties	
Heat Transfer Coefficient (U)	0,1142 W/(m^2·K) (4)
Thermal Resistance (R)	8,7582 (m^2·K)/W
Thermal Mass	451,368000 kJ/(m^2·K)
Absorptance	0,700000
Roughness	3

Figure 41 - Thermal resistance of the sloped roof.

$$4)\text{ - } U_{Re} = \mathbf{0.1142\ W/(m^2 * K)}$$

<u>R2 – Flat roof</u>

ČSN U-Value calculation

$$R_{Hy} = \frac{d\ [m]}{\lambda\ \left[\frac{m * K}{W}\right]} = \frac{0.01}{1.2} = 0.0120 \frac{m^2 * K}{W}$$

$$R_{con} = \frac{d\ [m]}{\lambda\ [(m * K)/W]} = \frac{0.300}{0.0348} = 1.4368\ (m^2 * K)/W$$

$$R_{Hy} = \frac{d\ [m]}{\lambda\ [(m * K)/W]} = \frac{0.005}{1.2} = 0.075\ (m^2 * K)/W$$

$$R_{eps} = \frac{d\ [m]}{\lambda\ [(m * K)/W]} = \frac{0.250}{2.000} = 7.1839\ (m^2 * K)/W$$

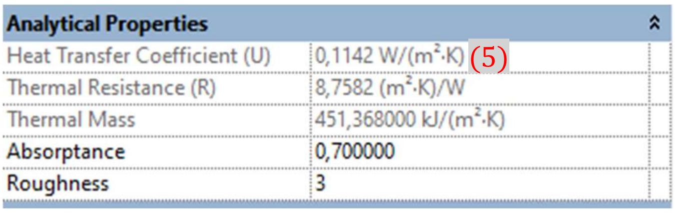
$$R = R_{Hy} + R_{con} + R_{Hy} + R_{eps} =$$

$$= 0.0083 + 8.6207 + 0.0042 + 0.1250 = 8.7582\ (m^2 * K)/W$$

$$R_{si} = 0.10\ (m^2 * K)/W$$

$$R_{se} = 0.04\ (m^2 * K)/W$$

$$U_{ČSN} = \frac{1}{R_{si} + R + R_{se}} = \frac{1}{0.10 + 8.7629 + 0.04} = \mathbf{0.1124\ W/(m^2 * K)}$$



Analytical Properties	
Heat Transfer Coefficient (U)	0,1142 W/(m^2·K) (5)
Thermal Resistance (R)	8,7582 (m^2·K)/W
Thermal Mass	451,368000 kJ/(m^2·K)
Absorptance	0,700000
Roughness	3

Figure 42 - Thermal resistance of the flat roof.

$$5)\text{ - } U_{Re} = \mathbf{0.1142\ W/(m^2 * K)}$$

2.3) Summary of results

As you can see in the calculations, Revit results have fatal mistake, and engineer can´t rely on them. My plugin will make the calculations according to ČSN, it will also do average values of each element, average U-Value of all elements and compare everything to ČSN.

3) Preparing system architecture by making Dynamo script

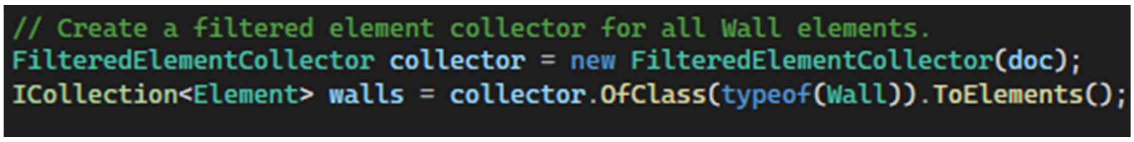
In this chapter I will tell you what Autodesk Dynamo is, why I have chosen it to try to create first plugin version, what issues I have faced, because some of them have defined the final system architecture and, step by step how the final script have looked and some summary of the Dynamo.

3.1) Introduction of Autodesk Dynamo

Autodesk Dynamo is tool, that comes with Revit, but it is plugin that works with Revit and can be useful from easy tasks like spin all columns around Z – axis for 30°, which is impossible directly in Revit.

Dynamo is visual programming tool.

It means, that when you for example want to get list of all walls, in C# you must make plugin connected to Revit by multiple steps and writing the code, that will look like:



```
// Create a filtered element collector for all Wall elements.
FilteredElementCollector collector = new FilteredElementCollector(doc);
ICollection<Element> walls = collector.OfClass(typeof(Wall)).ToElements();
```

Figure 43 - Wall elements collector in C#.

You can just start Dynamo by few clicks directly from Revit 1), 2).

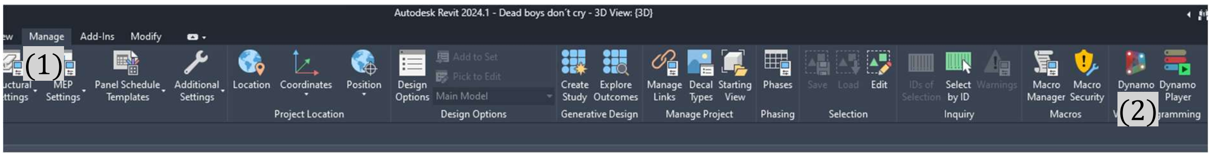


Figure 44 - Starting Dynamo.

Create new script, find 2 nods, just connect them and click on run 1):

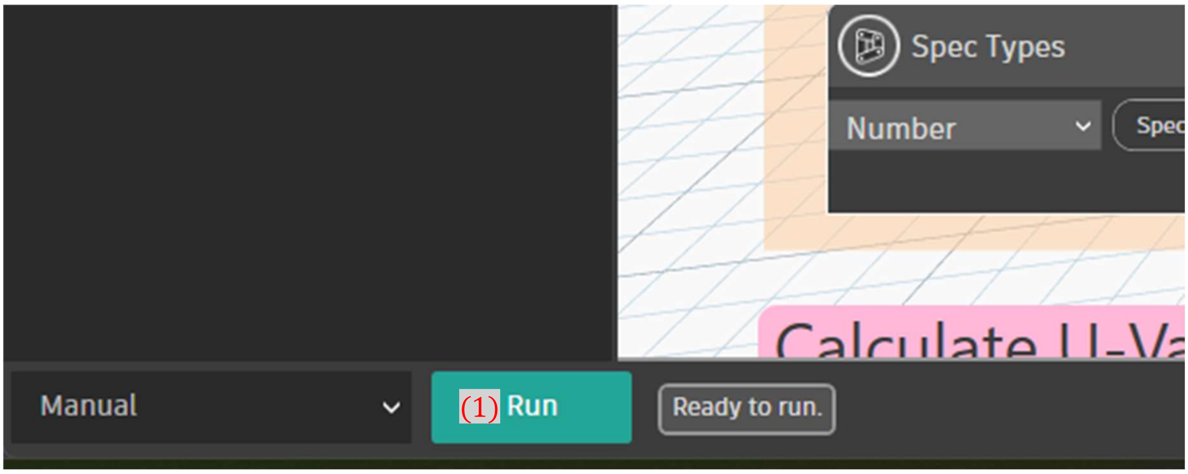


Figure 45 - Nods for list of wall elements.

Nods are these small windows, there are tons of Nods in Dynamo, so you can make pretty much in Dynamo. Nods can do things like collecting elements, do some number calculations, move elements in Revit, create Elements.

Everything is categorized in list of nods on the left side of Dynamo. You can also search directly for the nod if you know or can guess what the nod name is.

Reasons to use Autodesk Dynamo

I have never tried to code, so since Dynamo is visual programming language, it is a lot easier to learn how it works when you just search for windows and connect them.

The architecture of the system is simpler, because lot of things are created, you just must find the right nod, instead of creating dozens of code lines, so when it doesn´t work, you know that there is mistake only in your system logic, not in the functions.

Since Dynamo is part of Revit, you don´t need to learn how to connect it to Revit, which take some time and you can make mistake and it will not work.

Also, when I have finished my Dynamo script, I knew how the program would work and I had space to focus mainly on how to write it.

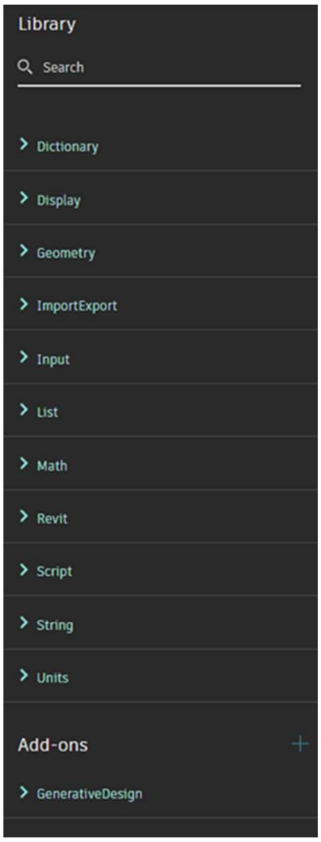
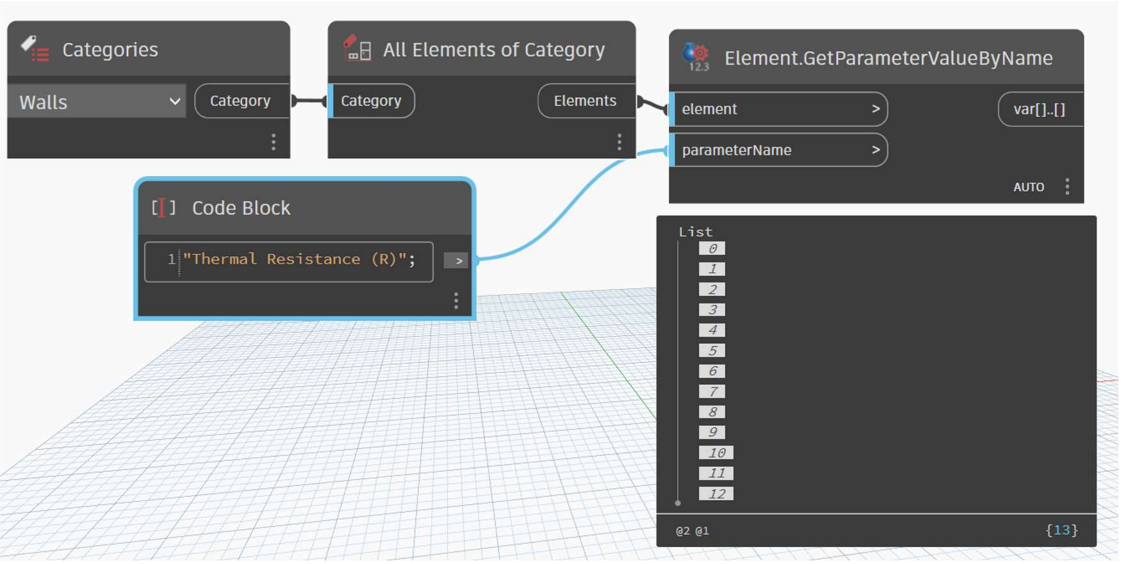


Figure 46 - Categories of nods.

3.2) Issues with getting parameters.

First step in my plugin was to collect some parameters from my elements in Revit, I have decided to start with walls.

As I have shown you before, collect all wall elements is pretty easy, but getting the parameters wasn´t as easy as it has looked. I have written name of desired parameter in code block, connected both to nod, that is supposed to get you parameter value by connecting with the element list and parameter names. My result was blank list of parameters.

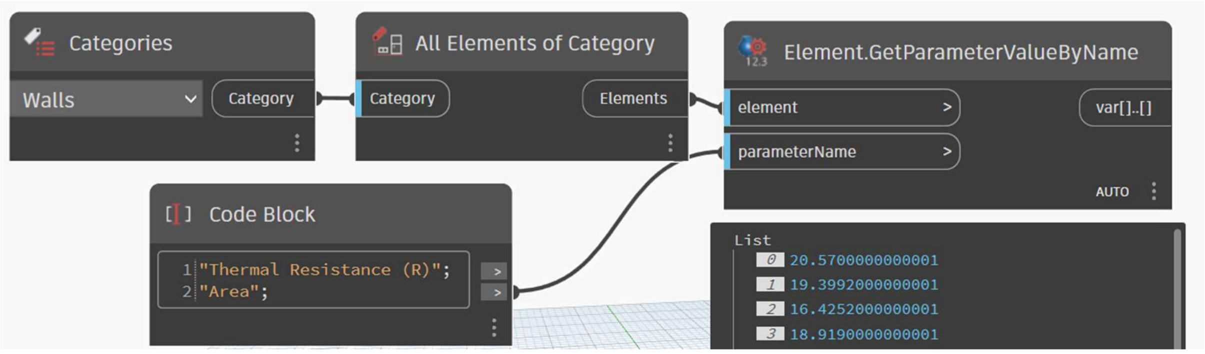


```
"Thermal Resistance (R)";
```

```
List
  0
  1
  2
  3
  4
  5
  6
  7
  8
  9
  10
  11
  12
@2 @1                                   {13}
```

Figure 47 - Blank parameters.

I have tried different parameters and some of them have worked properly.



```
"Thermal Resistance (R)";
"Area";
```

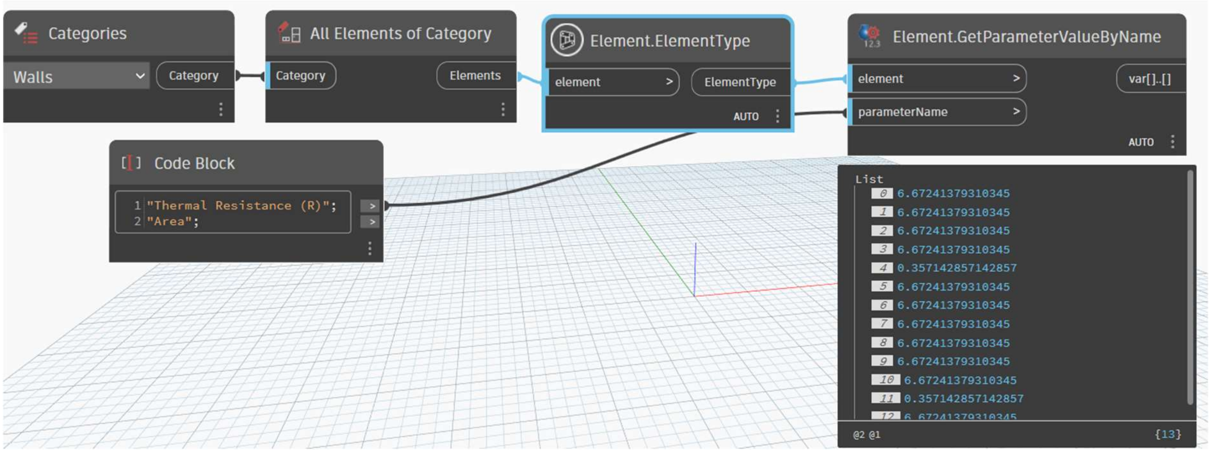
```
List
  0 20.5700000000001
  1 19.3992000000001
  2 16.4252000000001
  3 18.9190000000001
```

Figure 48 - Parameter area.

It has taken me some time to find out why it doesn´t work.

The reason is that in Revit, there are multiple types of parameters:

- Instance parameters – Parameters bounded right into element like Area, Width etc.
    - Can be searched this easier way.
- Shared parameters – Will be discussed later.
- Family parameters – Parameters that are bounded to family, for example some families of columns will have just width and height, but when you have family for I profiles, there can be parameters like width of flange.
- Type parameters – Parameters bounded to element type (walls with same structure, for example all walls with bricks 300 mm and EPS 180 mm, have these parameters same). It is parameters like Thermal Resistance (R) or Revit´s indicative U-Value.
    - You have to reach element type instead of just element.



```
"Thermal Resistance (R)";
"Area";
```

```
List
  0 6.67241379310345
  1 6.67241379310345
  2 6.67241379310345
  3 6.67241379310345
  4 0.357142857142857
  5 6.67241379310345
  6 6.67241379310345
  7 6.67241379310345
  8 6.67241379310345
  9 6.67241379310345
  10 6.67241379310345
  11 0.357142857142857
  12 6.67241379310345
@2 @1                                {13}
```

Figure 49 - Working nod script to reach type parameters.

I had also a little problem with this in C#, but if I didn´t know it from here, the time invested in this problem would be much bigger.

3.3) Filter of elements.

Naturally I have wanted my plugin to be as automatic as it can be.

First idea of filter of elements to include in my calculations was to tell the user to make an Area, that will declare what is interior and just exclude elements, that are inside the area.

There were multiple problems with this solution.

I had to create element out of an area, but the Dynamo was able to operate only with area boundary. Sadly, Area boundary is not one curve, but multiple lines, so I have needed to connect them into one curve.

First problem was that when you let Revit create Area by exterior walls, the area boundary lines 1) do not end in the same points 2), so the Dynamo couldn´t make one curve from it.

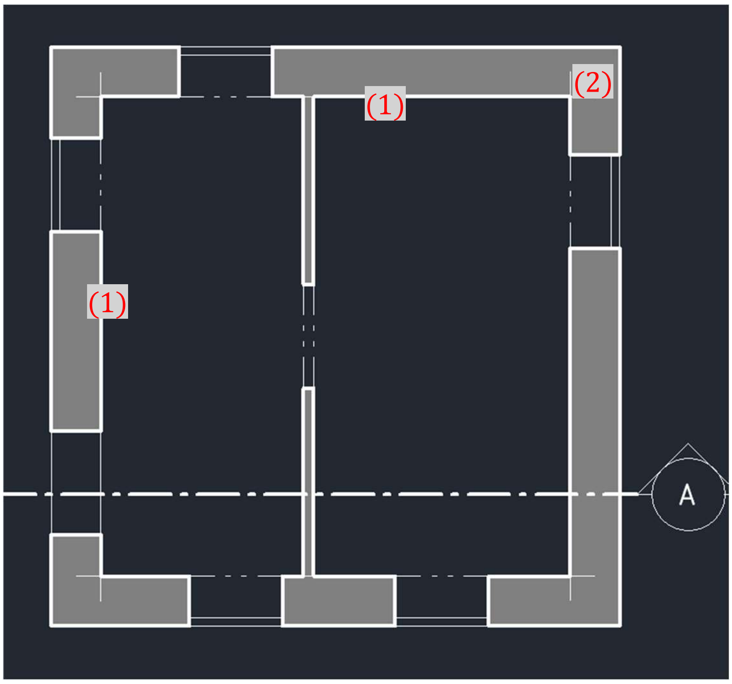
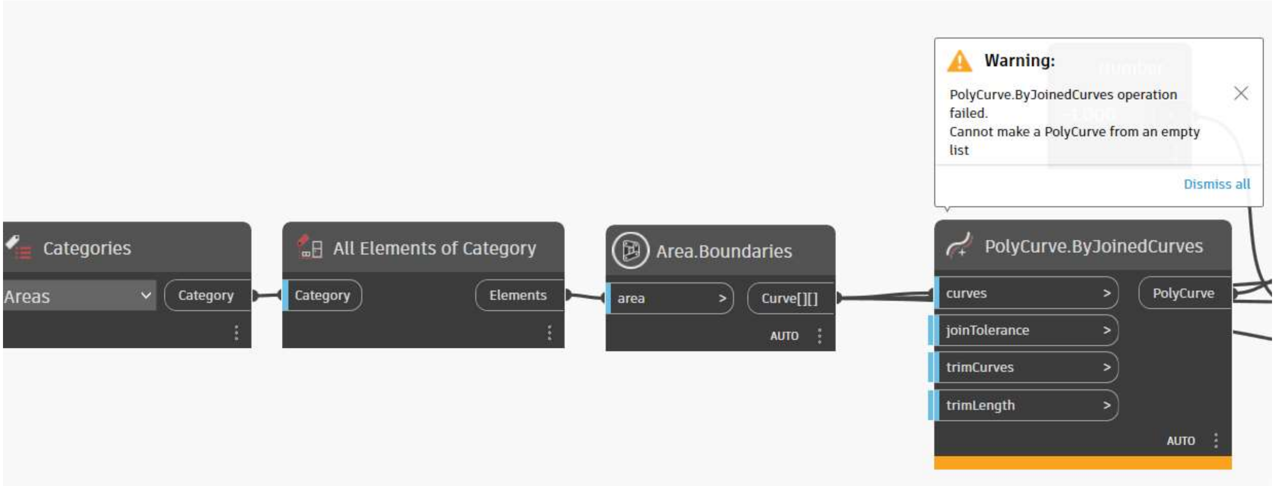


Figure 50 - Area lines.



Warning:
PolyCurve.ByJoinedCurves operation failed.
Cannot make a PolyCurve from an empty list

Figure 51 - Dynamo Error.

Since I have discovered it, this solution was not going to be fully automatic, because I would have to tell the user to manually edit Area and make the lines end in the same points.

It wouldn´t be that terrible solution, but it has already lost a lot of its advantages.

In the next step I have set the curve offset (1) to -1 (2) to be sure, that it will not touch the exterior walls and I have extruded (3) curve to the height of the floor (4). Also made solid from it (5).

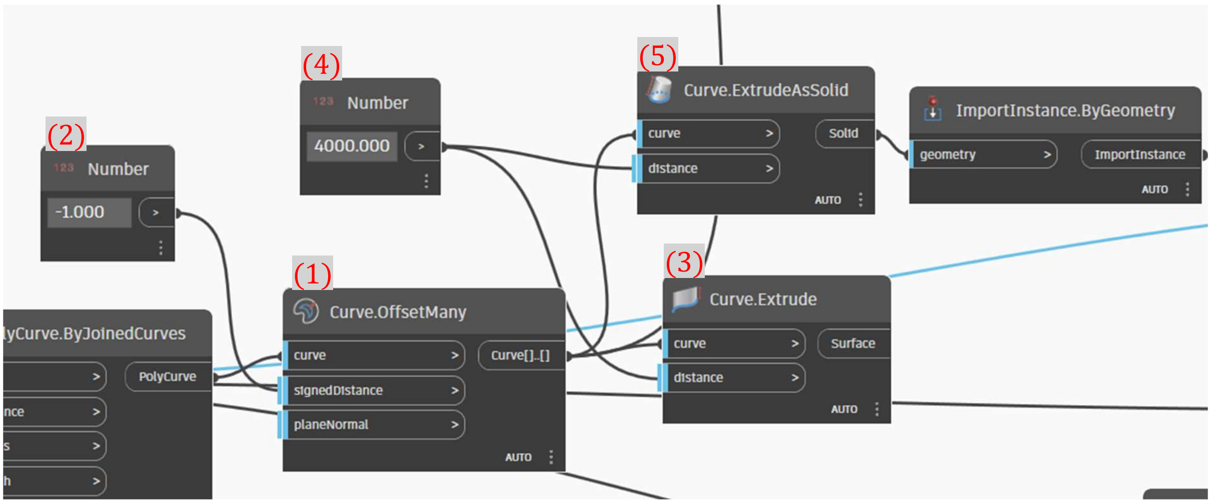
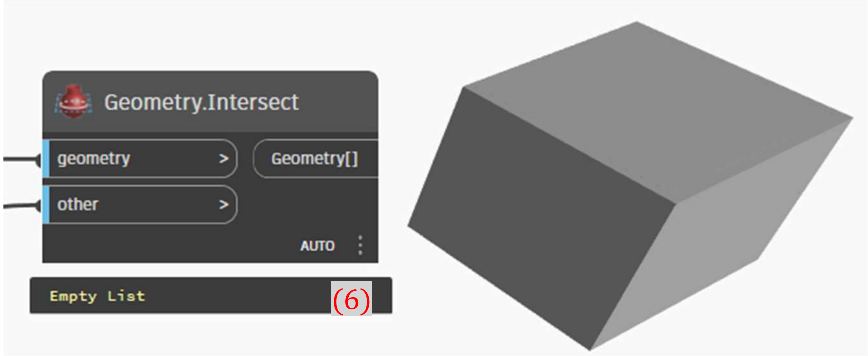


Figure 52 - Making solid from the curve.

It has created the solid, but when I have connected it with the list of my walls into intersection detector, the list was empty (6).



```
Empty List
```

Figure 53 - Geometry intersect - empty list.

The reason was that this node can´t recognize if element and solid intersects, because they are different object.

I have tried many other solutions, like creating a mass instead of solid, because it is element, that would be created even in Revit (solid wouldn´t).

I have spent about 2.5 workdays with my tries to make this method work, my most successful result was showing me that yes there are walls that intersects with the area.

After this long time of unsuccessful tries I have made new concept of element filter.

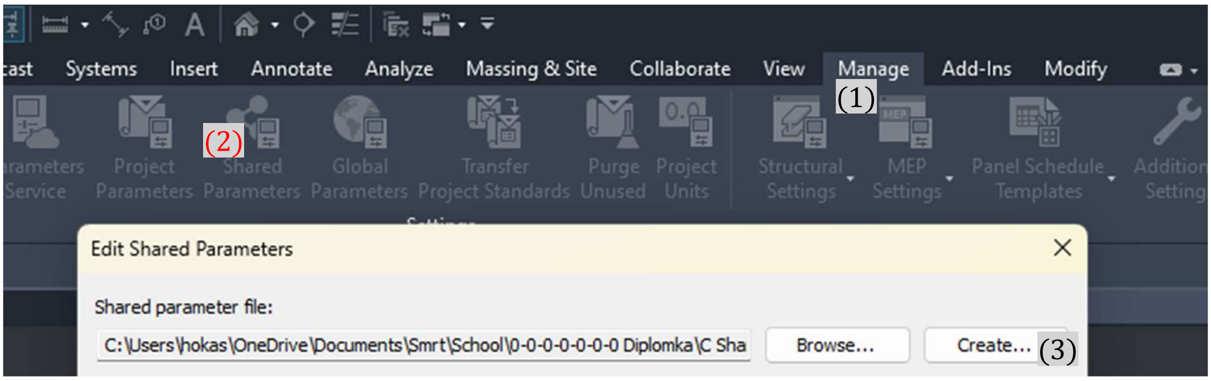
I will just add new parameter "COVER" into project and user will have define if the element is part of the building cover by inserting a value 1 for the elements that are part the cover.

3.4) Inserting parameters

As I have promised, now is the right time to tell you about shared parameters.

Shared parameters are your custom parameters, they are stored in .txt file and before you can work with them, you must import existing file or create new.

You can add them to elements and make types of values like numbers or text.



Edit Shared Parameters

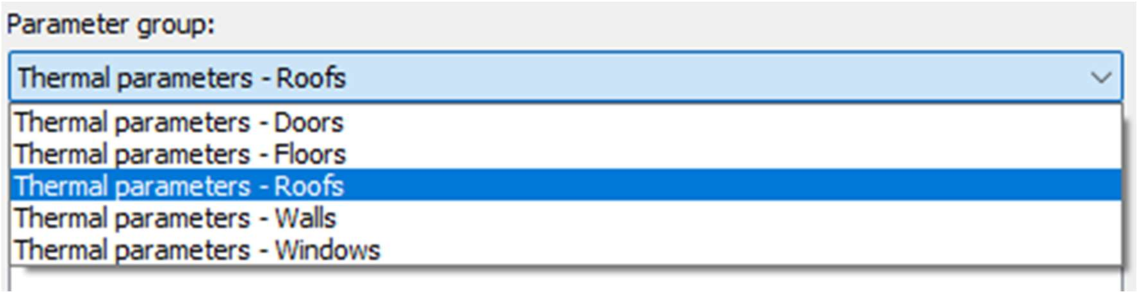
Shared parameter file: C:\Users\hokas\OneDrive\Documents\Smrt\School\0-0-0-0-0-0 Diplomka\C Sha

Figure 54 - Creating shared parameters file.

That was all I did in Revit before I can run my Dynamo script, the rest can be also created manually, but in my case, I made the Dynamo to do it.

When you create the file, you need create parameters group.

I have created group for each element that will be included in calculation.



Parameter group: Thermal parameters - Roofs

Thermal parameters - Doors
Thermal parameters - Floors
Thermal parameters - Roofs
Thermal parameters - Walls
Thermal parameters - Windows

Figure 55 - Parameter groups.

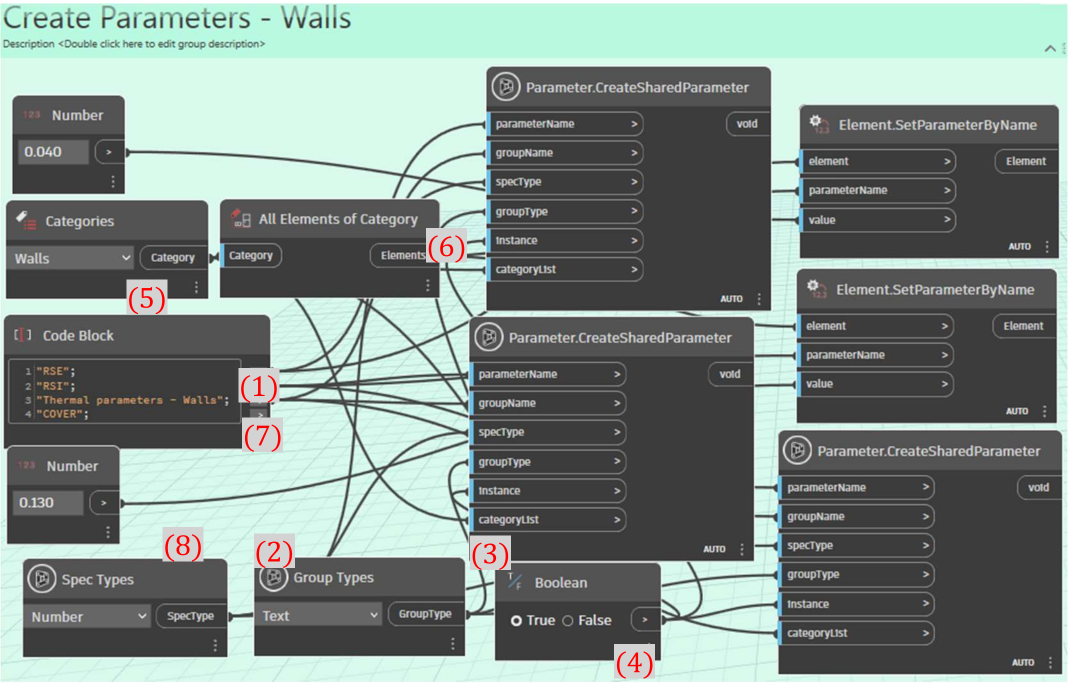
When you are running script for the first time in the project, you need to run it twice in order to make parameters work – first run creates parameter groups and parameters, the second will add values to them.

In the parameter creation, you need to add to input:

1) Parameter name, group
2) spec type (number, material etc)
3) group type (something like group of parameters, sorts them in the place where are they displayed in the Revit)
4) instance (true = parameter will be reachable as element parameter, false = parameter will be reached as type parameter)
5) category list (category of elements to which will be parameter added).

When you want to add value to a parameter, you need to input:

6) List of elements
7) Name of parameter
8) Value of a parameter



Create Parameters - Walls

Description <Double click here to edit group description>

```
"RSE";
"RSI";
"Thermal parameters - Walls";
"COVER";
```

Figure 56 – creating wall parameters.

3.5) Summary of the Dynamo script

To get the results, you need to run the script twice.

First run will make parameters in the project.

;;Then you need to go back to your model, set exterior construction value of parameter “COVER” to 1.

Set your U-Values of windows and doors.

Run the script for the second time, it will add R_{si} and R_{se} parameter values and calculate your results.

This is the view on whole script, it is made from six parts.

1) Wall calculations
2) Roof calculations
3) Floor calculations
4) Windows calculations
5) Door calculations
6) Average calculations

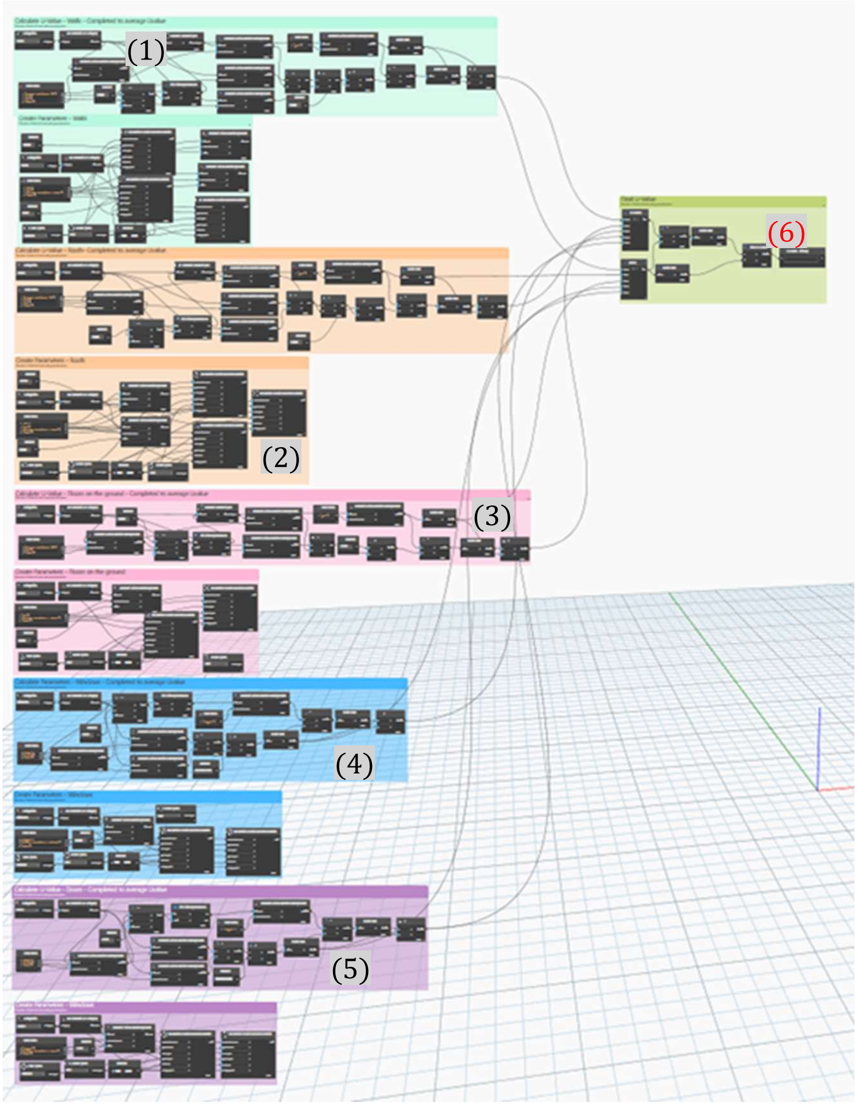


Figure 57 - Whole dynamo script.

Wall calculations step by step

Calculations for all elements are similar, so to save you time and my nerves while writing this work, I will describe only calculations for the walls.

Wall calculations are made from two parts:

1) Calculation part (will be explained soon)
2) Create parameters – explained in the previous chapter.

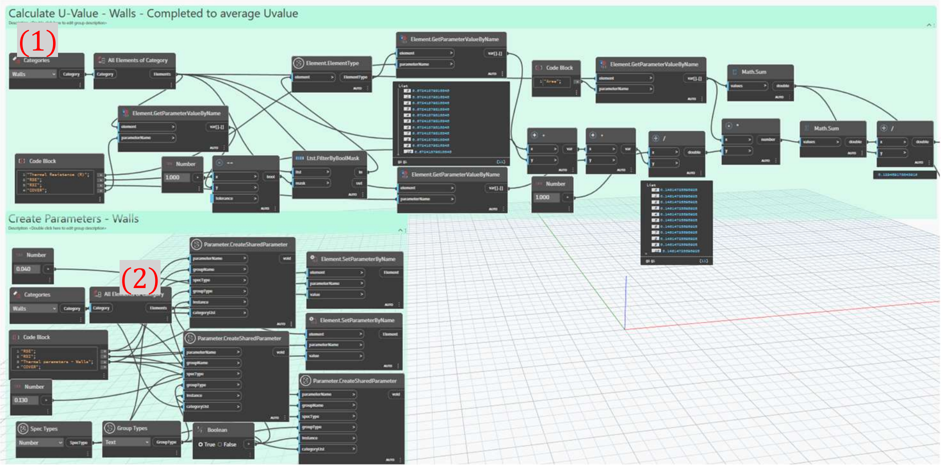
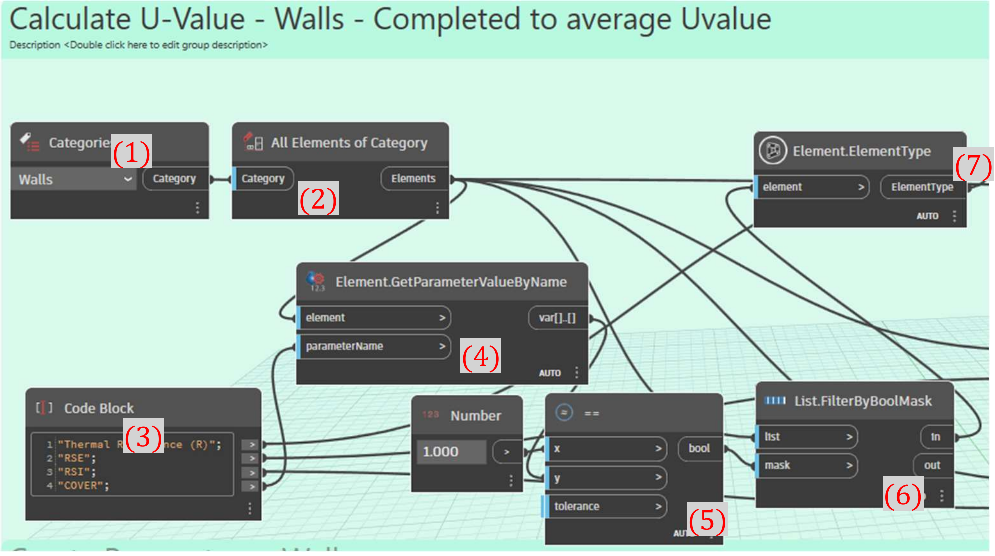

Calculate U-Value - Walls - Completed to average Uvalue
(1)
Categories
All Elements of Category
Element.ElementType
Element.GetParameterValueByName
Code Block
Math.Sum
List.FilterByBoolMask
Number
Create Parameters - Walls
(2)
Parameter.CreateSharedParameter
Element.SetParameterValueByName
Spec Types
Group Types
Boolean

Figure 58 - Structure of the wall calculator.

First part of the script is wall collector.

1) Collector that selects all walls
2) Nod that makes accessible instance parameters
3) Names of parameters we will need in this part of code.
4) This nod gets value of parameter "COVER".
5) This nod will declare that all walls that user marked that they are exterior (by setting parameter COVER to value 1), have now value "true".
6) Next step is filter that will add to list of calculated walls only the ones that are marked true from the == nod.
7) This nod just adds possibility to access type parameters.

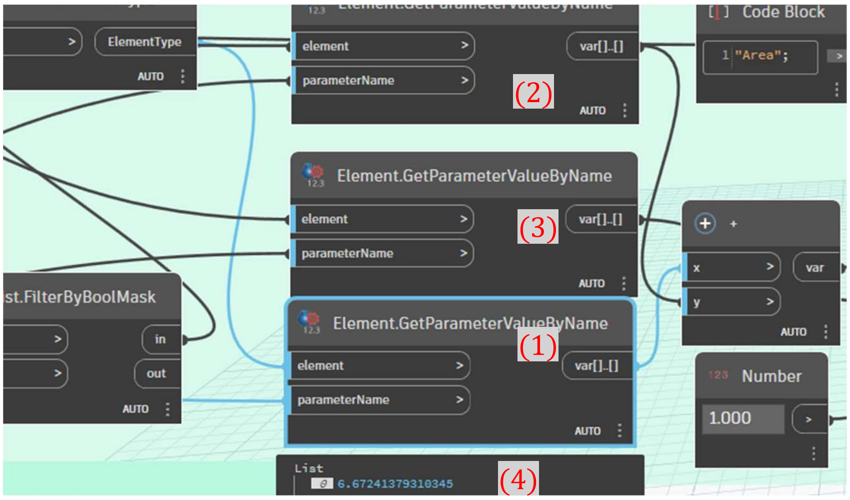


```
"Thermal R(3)nce (R)";
"RSE";
"RSI";
"COVER";
```

Figure 59 - First part of the wall calculator.

In the next part we access our parameters, Thermal Resistance (R) (1), R_{si} (2) and R_{se} (3).

We know that R of our walls is $R = 6.6724\ (m^2 * K)/W$, as you can see (4), Dynamo can access more decimals, than Revit shows us.



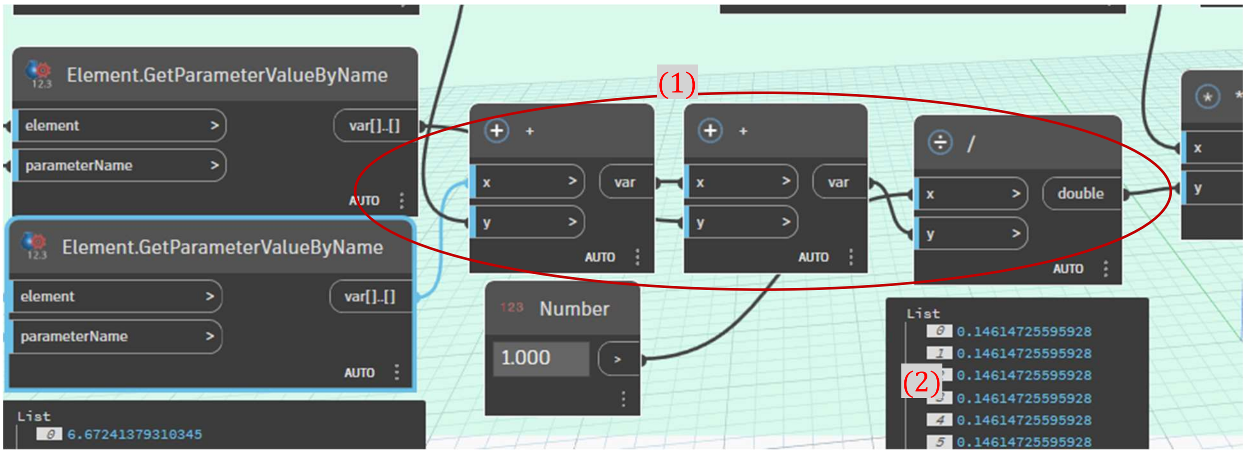
```
List
 0 6.67241379310345
```

Figure 60 - Searched parameters.

In this part we just take the parameters and do with them this calculation.

$$U_{ČSN} = \frac{1}{R_{si}+R+R_{se}} = \frac{1}{0.13+6.672\quad .04} = \mathbf{0.1461\ W/(m^2 * K)}\ (1)$$

And as you can see, we have got the same numbers as in manual calculation (2).



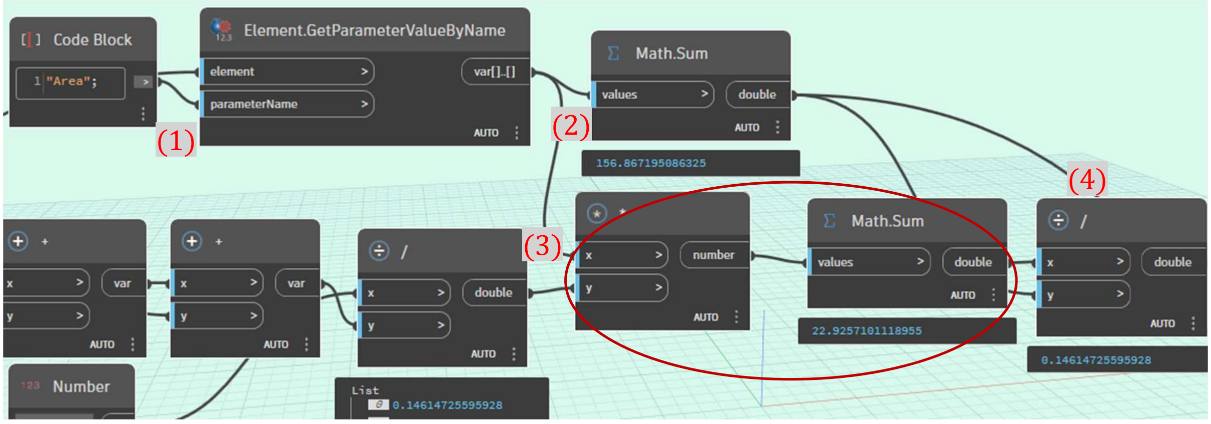
```
List
 0 6.67241379310345
```

```
List
 0 0.14614725595928
 1 0.14614725595928
 2 0.14614725595928
 3 0.14614725595928
 4 0.14614725595928
 5 0.14614725595928
```

Figure 61 - Calculations.

In the last part of the wall calculations the script:

1) Access Areas of walls
2) Makes sum of all areas
3) Makes sum of Areas * U-Values
4) Tells us weighted average of all wall U-Values



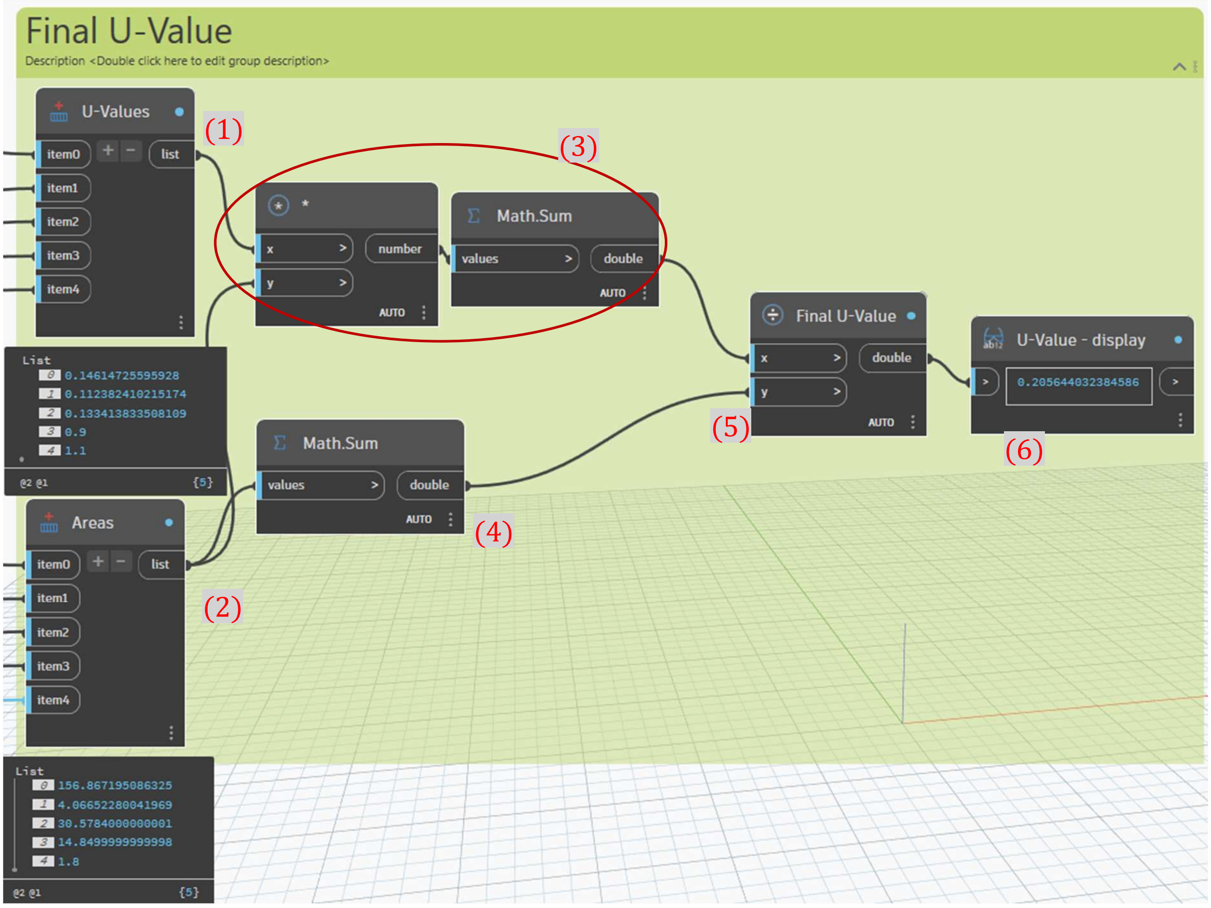
```
List
 0 0.14614725595928
```

Figure 62 - Average wall calculations.

Final part – Average U-Value

In this part the script just calculates weighted average U-Value of all counted elements.

1) This nod makes list of average U-Values of our counted elements.
2) This nod makes list of areas of counted elements.
3) These two nods make sum of area * U-Value.
4) This nod makes sum of all areas.
5) This nod makes final average U-Value.
6) This nod is just to display the U-Value.



```
List
 0 0.14614725595928
 1 0.112382410215174
 2 0.133413833508109
 3 0.9
 4 1.1
@2 @1                {5}
```

```
List
 0 156.867195086325
 1 4.06652280041969
 2 30.5784000000001
 3 14.8499999999998
 4 1.8
@2 @1                {5}
```

Figure 63 - Average U-Value script.

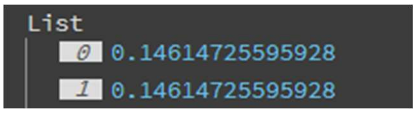
3.7) Summary of Dynamo results

Dynamo script had to be simplified, so I wasn´t including all calculations. I have decided to include only one type for each element. So, there is no calculation for sloped roof (because it is double shelled, so there will be different R_{se}, than in single shelled flat roof), also calculation for bay floor is not included.

ČSN Calculation of the wall

$$U_{ČSN} = \mathbf{0.1461}\ \boldsymbol{W/(m^2 * K)}$$

Dynamo result is same as manual calculation according to ČSN.



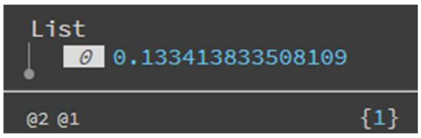
```
List
  0 0.14614725595928
  1 0.14614725595928
```

Figure 64 - Wall calculation results.

ČSN Calculation of floor on the ground.

$$U_{ČSN} = \mathbf{0.1334}\ \boldsymbol{W/(m^2 * K)}$$

Dynamo result is same as manual calculation according to ČSN.



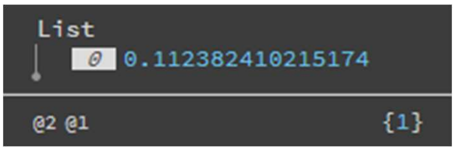
```
List
  0 0.133413833508109
@2 @1                {1}
```

Figure 65 - Floor calculation result.

ČSN Calculation of the flat roof

$$U_{ČSN} = \mathbf{0.1124}\ \boldsymbol{W/(m^2 * K)}$$

Dynamo result is same as manual calculation according to ČSN.



```
List
  0 0.112382410215174
@2 @1                {1}
```

Figure 66 - Roof calculation result.

Tab. 2 – Summary of dynamo script results.

Element	ČSN U-Value [W/(m²*K)]	Dynamo U-Value [W/(m²*K)]	Value fits
Wall	0.1461	0.1461	OK
Floor	0.1334	0.1334	OK
Roof	0.1124	0.1124	OK

3.8) Evaluation of the Dynamo script

All of Dynamo results has corresponded to manual calculations.

I have made simplified working architecture of my script. I have learned how to “think as coder” so my process of programming in C# was a lot easier, than if I went straight to C#. I think even whole process of Dynamo + C# was faster, than if I have made just the C# plugin.

Also, even when in Dynamo it wasn´t that useful I have learned a bit how to work with ChatGPT which was extremely useful tool while I was working in C#.

4) Creating the add-in in the coding language C#

In this chapter, I will tell you why I have chosen to write the code in C# and what were the other options. How I have learned to code and how helpful was AI while learning process. Then I will show you step by step my code, I will show you chronologically how the code works and for the most of parts I will tell you about the obstacles I have faced trough their development. In the end of the chapter, I will write some summary of the results.

4.1) Why C#?

I was trying to decide between Python and C#, because Python have simpler syntax and is easier to learn than C#, but this is what Autodesk says about creating Revit plugins:

“Autodesk Revit has a .NET API which means you can use any of the .NET compliant programming languages (C#, VB.NET, F#, etc.) to develop a plug-in. While each language has its own relative benefits, C# is the natural choice for this guide: it is easy-to-learn, easy-to-use and leverages the power of the underlying .NET Framework. As you become comfortable with the content of this guide - and more proficient with C# - you’ll be able to move on to solving more complex problems with the language.” [2]

So, I have picked C# and as I can say after finishing the job, it was good choice.

4.3) Script step by step

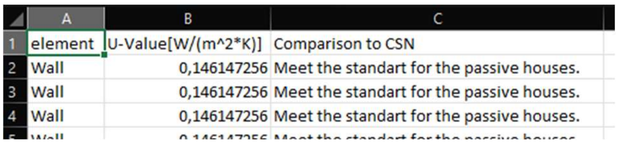
In this chapter I will guide you, trough whole my code Chronologically of how it is processed (not how it was made). I will explain my program logic and in lot of parts I will tell you about obstacles I have faced trough the coding process.

Since I have never tried to code before, I have spent about three days just looking at videos about C# basics, what basic functions are, how variables like strings and doubles works, but it would be boring, so I will save you from it and I will skip to the complete code part.

Exporter Method

This is part of code, stored in the different place, reason to use it like this and not just put everything in one place is because this thing makes code clearer to read and debug, but harder to make the program logic. If I was real coder, I would split the huge code into parts like this and in the main part just call them, but it requires experience and since I have worked in C# just about a one month, it was impossible for me to learn how to make whole code this way, so I have just tried it once, to get at least a bit how it works.

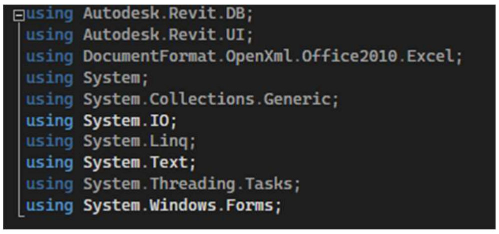
I have made an exporter, that will make table in the format .CSV, which will natively open in Excel. This format is text, where columns are separated by “, symbol”, so I just made String builder, that store in every line: Name of element, Element´s U-Value, Comparison to ČSN. When I open it in Excel, it looks like this.



	A	B	C
1	element	U-Value[W/(m^2*K)]	Comparison to CSN
2	Wall	0,146147256	Meet the standart for the passive houses.
3	Wall	0,146147256	Meet the standart for the passive houses.
4	Wall	0,146147256	Meet the standart for the passive houses.
5	Wall	0,146147256	Meet the standart for the passive houses.

Figure 67 - Example of exported table

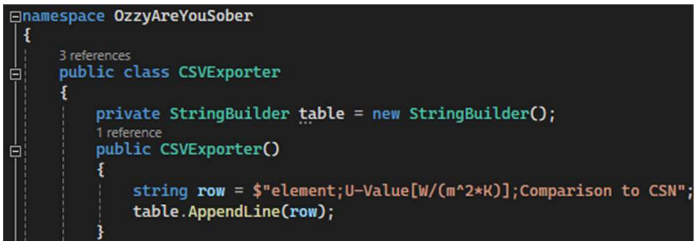
Firstly, I need to call every method that is included in Visual studio or for example RevitAPI, which is programme, that allows us to reach Revit things like elements.



```
using Autodesk.Revit.DB;
using Autodesk.Revit.UI;
using DocumentFormat.OpenXml.Office2010.Excel;
using System;
using System.Collections.Generic;
using System.IO;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using System.Windows.Forms;
```

Figure 68 - Calling systems.

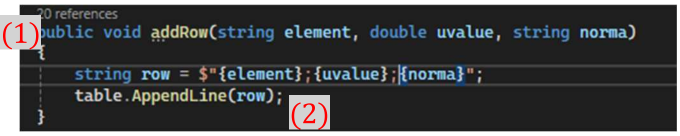
Then I have made method, that can be called on the start of my main method, it is made from 2 parts.



```
namespace OzzyAreYouSober
{
    3 references
    public class CSVExporter
    {
        private StringBuilder table = new StringBuilder();
        1 reference
        public CSVExporter()
        {
            string row = $"element;U-Value[W/(m^2*K)];Comparison to CSN";
            table.AppendLine(row);
        }
```

Figure 69 - First part - header of the table.

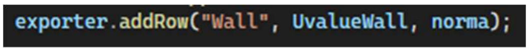
Second part creates a function (1), in which I will insert parameters (2) to store them in the table.



```
20 references
public void addRow(string element, double uvalue, string norma)
{
    string row = $"{element};{uvalue};{norma}";
    table.AppendLine(row);
}
```

Figure 70 - Second part.

In the main code I will just write a line, that will add row to table every time I need it.



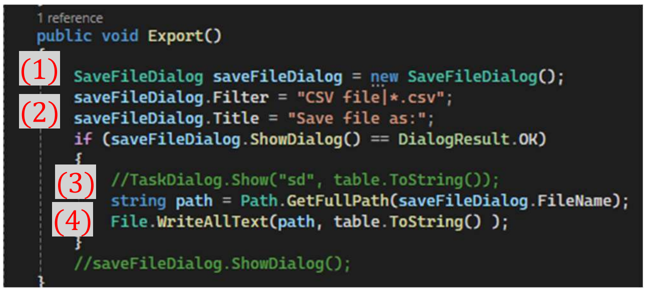
```
exporter.addRow("Wall", UvalueWall, norma);
```

Figure 71 - Adding rows in the main code.

UvalueWall and norma are local variables where are stored values for currently calculated wall.

Second part of this method is there to export my table and make pop up the "save file as" window.

1) This part tells the program to save file as .csv.
2) Title of pop-up window
3) Saving the file
4) Writing the text in it.

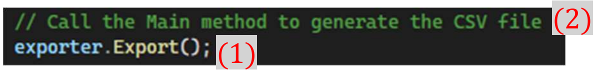


```
1 reference
public void Export()
{
    SaveFileDialog saveFileDialog = new SaveFileDialog();
    saveFileDialog.Filter = "CSV file|*.csv";
    saveFileDialog.Title = "Save file as:";
    if (saveFileDialog.ShowDialog() == DialogResult.OK)
    {
        //TaskDialog.Show("sd", table.ToString());
        string path = Path.GetFullPath(saveFileDialog.FileName);
        File.WriteAllText(path, table.ToString() );
    }
    //saveFileDialog.ShowDialog();
}
```

Figure 72 - Save file as.

In the end of my main method, I just call this method, to make pop up the save as window (1).

The green lines after "//" are just labels for better orientation on the code, the program is not doing anything with them (2).



```
// Call the Main method to generate the CSV file
exporter.Export();
```

Figure 73 - Line to call "save as" method.

Obstacles I have faced in this part:

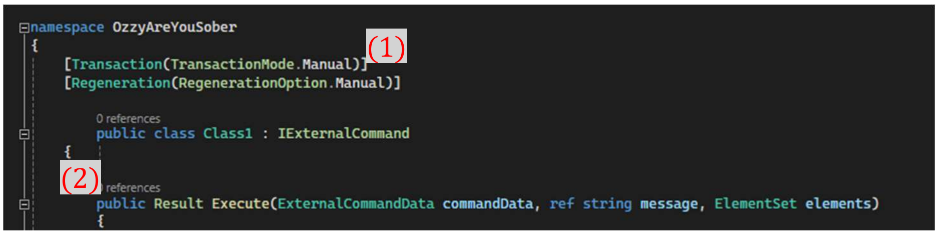
Only mistake I have made in this part was that I was trying to work with Visual Studio´s plugin XMLExporter, which is free plugin, that can export the .XLSX files. It was wrong choice, because it was far more complicated to work with, than with this solution. Also using plugin when it is not required is needles complicating the work.

Only one thing that would be better in XMLExporter is that it would be directly in .XLSX format, now when I have .CSV and I want to make some changes in the file with results, Excel will require to save it as new file, but it is small problem and in my opinion even better, because you can´t accidentally overwrite your updated file, because it will be in different format.

Beginning of the main code

Now I am going to show you the main part of my code. Before the actual code starts, there is the calling systems part, but it is needles to show it to you again, it is same as in exporter method, but there are a lot more lines.

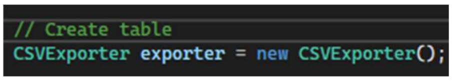
This is just start of the code, setting the used functions options (1) and the part where the code says: "There starts the main code." (2)



```
namespace OzzyAreYouSober
{
    [Transaction(TransactionMode.Manual)]
    [Regeneration(RegenerationOption.Manual)]

        0 references
        public class Class1 : IExternalCommand
    {

          references
        public Result Execute(ExternalCommandData commandData, ref string message, ElementSet elements)
        {
```

Figure 74 - Code intro.

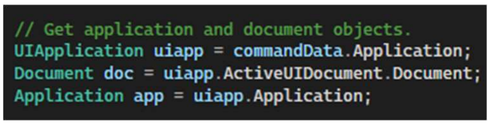
Then I call exporter method, which creates heading for my export table and allows me to add rows in the table.



```
// Create table
CSVExporter exporter = new CSVExporter();
```

Figure 75 - Calling the table method.

These lines say something like: "You are working with Revit, and you are working with project that is now opened and you are working with the objects inside".



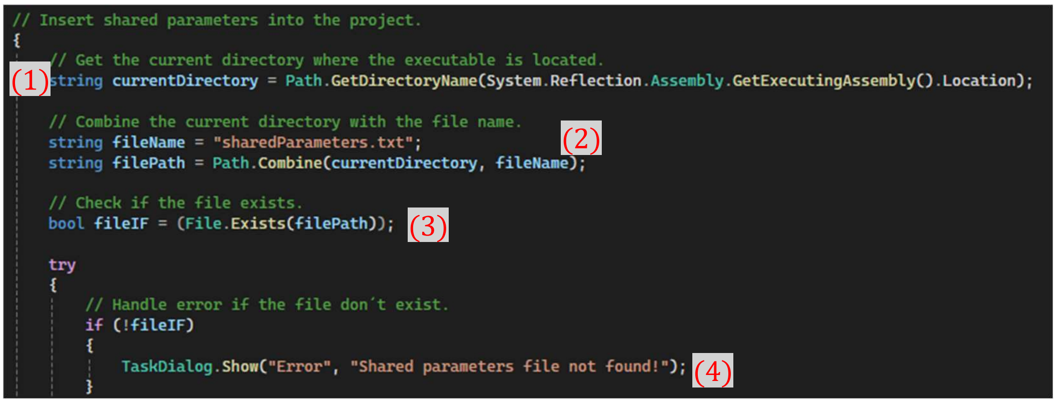
```
// Get application and document objects.
UIApplication uiapp = commandData.Application;
Document doc = uiapp.ActiveUIDocument.Document;
Application app = uiapp.Application;
```

Figure 76 - Connection to currently opened Revit session.

Assigning Shared parameters into Revit project

I have created a shared parameters .txt file with the parameters I need to, I have inserted it into my plugin files into the same directory, where the actual plugin is. My first error there was that I have inserted it into folder where the "Human readable" code is (visual studio file, containing text code), but it needs to be in the folder with the "computer readable file" is, it is generated by Visual Studio, and it contains only zeros and ones.

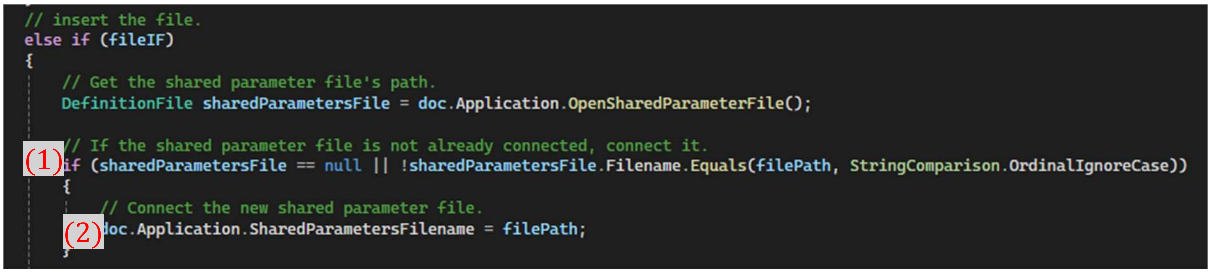
1) Get the folder path, where the files are.
2) Add name of file to the path.
3) Check if the file exists and show the error window if the file doesn´t exists (4).



```
// Insert shared parameters into the project.
{
    // Get the current directory where the executable is located.
    string currentDirectory = Path.GetDirectoryName(System.Reflection.Assembly.GetExecutingAssembly().Location);

    // Combine the current directory with the file name.
    string fileName = "sharedParameters.txt";
    string filePath = Path.Combine(currentDirectory, fileName);

    // Check if the file exists.
    bool fileIF = (File.Exists(filePath));

    try
    {
        // Handle error if the file don´t exist.
        if (!fileIF)
        {
            TaskDialog.Show("Error", "Shared parameters file not found!");
        }
```

Figure 77 - First part.

This part checks if my file isn´t already connected (1) and connect it (2).



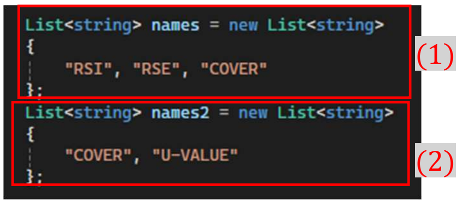
```
// insert the file.
else if (fileIF)
{
    // Get the shared parameter file's path.
    DefinitionFile sharedParametersFile = doc.Application.OpenSharedParameterFile();

    // If the shared parameter file is not already connected, connect it.
    if (sharedParametersFile == null || !sharedParametersFile.Filename.Equals(filePath, StringComparison.OrdinalIgnoreCase))
    {
        // Connect the new shared parameter file.
        doc.Application.SharedParametersFilename = filePath;
    }
```

Figure 78 - Assigning the file.

At this point I was thinking that I have already completed adding the parameters part, because everything was working well, but I was testing it on a project where I had already run my Dynamo script, when I have tested it on new Revit model, it wasn´t working. The reason was that it has connected the file, but not Assigned parameters into the elements, so I had to made whole new part.

I create lists of parameters I want to assign to elements:

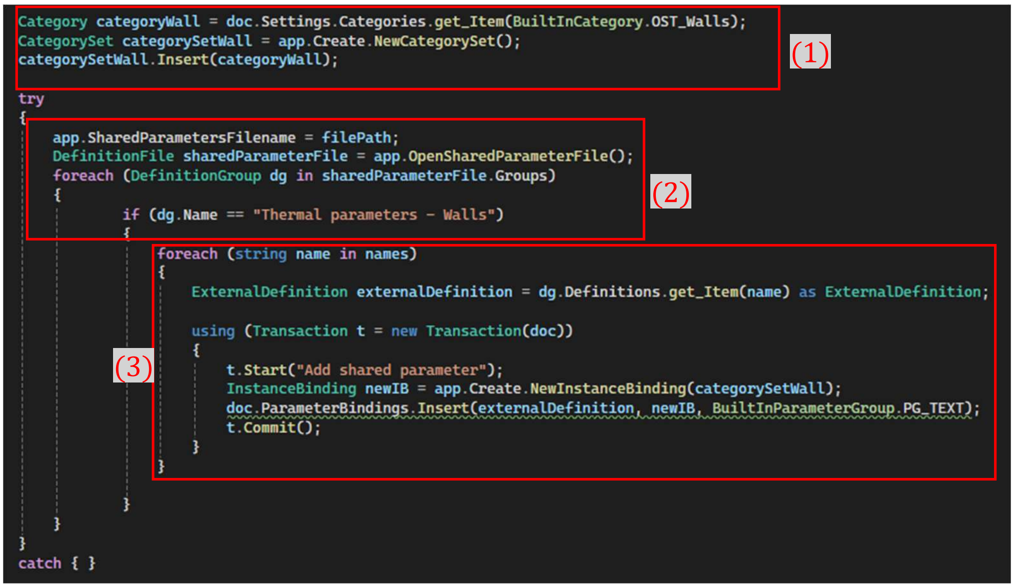
1) Walls, Roofs, Floors
2) Windows, Doors



```
List<string> names = new List<string>
{
    "RSI", "RSE", "COVER"
};
List<string> names2 = new List<string>
{
    "COVER", "U-VALUE"
};
```

Figure 79 - Assigned parameters.

1) I access the category of walls to be available to insert parameters in them.
2) I reach the category of parameters for walls in the sharedParameters.txt file.
3) I reach every parameter that wall needs to have and assign them.



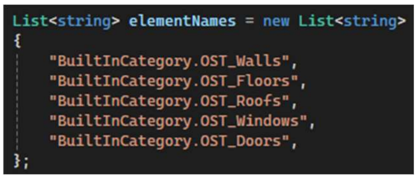
```
Category categoryWall = doc.Settings.Categories.get_Item(BuiltInCategory.OST_Walls);
CategorySet categorySetWall = app.Create.NewCategorySet();
categorySetWall.Insert(categoryWall);

try
{
    app.SharedParametersFilename = filePath;
    DefinitionFile sharedParameterFile = app.OpenSharedParameterFile();
    foreach (DefinitionGroup dg in sharedParameterFile.Groups)
    {
            if (dg.Name == "Thermal parameters - Walls")
            {
                foreach (string name in names)
                {
                    ExternalDefinition externalDefinition = dg.Definitions.get_Item(name) as ExternalDefinition;

                    using (Transaction t = new Transaction(doc))
                    {
                        t.Start("Add shared parameter");
                        InstanceBinding newIB = app.Create.NewInstanceBinding(categorySetWall);
                        doc.ParameterBindings.Insert(externalDefinition, newIB, BuiltInParameterGroup.PG_TEXT);
                        t.Commit();
                    }
                }

            }
    }
}
catch { }
```

Figure 80 - Assigning parameters to walls.

Now I repeat the same process with all remaining elements – Roofs, floors, windows, and doors.

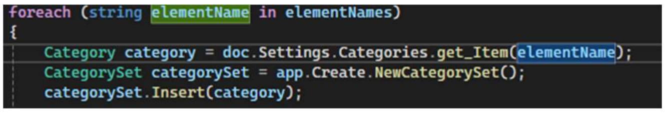
To save some code lines I have tried to loop it through elements, and categories, that were stored in list like the element parameters are, but it wasn´t working from two reasons, firstly when the code reached parameter like "RSE", that was not available for the element like window, it have just ended the work, so I had at least split it to two parts – walls + roofs + floors and windows + doors.

But then the unsolvable problem has appeared. If I write the element name manually in the right code part, it works properly. But when I just put in the reference on the list, even when the code for the program reading looks the same, it will just show an error.



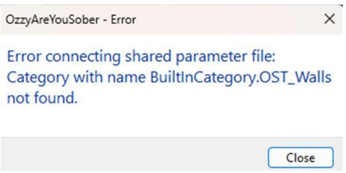
```
List<string> elementNames = new List<string>
{
    "BuiltInCategory.OST_Walls",
    "BuiltInCategory.OST_Floors",
    "BuiltInCategory.OST_Roofs",
    "BuiltInCategory.OST_Windows",
    "BuiltInCategory.OST_Doors",
};
```

Figure 81 - List of Element categories.



```
foreach (string elementName in elementNames)
{
    Category category = doc.Settings.Categories.get_Item(elementName);
    CategorySet categorySet = app.Create.NewCategorySet();
    categorySet.Insert(category);
```

Figure 82 - Loop to reach each element.



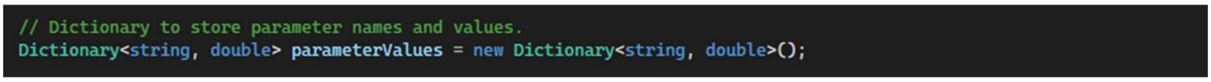
OzzyAreYouSober - Error

Error connecting shared parameter file:
Category with name BuiltInCategory.OST_Walls
not found.

Figure 83 - Error that Revit shows after running the plugin.

Getting parameters

Firstly, I create a dictionary – option to store parameters as – Name + value.

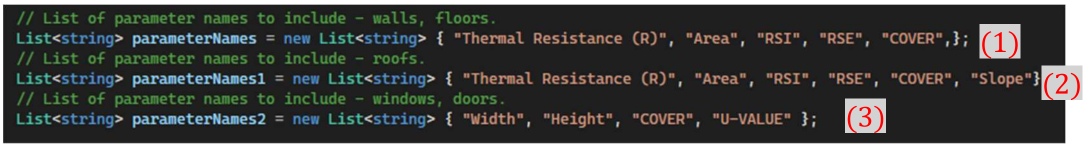


```
// Dictionary to store parameter names and values.
Dictionary<string, double> parameterValues = new Dictionary<string, double>();
```

Figure 84 - Dictionary to store parameters.

lists of parameters I want to get.

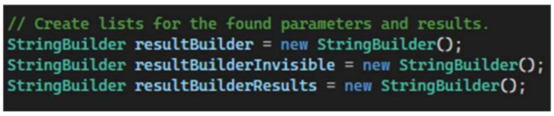
1) Walls, floors
2) Roofs
3) Windows, Doors



```
// List of parameter names to include - walls, floors.
List<string> parameterNames = new List<string> { "Thermal Resistance (R)", "Area", "RSI", "RSE", "COVER",};
// List of parameter names to include - roofs.
List<string> parameterNames1 = new List<string> { "Thermal Resistance (R)", "Area", "RSI", "RSE", "COVER", "Slope"}
// List of parameter names to include - windows, doors.
List<string> parameterNames2 = new List<string> { "Width", "Height", "COVER", "U-VALUE" };
```

Figure 85 - Parameters to get.

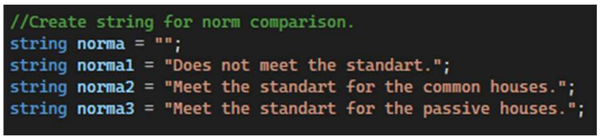
Then I set string builders to store the parameters.



```
// Create lists for the found parameters and results.
StringBuilder resultBuilder = new StringBuilder();
StringBuilder resultBuilderInvisible = new StringBuilder();
StringBuilder resultBuilderResults = new StringBuilder();
```

Figure 86 - String builders as storages for values

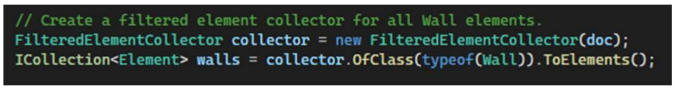
I create comparison with ČSN, value of string "norma" will always change to value of the norma1 – norma3, depending on the results.



```
//Create string for norm comparison.
string norma = "";
string norma1 = "Does not meet the standart.";
string norma2 = "Meet the standart for the common houses.";
string norma3 = "Meet the standart for the passive houses.";
```

Figure 87 - ČSN comparison

I get all the wall elements.

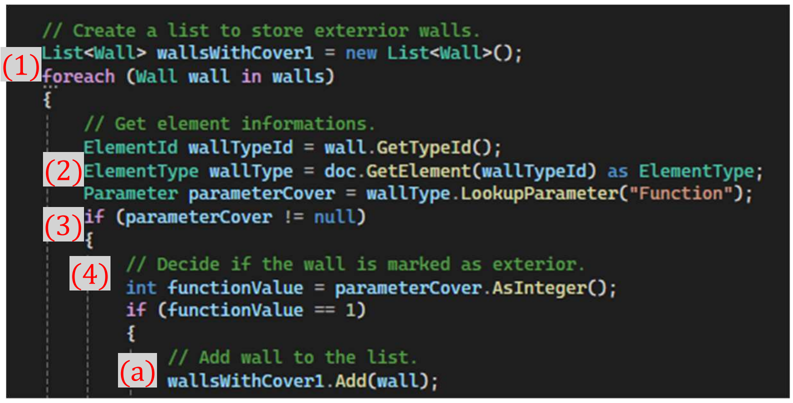


```
// Create a filtered element collector for all Wall elements.
FilteredElementCollector collector = new FilteredElementCollector(doc);
ICollection<Element> walls = collector.OfClass(typeof(Wall)).ToElements();
```

Figure 88 - Wall element collector

In the walls there is a function exterior/ interior, so I have decided to use this parameter instead of cover for the walls, but for the rest of elements, the filter remains depending on the parameter cover.

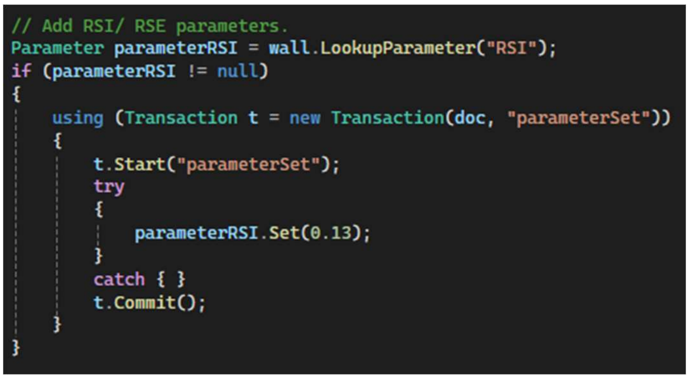
1) I create list for exterior walls.
2) Get to the instance and type of wall parameters.
3) If the wall is interior, loop goes to start for another wall.
4) If it is exterior
    a. Add it to the list of exterior walls.



```
// Create a list to store exterior walls.
List<Wall> wallsWithCover1 = new List<Wall>();
foreach (Wall wall in walls)
{
    // Get element informations.
    ElementId wallTypeId = wall.GetTypeId();
    ElementType wallType = doc.GetElement(wallTypeId) as ElementType;
    Parameter parameterCover = wallType.LookupParameter("Function");
    if (parameterCover != null)
    {
        // Decide if the wall is marked as exterior.
        int functionValue = parameterCover.AsInteger();
        if (functionValue == 1)
        {
            // Add wall to the list.
            wallsWithCover1.Add(wall);
```

Figure 89 - Getting info about exterior walls.

b. Set R_{si}

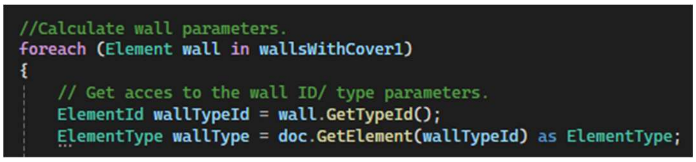


```
// Add RSI/ RSE parameters.
Parameter parameterRSI = wall.LookupParameter("RSI");
if (parameterRSI != null)
{
    using (Transaction t = new Transaction(doc, "parameterSet"))
    {
        t.Start("parameterSet");
        try
        {
            parameterRSI.Set(0.13);
        }
        catch { }
        t.Commit();
    }
}
```

Figure 90 - Setting R_{si}.

c. Set R_{se} (with the same principle)

Next part is loop, that takes every wall, get parameters, and do the calculations.

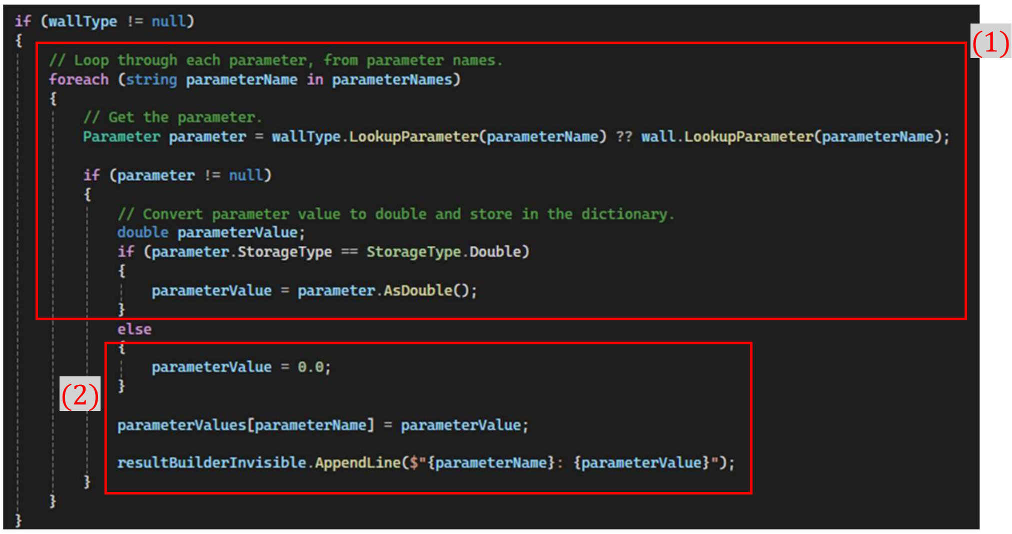


```
//Calculate wall parameters.
foreach (Element wall in wallsWithCover1)
{
    // Get acces to the wall ID/ type parameters.
    ElementId wallTypeId = wall.GetTypeId();
    ElementType wallType = doc.GetElement(wallTypeId) as ElementType;
```

Figure 91 - Getting every wall.

This part reaches every parameter in my list of desired parameters.

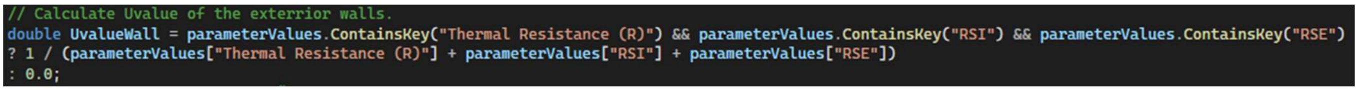
1) Get every parameter value.
2) Store the parameter name and value in my String Builder.



```
if (wallType != null)
{
    // Loop through each parameter, from parameter names.
    foreach (string parameterName in parameterNames)
    {
        // Get the parameter.
        Parameter parameter = wallType.LookupParameter(parameterName) ?? wall.LookupParameter(parameterName);

        if (parameter != null)
        {
            // Convert parameter value to double and store in the dictionary.
            double parameterValue;
            if (parameter.StorageType == StorageType.Double)
            {
                parameterValue = parameter.AsDouble();
            }
            else
            {
                parameterValue = 0.0;
            }

            parameterValues[parameterName] = parameterValue;

            resultBuilderInvisible.AppendLine($"{parameterName}: {parameterValue}");
        }
    }
}
```

Figure 92 - Getting wall parameters.

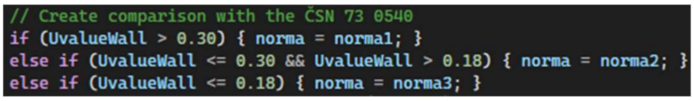
Then it gets to value of parameter Thermal Resistance (R), R_{si}, R_{se}, and calculates U-Value.



```
// Calculate Uvalue of the exterrior walls.
double UvalueWall = parameterValues.ContainsKey("Thermal Resistance (R)") && parameterValues.ContainsKey("RSI") && parameterValues.ContainsKey("RSE")
? 1 / (parameterValues["Thermal Resistance (R)"] + parameterValues["RSI"] + parameterValues["RSE"])
: 0.0;
```

Figure 93 - U-Value calculation.

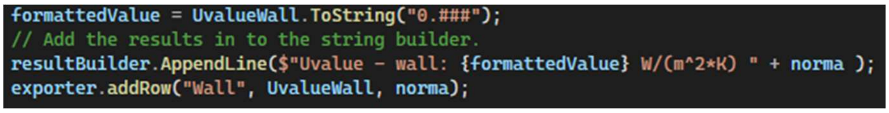
Compare the U-Value to ČSN.



```
// Create comparison with the ČSN 73 0540
if (UvalueWall > 0.30) { norma = norma1; }
else if (UvalueWall <= 0.30 && UvalueWall > 0.18) { norma = norma2; }
else if (UvalueWall <= 0.18) { norma = norma3; }
```

Figure 94 - Comparing to ČSN.

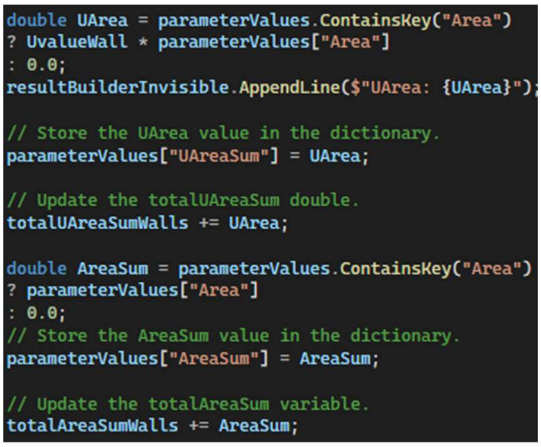
Format the U-Value to be just in 3 decimal places in the dialog window and add it to the string builder, that will be shown as Dialog window, with U-Values for all elements.



```
formattedValue = UvalueWall.ToString("0.###");
// Add the results in to the string builder.
resultBuilder.AppendLine($"Uvalue - wall: {formattedValue} W/(m^2*K) " + norma );
exporter.addRow("Wall", UvalueWall, norma);
```

Figure 95 - Storing the values.

Then do the calculations for the weighted average of all walls.



```
double UArea = parameterValues.ContainsKey("Area")
? UvalueWall * parameterValues["Area"]
: 0.0;
resultBuilderInvisible.AppendLine($"UArea: {UArea}");

// Store the UArea value in the dictionary.
parameterValues["UAreaSum"] = UArea;

// Update the totalUAreaSum double.
totalUAreaSumWalls += UArea;

double AreaSum = parameterValues.ContainsKey("Area")
? parameterValues["Area"]
: 0.0;
// Store the AreaSum value in the dictionary.
parameterValues["AreaSum"] = AreaSum;

// Update the totalAreaSum variable.
totalAreaSumWalls += AreaSum;
```

Figure 96 - Parameters for the weighted average.

The same process is repeated for the rest of elements, there are some differences while calculating other elements.

Floors

I needed to separate the floors on the ground and floors in the bays, parameter cover needs to be:

- 0 – For interior floors (or empty, it doesn´t matter).
- 1 – For floors on the ground.
- 2 – For the bay floors

For floors on the ground there is se R_{si} to 0.17 and R_{se} stays empty. For the bay floors I add value 0.04 to R_{se}.

Roofs

Roofs need to be separated by the type, so the parameter cover need to be:

- 0 (or empty) – For roof overhang, or other roof that are not part of cover.
- 1 – Single shelled roofs.
- 2 – Double shelled roofs.

R_{si} is set for both counted roofs to 0.1. R_{se} is being set to 0.1 for double shelled roofs and 0.04 for single shelled roofs.

Second part, where are the roofs need to be separated is comparison to ČSN, because when the slope is $\leq 45°$, there is a different comparison, than when the slope is $> 45°$.

Windows

As the roof are, windows are being separated by comparison to ČSN by the same slope rule. Windows in the walls and windows in the roofs with slope $> 45°$ are being compared to the same values.

Getting into the slope of window was hard because windows doesn't have parameter slope, so I had to reach the hosting element, which have taken me another whole day of work, but I have made it work. So, it works well.

Problem with wall Area

Revit calculates area of the walls from height ant length, problem is, that the length of wall is in the middle of the wall (orange lines), and we need to calculate it form the exterior surface (green lines). Solution is to set all wall joins from butt to miter. But Revit API does not allow to change this option by the plugin, so user have to do it manually for all of the walls.

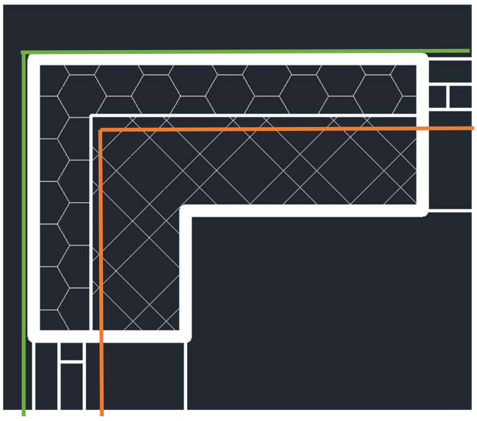


Figure 97 - Wall length calculation

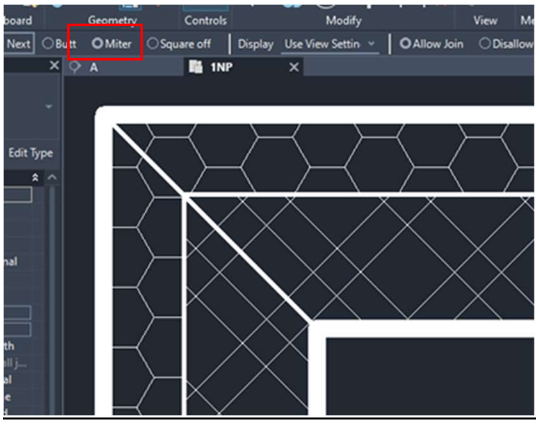
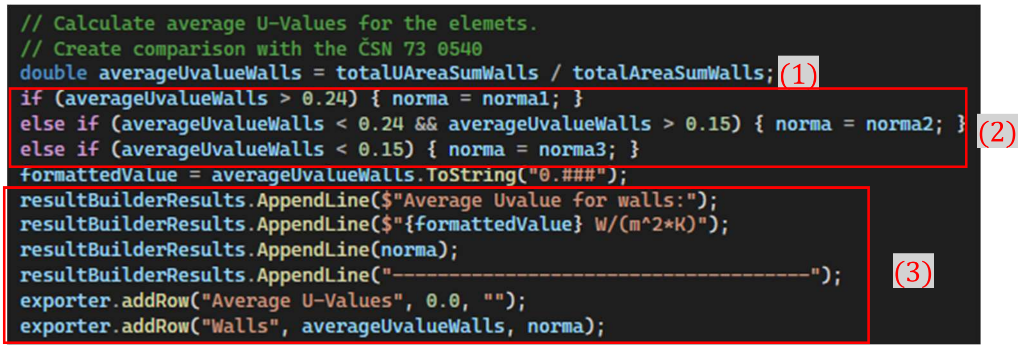


Figure 98 - Option to fix wall area.

Average values

In the last part, there is:

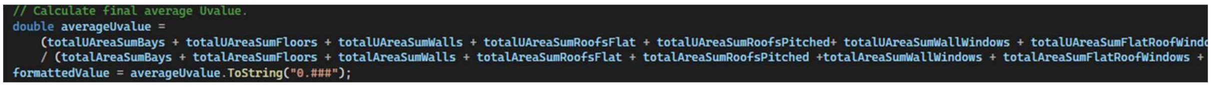
1) Calculation of the weighted average for elements.
2) Comparison to ČSN.
3) And adding it to Task Dialog windows and excel export.



```
// Calculate average U-Values for the elemets.
// Create comparison with the ČSN 73 0540
double averageUvalueWalls = totalUAreaSumWalls / totalAreaSumWalls;
if (averageUvalueWalls > 0.24) { norma = norma1; }
else if (averageUvalueWalls < 0.24 && averageUvalueWalls > 0.15) { norma = norma2; }
else if (averageUvalueWalls < 0.15) { norma = norma3; }
formattedValue = averageUvalueWalls.ToString("0.###");
resultBuilderResults.AppendLine($"Average Uvalue for walls:");
resultBuilderResults.AppendLine($"{formattedValue} W/(m^2*K)");
resultBuilderResults.AppendLine(norma);
resultBuilderResults.AppendLine("------------------------------------");
exporter.addRow("Average U-Values", 0.0, "");
exporter.addRow("Walls", averageUvalueWalls, norma);
```

Figure 99 - Average U-Value calculation for walls.

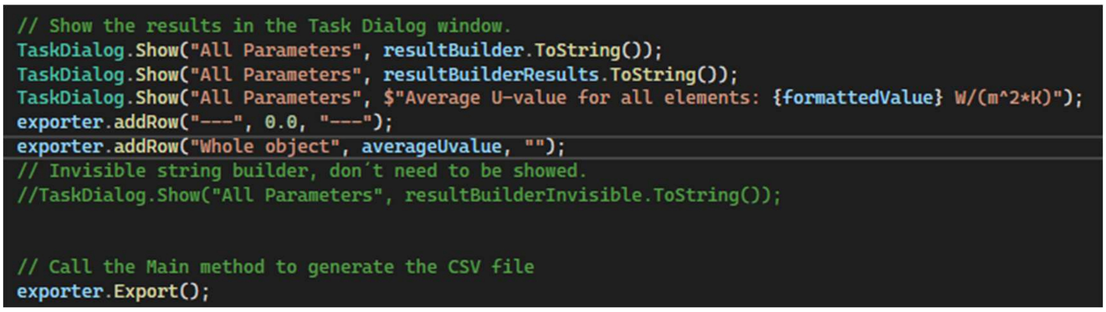
Weighted average for all elements, to create final U-Value.



```
// Calculate final average Uvalue.
double averageUvalue =
    (totalUAreaSumBays + totalUAreaSumFloors + totalUAreaSumWalls + totalUAreaSumRoofsFlat + totalUAreaSumRoofsPitched+ totalUAreaSumWallWindows + totalUAreaSumFlatRoofWindo
    / (totalAreaSumBays + totalAreaSumFloors + totalAreaSumWalls + totalAreaSumRoofsFlat + totalAreaSumRoofsPitched +totalAreaSumWallWindows + totalAreaSumFlatRoofWindows +
formattedValue = averageUvalue.ToString("0.###");
```

Figure 100 - Calculating average U-Value.

Show task dialog windows with results and generate the .CSV file.



```
// Show the results in the Task Dialog window.
TaskDialog.Show("All Parameters", resultBuilder.ToString());
TaskDialog.Show("All Parameters", resultBuilderResults.ToString());
TaskDialog.Show("All Parameters", $"Average U-value for all elements: {formattedValue} W/(m^2*K)");
exporter.addRow("---", 0.0, "---");
exporter.addRow("Whole object", averageUvalue, "");
// Invisible string builder, don´t need to be showed.
//TaskDialog.Show("All Parameters", resultBuilderInvisible.ToString());


// Call the Main method to generate the CSV file
exporter.Export();
```

Figure 101 - Showing and exporting results.

4.4) Summary of the C# Results

In this comparison, for better orientation there are results from task dialog window, in the exported excel, the results are on all calculated decimal places.

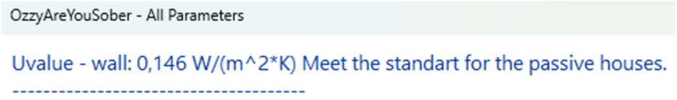
I have made working script, that calculates U-Values in whole project.

W1 – Wall

$$U_{\check{C}SN} = \mathbf{0.1461}\,\boldsymbol{W}/(\boldsymbol{m^2} * \boldsymbol{K})$$

In this dialog task window, there are results of all calculated elements.

Calculation of the wall meets the manual calculation according to ČSN.



OzzyAreYouSober - All Parameters

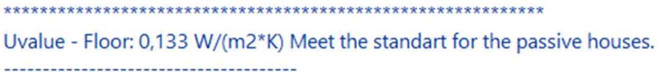
```
Uvalue - wall: 0,146 W/(m^2*K) Meet the standart for the passive houses.
------------------------------------
```

Figure 102 - Results of the calculations - wall.

F1 – Floor on the ground

$$U_{\check{C}SN} = \mathbf{0.1334}\,\boldsymbol{W}/(\boldsymbol{m^2} * \boldsymbol{K})$$

Calculation of the floor on the ground meets the manual calculation according to ČSN.



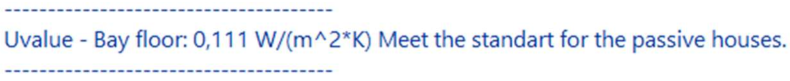
```
*************************************************************
Uvalue - Floor: 0,133 W/(m2*K) Meet the standart for the passive houses.
------------------------------------
```

Figure 103 - Results of the calculations - Floor on the ground.

F2 – Bay floor

$$U_{\check{C}SN} = \mathbf{0.1114}\,\boldsymbol{W}/(\boldsymbol{m^2} * \boldsymbol{K})$$

Calculation of the bay floor meets the manual calculation according to ČSN.



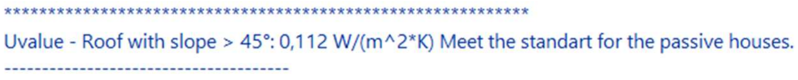
```
------------------------------------
Uvalue - Bay floor: 0,111 W/(m^2*K) Meet the standart for the passive houses.
------------------------------------
```

Figure 104 - Results of the calculations - Bay floor.

R1 – Sloped roof

$$U_{\check{C}SN} = \mathbf{0.1116}\,\boldsymbol{W}/(\boldsymbol{m^2} * \boldsymbol{K})$$

Calculation of the sloped roof meets the manual calculation according to ČSN.



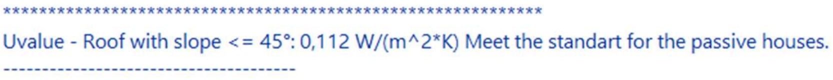
```
*************************************************************
Uvalue - Roof with slope > 45°: 0,112 W/(m^2*K) Meet the standart for the passive houses.
------------------------------------
```

Figure 105 - Results of the calculations - Sloped roof.

R2 – Flat roof

$$U_{\check{C}SN} = \mathbf{0.1124}\,\boldsymbol{W}/(\boldsymbol{m^2} * \boldsymbol{K})$$

Calculation of the e ground meets the manual calculation according to ČSN.



```
*************************************************************
Uvalue - Roof with slope <= 45°: 0,112 W/(m^2*K) Meet the standart for the passive houses.
------------------------------------
```

Figure 106 - Results of the calculations - Flat roof.

Evaluation of the results

Tab. 1 – Summary of the plugin results

Element	ČSN U-Value [W/(m²*K)]	Dynamo U-Value [W/(m²*K)]	Value fits
Wall	0.1461	0.146	OK
G. Floor	0.1334	0.133	OK
B. Floor	0.1114	0.111	OK
Sl. Roof	0.1116	0.112	OK
Flat Roof	0.1124	0.112	OK

I have made plugin, that can calculate the U-Values according to ČSN. It has all desired functions.

Plugin is really easy to use, every result and comparison to ČSN is available in just few clicks from Revit.

All of the results meet the manual calculation, it means that plugin works successfully.

Presentation of the final plugin step by step is included in the chapter 5.2) Summary of the results.

5) Summary

In this chapter I will tell you about the terms of use the plugin, evaluate whole work, summarize the benefits, and give you some future outlook.

5.1) Terms of use

Version of Revit should be 2024 on which it was tested, probably it will work on the other versions too, but it is not guaranteed.

The building needs to have just one thermal zone.

Every wall should me marked as exterior/ interior, if the foundation strips are modelled as walls, should be marked as interior, to not include them in the calculations.

All exterior walls should have the function “wall joins” set on “miter” to have accurate wall area.

Double shelled roofs should have the shells modelled separately.

Roof overhangs should be modelled separately to get the right roof area.

To get the results, you need to run the plugin twice, after first run you must set parameter values:

- Windows
  - Parameter COVER: “1” to all windows included in the cover of the building.
  - Parameter U-VALUE – insert U-Value of your windows.
- Doors
  - Parameter COVER: “1” to all doors included in the cover of the building.
  - Parameter U-VALUE – insert U-Value of your doors.
- Roofs
  - Parameter COVER: “1” to all single shelled roofs included in the cover of the building.
  - Parameter COVER: “2” to all double shelled roofs included in the cover of the building.
- Floors
  - Parameter COVER: “1” to all floors on the ground included in the cover of the building.
  - Parameter COVER: "2” to all floors in the bays included in the cover of the building.

Then run the plugin again to get you results and export them into excel.

5.2) Summary of the results

Goal of this diploma thesis was to create a Revit add-in, able to calculate thermal resistance of constructions while counting with R_{si} and R_{se}.

Goal was fulfilled in the expected scope.

Presentation of the final add-in.

All that is needed to get the results from model fulfilling the terms of use is:

Click on Add-ins (1), External Tools (2) and chose the plugin.

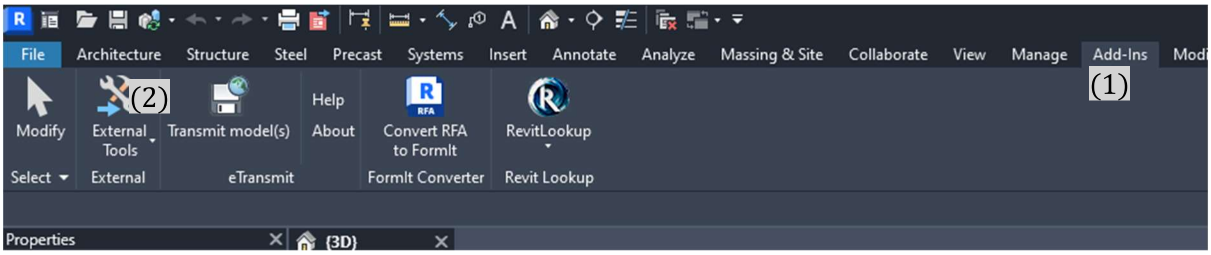


Figure 107 - Where to find add-in in the Revit.

On the first run in the project, just press key esc few times, plugin has added parameters into the model.

Insert values to your object according to terms of use. Cover (1) and U-Values (2).

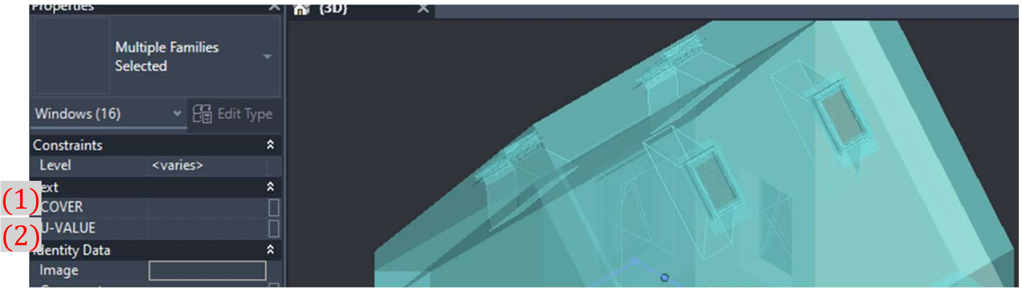
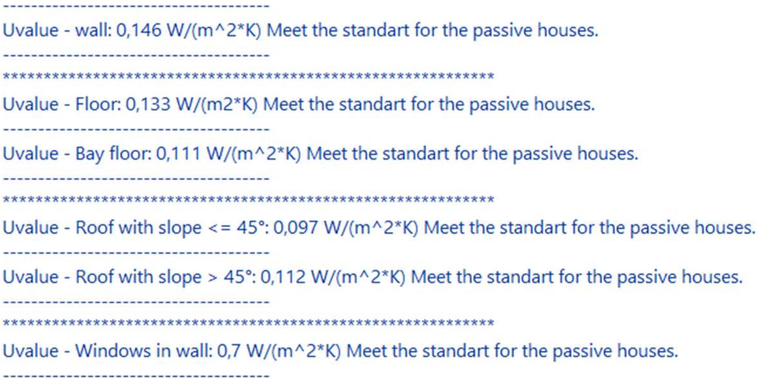


Figure 108 - Parameters to insert.

Then run the plugin again.

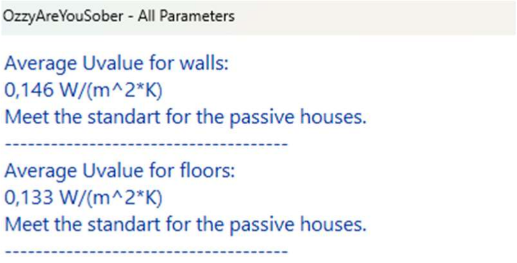
First thing that will show up is task dialog window with U-Values and comparison to the ČSN of every element marked as part of the cover. The elements are separated to categories: walls, floors on the ground, floors in the bays, roofs with slope > 45°, roofs with slope ≤ 45°, windows in the walls, windows in the roofs with slope > 45°, windows in the roofs with slope ≤ 45°, and doors.



```
----------------------------------------
Uvalue - wall: 0,146 W/(m^2*K) Meet the standart for the passive houses.
----------------------------------------
************************************************************
Uvalue - Floor: 0,133 W/(m2*K) Meet the standart for the passive houses.
----------------------------------------
Uvalue - Bay floor: 0,111 W/(m^2*K) Meet the standart for the passive houses.
----------------------------------------
************************************************************
Uvalue - Roof with slope <= 45°: 0,097 W/(m^2*K) Meet the standart for the passive houses.
----------------------------------------
Uvalue - Roof with slope > 45°: 0,112 W/(m^2*K) Meet the standart for the passive houses.
----------------------------------------
************************************************************
Uvalue - Windows in wall: 0,7 W/(m^2*K) Meet the standart for the passive houses.
----------------------------------------
```

Figure 109 – U-Values for the elements.

The next dialog window shows weighted average U-Value and ČSN comparison for each category from the first dialog window.



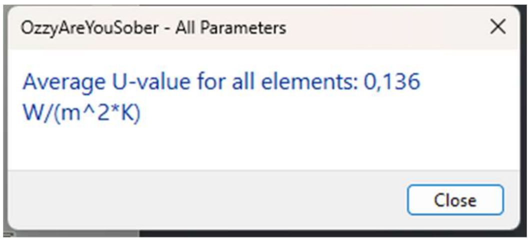
OzzyAreYouSober - All Parameters

Average Uvalue for walls:
0,146 W/(m^2*K)
Meet the standart for the passive houses.

Average Uvalue for floors:
0,133 W/(m^2*K)
Meet the standart for the passive houses.

Figure 110 - Average U-Values for elements.

The last dialog window shows weighted average U-Value for the whole house cover.

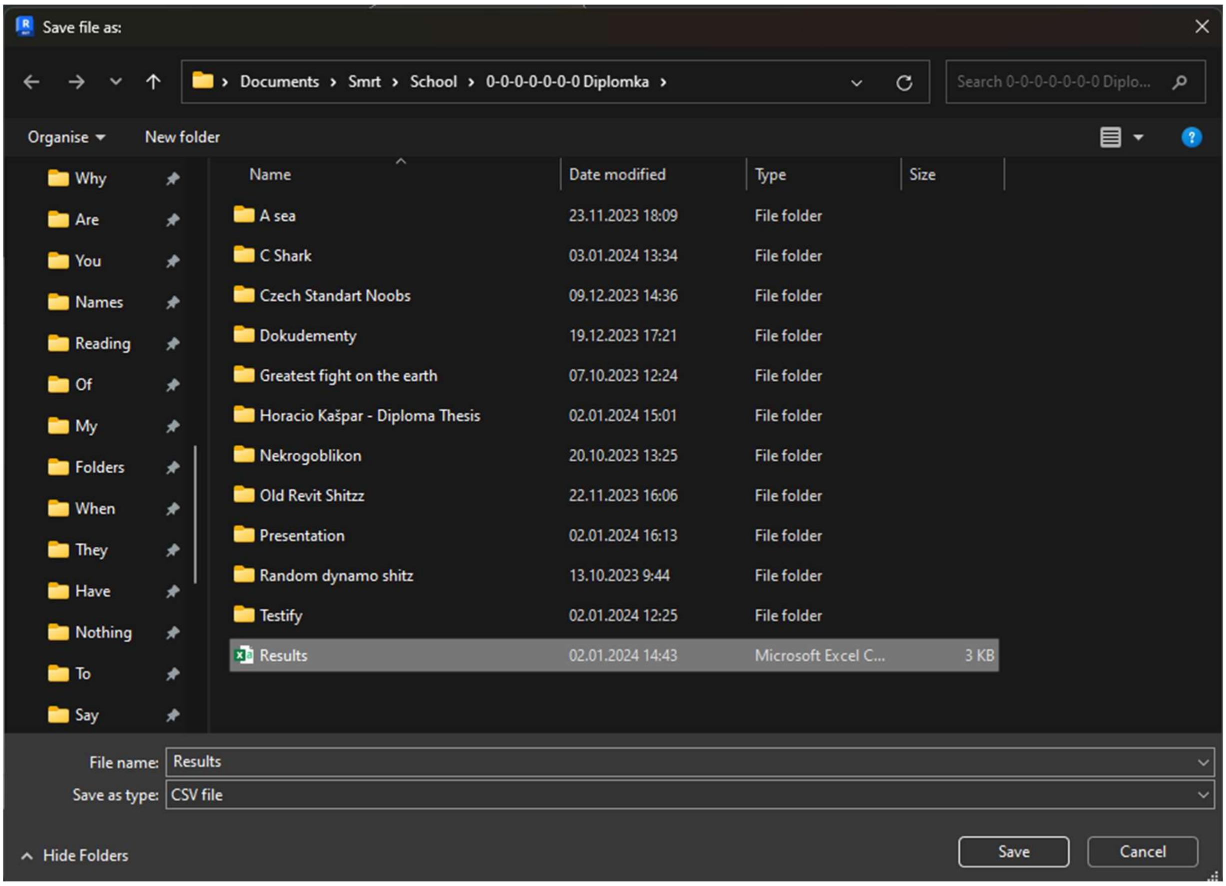


OzzyAreYouSober - All Parameters

Average U-value for all elements: 0,136 W/(m^2*K)

Figure 111 - Average U-Value for cover.

Then window “save file as” appears

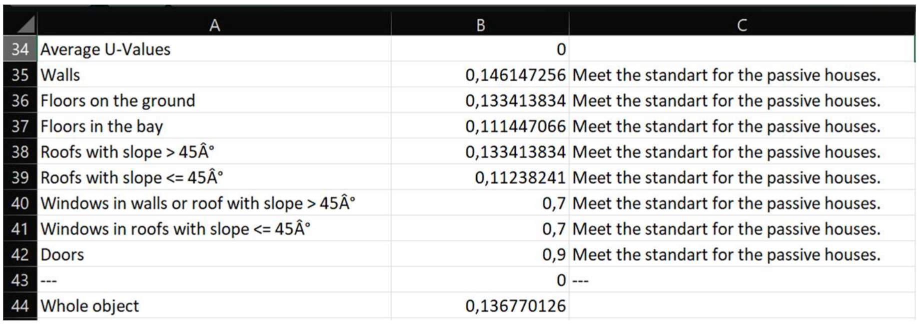


Name	Date modified	Type	Size
A sea	23.11.2023 18:09	File folder	
C Shark	03.01.2024 13:34	File folder	
Czech Standart Noobs	09.12.2023 14:36	File folder	
Dokudementy	19.12.2023 17:21	File folder	
Greatest fight on the earth	07.10.2023 12:24	File folder	
Horacio Kašpar - Diploma Thesis	02.01.2024 15:01	File folder	
Nekrogoblikon	20.10.2023 13:25	File folder	
Old Revit Shitzz	22.11.2023 16:06	File folder	
Presentation	02.01.2024 16:13	File folder	
Random dynamo shitz	13.10.2023 9:44	File folder	
Testify	02.01.2024 12:25	File folder	
Results	02.01.2024 14:43	Microsoft Excel C...	3 KB

File name: Results
Save as type: CSV file

Figure 112 - "Save as" window.

Congratulations all average U-Values were saved in the excel table.



	A	B	C
34	Average U-Values	0	
35	Walls	0,146147256	Meet the standart for the passive houses.
36	Floors on the ground	0,133413834	Meet the standart for the passive houses.
37	Floors in the bay	0,111447066	Meet the standart for the passive houses.
38	Roofs with slope > 45Â°	0,133413834	Meet the standart for the passive houses.
39	Roofs with slope <= 45Â°	0,11238241	Meet the standart for the passive houses.
40	Windows in walls or roof with slope > 45Â°	0,7	Meet the standart for the passive houses.
41	Windows in roofs with slope <= 45Â°	0,7	Meet the standart for the passive houses.
42	Doors	0,9	Meet the standart for the passive houses.
43	---	0	---
44	Whole object	0,136770126	

Figure 113 - Results in excel table.

5.3) Benefit summary

I have learned a lot about coding and learning is the main goal of doing any work to school, so in this point it was very successful.

Also, the final add-in can be very helpful tool for engineers while designing buildings and calculating the certificate. Seeing the U-Value and comparison to ČSN just after few clicks and having the numbers exported into excel is simply good help.

Add-in can calculate average U-Values for all types of elements, so engineer will not see just the elements itself, also the weighted averages, so he or she will instantly know, if there is or isn´t element, that will push object into the lower category.

5.4) Future outlook

The add-in can be developed to include multiple thermal zones. It wasn´t possible to do it in a such short time, because the filter of calculated elements will be a lot more complicated, it isn´t possible to use current system, because the user would be forced to divide elements into more pieces.

The filter of calculated elements can be upgraded to be more automatic while use the rooms and create intersections with elements.

Another useful update can be adding live connection to Revit, that should allow the ČSN comparison to be seen immediately while editing the compositions.

In the end, this add-in can be upgraded into complete energy label of the building envelope.

6) References

[1] m. a. s. z. Úřad pro technickou normalizaci, ČSN 73 0540, Úřad pro technickou normalizaci, metrologii a státní zkušebnictví., 2011.

[2] ©. 2. A. I. A. r. reserved, “Autodesk Support,” [Online]. Available: https://www.autodesk.com/support/technical/article/caas/tsarticles/ts/7I2bC1zUr4VjJ3U31uM66K.html.

7) Attachment

In this part I will tell you how I have personally got into BIM and how I have used AI while I was learning to code.

7.1) My road to BIM

Firstly, I like to do civil Engineering in the BIM softwares, because I am person who likes to do everything in the most possible effective way.

Also, I think that BIM is whole future of the civil engineering and I want to be part of it.

At the beginning of college, when I was asked if I want my current job, I have nearly refused, because on my high school we had terrible teacher on Archicad and I have kind of hated it because thanks to her I have not seen any advantage in it, just tons of useless functions. But I have took it as challenge and I have accepted the work. After some learning I have started to see how effective the 3D modelling is and now I know that it is really useful.

Most of my current job in NB Projekce is civil engineering. We work in Archicad and as I said earlier, I like to do things as effectively as it is possible. Thanks to this part of my personality, I have learned how to use this program really well. Also, I am in this company since it has started, and this part have happened in the beginning. Once when I was completing project after another colleague, the model was in horrible state. I have written about 10 pages of text and screenshots to my boss, and I have told him: "please give it to all colleagues, these are mistakes, that slows down work in the project."

He has read it, and he was like: "I like it, write a textbook about methods and how we draw everything."

So, I have done it, first complete version of the textbook had 128 pages, we still use it for teaching new or even current colleagues. I am still upgrading it when I find new more effective methods or that something is missing.

I am also taking care of our project template and libraries, giving learning tests to new colleagues, and checking if they have made it the right way. Also, I am helping everyone trough the messages or directly in the office.

I really like this part of my job so after graduating I want to find some job to learn properly with Revit and be BIM technician as most of my job.

7.2) Coding and thermal solving

Considering all the facts, the first day of my master studies I have reached Ing. Jakub Veselka, Ph.D. because it was the best BIM expert, I have met through my studies. We have talked about what I want to do in my diploma thesis.

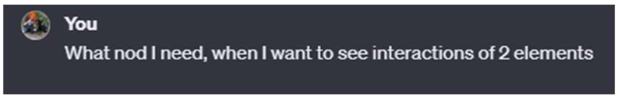
One of possible themes that he came up with, was coding a Revit plugin. I was immediately hooked because it has seemed like a great challenge. Mainly because I have never even tried to learn how to code. Also, it was going to be very useful, not just the plugin itself, but also in the way that when I want to become BIM expert, some coding skills may come in handy and there is not a better time to learn it than in diploma thesis.

7.3) Dynamo and AI

Artificial intelligence is a tool, that can be useful in many ways. So, I have tested how useful the AI can be when it comes to coding in C# and scripting in Dynamo.

I have used AI chatbot called ChatGPT made by firm OpenAI. I have used free version ChatGPT 3.5

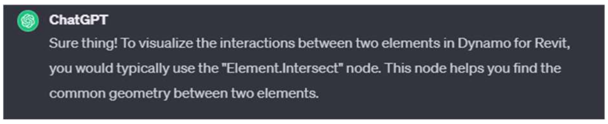
All you need to do is ask the AI question as if you were communicating with a college.



What nod I need, when I want to see interactions of 2 elements

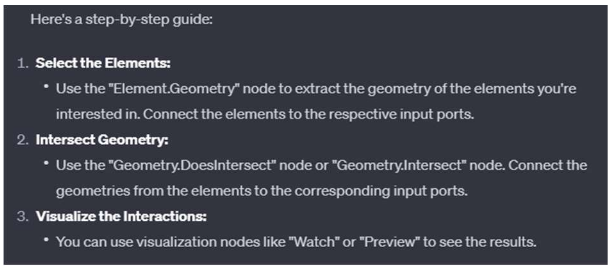
Figure 114 - Example for question given to AI.

Then you wait until your answer is generated, most of times it takes just few seconds until your answer is generated. Answer for dynamo questions is made from four parts.



Sure thing! To visualize the interactions between two elements in Dynamo for Revit, you would typically use the "Element.Intersect" node. This node helps you find the common geometry between two elements.

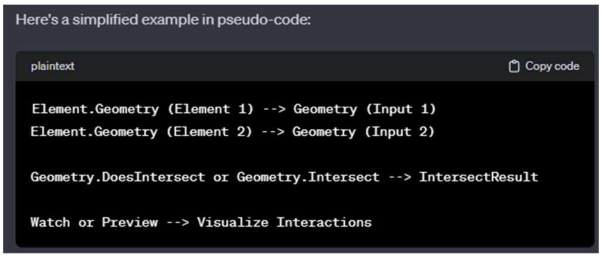
Figure 115 - Short answer for given question.



Here's a step-by-step guide:

1. **Select the Elements:**
   - Use the "Element.Geometry" node to extract the geometry of the elements you're interested in. Connect the elements to the respective input ports.
2. **Intersect Geometry:**
   - Use the "Geometry.DoesIntersect" node or "Geometry.Intersect" node. Connect the geometries from the elements to the corresponding input ports.
3. **Visualize the Interactions:**
   - You can use visualization nodes like "Watch" or "Preview" to see the results.

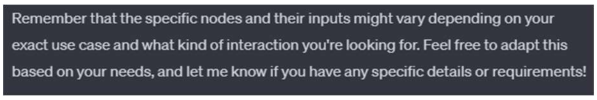
Figure 116 - Step by step gude.



Here's a simplified example in pseudo-code:

```
Element.Geometry (Element 1) --> Geometry (Input 1)
Element.Geometry (Element 2) --> Geometry (Input 2)

Geometry.DoesIntersect or Geometry.Intersect --> IntersectResult

Watch or Preview --> Visualize Interactions
```

Figure 117 - Pseudo - code.



Remember that the specific nodes and their inputs might vary depending on your exact use case and what kind of interaction you're looking for. Feel free to adapt this based on your needs, and let me know if you have any specific details or requirements!

Figure 118 - Warning that it might not work.

Sometimes it comes up with a good solution, but if it doesn´t know, it will come with made up answer, that doesn´t make sense and can advise you to search for nods that are don´t exist etc. However, these solutions are not useless, they can offer you a different point of view, that can help you to come up with the right idea.

7.4) AI – future of coding?

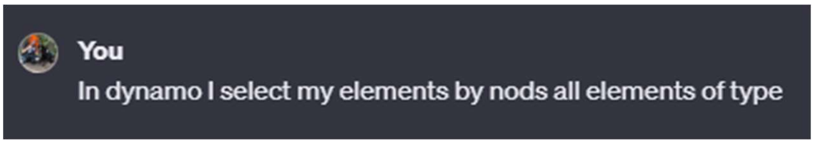
As I have advertised before, AI is inseparable part of our future. When I was learning with Dynamo, ChatGPT was useful just a bit, but while I was coding in C#, I had a better experience.

AI learns from every question you give to it, so when you ask it about more common topic, it is more likely to give you the right answer. Since C# is more common to use than Dynamo, it is logical that while I was asking AI the C# questions it was working better than with Dynamo.

AI is great tool, but you need to learn how to ask questions and while asking C# questions it is crucial, because it gives you answer with a written code, that you can just copy, paste, and use.

Before shown question I have told it that I have Dynamo script that I want to replicate and upgrade in C#, it has sent me some lines of code to include to make my plugin work with the Revit.

Then I have asked him how to select my elements.

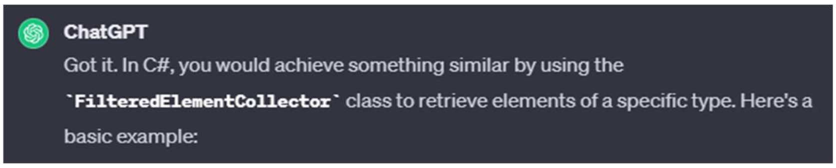


You

In dynamo I select my elements by nods all elements of type

Figure 119 - Question for the AI.

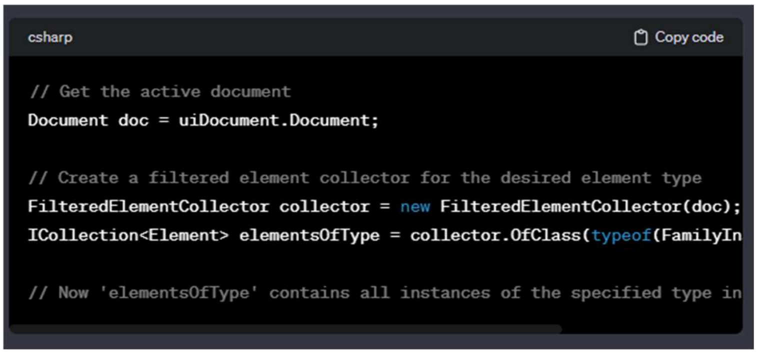
ChatGPT´s answers were always from three parts:



ChatGPT

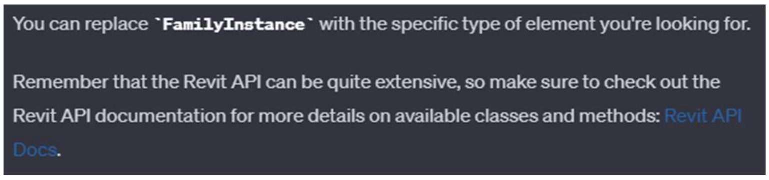
Got it. In C#, you would achieve something similar by using the `FilteredElementCollector` class to retrieve elements of a specific type. Here's a basic example:

Figure 120 - First part of the answer - solution description



```csharp
// Get the active document
Document doc = uiDocument.Document;

// Create a filtered element collector for the desired element type
FilteredElementCollector collector = new FilteredElementCollector(doc);
ICollection<Element> elementsOfType = collector.OfClass(typeof(FamilyIn

// Now 'elementsOfType' contains all instances of the specified type in
```

Figure 121 - Second part of the answer - code.



You can replace `FamilyInstance` with the specific type of element you're looking for.

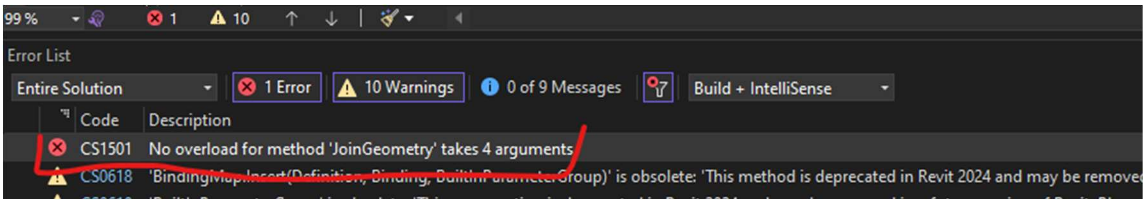
Remember that the Revit API can be quite extensive, so make sure to check out the Revit API documentation for more details on available classes and methods: Revit API Docs.

Figure 122 - Third part of the answer - cautions and the words replacement advice.

For the first example I have chosen the part, that was working right away. It has successfully made a "storage" in the code, where were stored wall elements, but it is not as easy as it looks, when you want to access to them and show for example their list, you need much more code lines that will access each wall and find for example parameter of their ID, but I will tell you about it later when I will be talking about the code itself.

Most of times copied will not work right away, but in Visual Studio Community (Coding program I have used), is another AI that will tell you what is wrong with your code.

For example, this time it has advised me something that was not possible.



Code	Description
CS1501	No overload for method 'JoinGeometry' takes 4 arguments
CS0618	'BindingMap.Insert(Definition, Binding, BuiltInParameterGroup)' is obsolete: 'This method is deprecated in Revit 2024 and may be remove

Figure 123 - Wrong advice

Lot of times this error window told me that I am trying to reach some category of parameters, but it is stored in different type of "element parameter storage".

Summary of AI coding in C#

As I have advised a lot of times yet, it was useful tool. I am sure that if I wasn´t able to work with ChatGPT, I would have no chance to learn and make the code working in that short time.

But it is just tool, it will not make the work for you, it is more like a kind colleague who will answer all of your "Hey bro, how do I access this" questions immediately, but he is not always right and if your program logic is wrong, he will probably have no idea how to help you.

But it will always provide some opinion, which can be wrong, but it can still help you, because you will think about your problem in a different way.