

# Automatiziranje razvojnih procesa unutar BIM programske podrške

---

Milić, Ana

Master's thesis / Diplomski rad

2021

*Degree Grantor / Ustanova koja je dodijelila akademski / stručni stupanj:* **University of Zagreb, Faculty of Mechanical Engineering and Naval Architecture / Sveučilište u Zagrebu, Fakultet strojarstva i brodogradnje**

*Permanent link / Trajna poveznica:* <https://urn.nsk.hr/urn:nbn:hr:235:134699>

*Rights / Prava:* [In copyright](#)/[Zaštićeno autorskim pravom.](#)

*Download date / Datum preuzimanja:* **2025-01-13**

*Repository / Repozitorij:*

[Repository of Faculty of Mechanical Engineering and Naval Architecture University of Zagreb](#)



SVEUČILIŠTE U ZAGREBU  
FAKULTET STROJARSTVA I BRODOGRADNJE

# **DIPLOMSKI RAD**

**Ana Milić**

Zagreb, 2021.

SVEUČILIŠTE U ZAGREBU  
FAKULTET STROJARSTVA I BRODOGRADNJE

# **DIPLOMSKI RAD**

Mentori:

Doc. dr. sc. Tomislav Stipančić

Student:

Ana Milić

Zagreb, 2021.

Izjavljujem da sam ovaj rad izradila samostalno koristeći znanja stečena tijekom studija i navedenu literaturu.

Zahvaljujem mentoru doc. dr. sc. Tomislavu Stipančiću na ukazanom povjerenju i stručnim savjetima pri izradi ovog rada. Također se zahvaljujem Borisu Selešu, ing. stroj., na pruženoj mogućnosti pri svladavanju uvoda u BIM.

Posebno se zahvaljujem Ivanu Paradžiku, mag. ing. mech., i Branislavu Radanoviću, mag. ing. el., te ostalim inženjerima tvrtke Vertiv Croatia na pruženim računalnim alatima i korisnim savjetima prilikom izrade ovog rada.

Najveće zahvale upućujem svojoj obitelji, Antunu, prijateljima, te obitelji Gadžić na strpljenju i bezuvjetnoj podršci.

Ana Milić



SVEUČILIŠTE U ZAGREBU  
FAKULTET STROJARSTVA I BRODOGRADNJE



Središnje povjerenstvo za završne i diplomske ispite  
Povjerenstvo za diplomske radove studija strojarstva za smjerove:  
proizvodno inženjerstvo, računalno inženjerstvo, industrijsko inženjerstvo i menadžment,  
inženjerstvo materijala te mehatronika i robotika

Sveučilište u Zagrebu Fakultet strojarstva i brodogradnje	
Datum:	Prilog:
Klasa:	602-04/21-6/1
Ur. broj:	15-1703-21

## DIPLOMSKI ZADATAK

Student: **ANA MILIĆ** Mat. br.: 0035201359

Naslov rada na hrvatskom jeziku: **Automatiziranje razvojnih procesa unutar BIM programske podrške**

Naslov rada na engleskom jeziku: **Automation of development processes within the BIM software**

### Opis zadatka:

BIM metodologija (engl. Building Information Modelling - BIM) sve je zastupljenija pri projektiranju različitih struktura i građevinskih objekata te kod praćenja toka informacija u različitim fazama razvoja projekta. Osim toga, omogućuje izbjegavanje kolizija te paralelni razvoj različitih sustava uključujući HVAC – (grijanje, ventilacija, klimatizacija), kanalizaciju te električne, plinske i vodovodne instalacije. Kako bi se izbjegli ponavljajući procesi, uštedjeli inženjerski resursi te povećala efikasnost, projektiranje navedenih sustava je moguće računalno automatizirati.

U radu je potrebno osmisliti i napraviti programsko rješenje za automatizaciju procesa 3D modeliranja rashladnog sustava podatkovnog centra od unutarnje jedinice do izlaza iz zgrade. Programskim rješenjem treba obuhvatiti plinsku i tekuću stranu rashladnog kruga, dobavu sanitarne vode za napajanje ovlaživača te odvod kondenzata. Programsko rješenje potrebno je temeljiti na Autodesk Dynamo ekstenziji koja se može uključiti kao dio BIM/Revit programske okoline za modeliranje.

U radu je potrebno navesti korištenu literaturu te eventualno dobivenu pomoć.

Zadatak zadan:  
30. rujna 2021.

Rok predaje rada:  
2. prosinca 2021.

Predviđeni datum obrane:  
13. prosinca do 17. prosinca 2021.

Zadatak zadao:  
doc. dr. sc. Tomislav Stipančić

Predsjednica Povjerenstva:  
prof. dr. sc. Biserka Runje

## SADRŽAJ

SADRŽAJ .....	I
POPIS SLIKA .....	II
POPIS TABLICA.....	III
POPIS KRATICA .....	IV
SAŽETAK.....	V
SUMMARY .....	VI
1. UVOD.....	1
2. MODELIRANJE INFORMACIJA O GRAĐEVINAMA (BIM) .....	2
2.1. Općenito o BIM-u .....	2
2.1.1. Razine zrelosti BIM-a .....	5
2.1.2. BIM model .....	6
2.1.2.1. Elementi BIM modela.....	7
2.2. Razvoj BIM-a.....	9
2.2.1. BIM softver .....	11
2.2.2. Razmjena informacija i kolaboracija .....	12
2.2.3. BIM projektni sudionici .....	14
3. VIZUALNO PROGRAMIRANJE .....	16
3.1. Primijenjeni alati .....	16
3.1.1. Revit.....	16
3.1.2. Dynamo.....	18
3.1.3. Python .....	20
4. PROGRAMSKO RJEŠENJE .....	21
4.1. Definiranje i sortiranje podataka modela .....	22
4.2. Obrada i izrada modela .....	34
4.3. Rezultat programskog rješenja.....	43
5. ZAKLJUČAK.....	47
LITERATURA.....	48
PRILOZI.....	50

## POPIS SLIKA

Slika 1. Uporaba BIM-a tijekom životnog vijeka objekta [4].....	4
Slika 2. Razvojne razine zrelosti BIM-a [8].....	6
Slika 3. Dimenzije informacija BIM modela [11].....	7
Slika 4. Ventil familija u Revitu.....	8
Slika 5. Sučelje <i>Family Types</i> u Revitu.....	9
Slika 6. Archicad sučelje iz 1984. godine [15] .....	10
Slika 7. Povijest BIM definicije [6] .....	11
Slika 8. Dijagram toka razmjene informacija i kolaboracija [6] .....	13
Slika 9. Hijerarhijski prikaz BIM uloga [6] .....	14
Slika 10. Usporedba standardnih uloga s BIM ulogama na projektu [6] .....	15
Slika 11. Korisničko sučelja u Revitu .....	17
Slika 12. Dynamo sučelje.....	19
Slika 13. Prikaz Dynama i Dynamo Playera unutar alatne trake Revita.....	19
Slika 14. Python skripta u Dynamu.....	20
Slika 15. Prikaz algoritma za definiranje ulaznih podataka modela u dijagramu toka .....	23
Slika 16. Definiranje strojarskih instalacija u Dynamu.....	24
Slika 17. Algoritam za preuzimanje geometrije cijevi .....	25
Slika 18. Algoritam za sortiranje strojarskih cijevi po udaljenosti .....	25
Slika 19. Sortiranje cijevi po udaljenosti Python skriptom.....	26
Slika 20. Python skripta za razvrstavanje iznosa cijevi .....	27
Slika 21. Prikaz algoritma izlaznih podataka modela .....	28
Slika 22. Algoritam izvoza podataka modela u Excel .....	29
Slika 23. Algoritam preuzimanja geometrije ventila .....	30
Slika 24. Algoritam izvoza podataka ventila u Excel .....	31
Slika 25. Izlazni podaci početnih točki cijevi standardnog modela u Excelu .....	32
Slika 26. Izlazni podaci krajnjih točki cijevi standardnog modela u Excelu .....	32
Slika 27. Izlazni podaci vektora cijevi standardnog modela u Excelu.....	33
Slika 28. Izlazni podaci ventila standardnog modela u Excelu.....	34
Slika 29. Prikaz algoritma za obradu i izradu modela u dijagramu toka .....	35
Slika 30. Prikaz algoritma za uvoz podatka iz Excela .....	36
Slika 31. Algoritam za razvrstavanje početnih točki cijevi.....	37
Slika 32. Algoritam za razvrstavanje krajnjih točki cijevi .....	38
Slika 33. Algoritam za razvrstavanje geometrijskih karakteristika ventila.....	38
Slika 34. Algoritam za kreiranje strojarskih cijevi.....	39
Slika 35. Algoritam za povezivanje strojarskih instalacija .....	40
Slika 36. Prvi dio Python skripte za spajanje strojarskih instalacija.....	41
Slika 37. Drugi dio Python skripte za spajanje strojarskih instalacija .....	42
Slika 38. Prikaz generiranog modela u 2D obliku .....	43
Slika 39. Prikaz programskog rješenja 3D BIM modela unutar Revita .....	44
Slika 40. Prikaz Dynamo Playera u sučelju Revita .....	45
Slika 41. Prikaz odabranih ulaznih podataka algoritma u Dynamo Playeru .....	46

## POPIS TABLICA

Tablica 1. Korišteni paketi u Dynamu .....	22
--	----



**POPIS KRATICA**

BIM	<i>Building Information Modelling</i> – Modeliranje informacija o građevinama
3D	<i>Trodimensionalna veličina</i>
2D	<i>Dvodimensionalna veličina</i>
HVAC	<i>Heating, ventilation and air conditioning</i> – Grijanje, ventilacija i klimatizacija
CDE	<i>Common Data Environment</i> – Zajednički izvor podataka
IFC	<i>Industry Foundation Class</i> – Međunarodne klase zaklade
COBie	<i>Construction Operations Building Information Exchange</i> – Razmjena informacija o građevinskim operacijama zgrade
CAD	<i>Computer Aided Design</i> – Računalom potpomognuto oblikovanje
BDS	<i>Building Description System</i> – Sustav za opis građevine
GLIDE	<i>Graphical Language for Interactive Design</i> – Grafički jezik za interaktivne konstrukcije
MEP	<i>Mechanical, electrical and plumbing</i> – Mehaničke, električne i vodoinstalacije

## SAŽETAK

Sveprisutna je intenzivna potreba za primjenom BIM metodologije (eng. *Building Information Modeling*) pri projektiranju građevinskih objekata. Navedena metodologija se primjenjuje od najranije faze projektiranja zbog oslanjanja na 3D BIM model građevine, koji je bogat podacima i informacijama. BIM kao sustav omogućuje multidisciplinarnom projektnom timu paralelni razvoj različitih projektnih sustava, uključujući i strojarske instalacije grijanja, ventilacije i klimatizacije (eng. *Heating, ventilation and air conditioning, HVAC*). Na taj se način sprječava kolizija projektnih sustava različitih disciplina pri cjelokupnom razvoju projekta. Projektiranje instalacijskih sustava moguće je računalno automatizirati kako bi se izbjegli ponavljajući procesi, te na taj način uštedjeli inženjerski resursi i povećala učinkovitost.

U okviru ovog rada razrađeno je programsko rješenje za automatizaciju procesa 3D modeliranja rashladnog sustava podatkovnog centra. Sklop rashladnog sustava sastoji se od unutarnje jedinice te mreže cjevovodnih instalacija do izlaza iz zgrade. Cjevovodne instalacije obuhvaćaju plinsku i tekuću stranu rashladnog kruga, dobavu sanitarne vode za napajanje ovlaživača te odvod kondenzata. Pritom se, unutar BIM programske podrške Revit, primjenjuje Autodesk Dynamo ekstenzija za modeliranje predmetnog projektnog rješenja.

Ključne riječi: BIM, automatizacija procesa, 3D modeliranje, Revit, Dynamo

## **SUMMARY**

There is an ubiquitous need for the application of BIM methodology (Building Information Modeling) in the design of buildings. This methodology is applied in the earliest phase of design due to the reliance on the 3D BIM model of the building, which provides relevant data and information. As a system, BIM enables a multidisciplinary project team to model many different systems simultaneously, including heating, ventilation and air conditioning (HVAC). This prevents possible collisions of modeled systems of different disciplines in the overall project development. Modeling of systems can be computer-automated in order to avoid repetitive processes, thus saving engineering resources and increasing efficiency.

In this thesis, a software solution for automation of the 3D modeling process of the data center cooling system assembly was developed. The cooling system assembly consists of an indoor cooling unit and corresponding piping distribution, which layouts between cooling unit and data center exit. Piping distribution include gas and liquid phase of the cooling circuit, the supply of sanitary water for the humidifier and the condensate drain. Autodesk Dynamo extension within the BIM software Revit is used to model complete automation process of the data center cooling system assembly.

Key words: BIM, automation of processes, 3D modeling, Revit, Dynamo

## 1. UVOD

BIM softveri sve su zastupljeniji alat pri projektiranju građevinskih objekata zbog same složenosti projekata, ali i povećanja projektnih ciljeva, uzrokovanih dinamikom suvremenog razvoja tehnologije. Programska rješenja u sklopu BIM-a za upravljanje cjelokupnim životnim ciklusom proizvoda nude fleksibilnost kod razvoja ideja i njihove implementacije, uz efikasno dijeljenje resursa te istovremeni rad projekatana različitih disciplina. Osim toga, dostupnost velike količine informacija u 3D BIM modelu, na kojima se temelji i oslanja BIM metodologija, omogućuje automatizaciju dijelova procesa projektiranja. Na taj način je moguće pojednostaviti projektiranje i povećati učinkovitost na način da se izbjegavaju ponavljajući procesi, što u konačnici rezultira uštedom inženjerskih resursa, uz ispunjenje svih ekoloških, socioekonomskih i konstrukcijskih projektnih ciljeva [1 , 2]. Stoga se u okviru ovog diplomskog rada, unutar BIM softvera Revit, primjenjuje Dynamo kao Autodesk ekstenzija za vizualno programiranje. Dynamo omogućuje izvoz parametarskih i drugih podataka BIM elemenata, pohranjenih u Revit modelu.

Diplomski rad se sastoji od pet poglavlja, od kojih je uvodno poglavlje prvo. U drugom poglavlju diplomskog rada opisan je osnovni koncept BIM-a i njegov razvoj uz razmjenu informacija i kolaboraciju između multidisciplinarnih projektnih timova. Osim toga, u trećem poglavlju opisani su primijenjeni programski alati Revit, Dynamo i Python. U četvrtom poglavlju je objašnjeno i provedeno programsko rješenje za automatizaciju procesa 3D modeliranja rashladnog sustava podatkovnog centra primjenom Dynama unutar BIM programske okoline Revit. Temeljem dobivenih rezultata programskog rješenja iznesen je zaključak o automatizaciji razvojnih procesa unutar BIM programske podrške u posljednjem, petom poglavlju.

## 2. MODELIRANJE INFORMACIJA O GRAĐEVINAMA (BIM)

### 2.1. Općenito o BIM-u

Modeliranje informacija o građevinama, odnosno *Building Information Modeling* (BIM), predstavlja suvremen, logičan i cjelovit softver i proces uz uporabu specifičnih alata pri projektiranju, izgradnji, upravljanju i održavanju objekta tijekom životnog vijeka [1, 3]. Na slici 1 prikazana je uporaba BIM-a tijekom cijelog uporabnog vijeka [4]. Nadalje, BIM sustav su inteligentni modelirani objekti koji osim vizualnog prikaza sadrže sve potrebne projektne informacije pohranjene u zajedničkoj centralne bazi podatak. Tako je moguće pružiti neograničen i istovremeni pristup informacija cijelom multidisciplinarnom projektnom timu. Stoga BIM proces od projektnog tima zahtijeva visoku razinu iskustva i znanja kako bi se spriječili problemi implementacije [2, 3, 5].

Postoje neke praktične primjene BIM pristupa [6]:

#### 1. Planiranje

- Prostorne analize lokacija građevine (eng. *Site Analysis*)
- Prostorna analiza interpolacije građevine (eng. *Programming*)
- Snimanje postojećeg stanja (eng. *Existing Conditions Modeling*)

#### 2. Projektiranje

- Analiza objektnog sustava (eng. *Building System Analysis*).
- Projektiranje i dizajn (eng. *Design Authoring*)
- Inženjerske analize (eng. *Engineering Analysis*)
  - Analiza energetske učinkovitosti (eng. *Energy Analysis*)
  - Analiza nosive konstrukcije (eng. *Structural Analysis*)
- 3D koordinacija (eng. *3D Coordination*)
- Procjene sukladnosti s propisima (eng. *Code Validation*)
- Procjene održivosti (eng. *Sustainability Evaluation*)
- Pregled uspješnosti projektnog rješenja (eng. *Design Reviews*)

- Procjena troškova (eng. *Cost Estimation - 5D Modeling*)

### 3. Građenje

- Vremensko planiranje (eng. *Phase Planning - 4D Modeling*)
- Planiranje i organizacija gradilišta (eng. *Site Utilization Planning*)
- Projektiranje sustava građenja (eng. *Construction System Design*)
- Digitalna kontrola proizvodnje (eng. *Digital Fabrication*)
- Geodetsko 3D upravljanje i planiranje (eng. *3D Control and Planning*)
- Snimanje izvedenog stanja (eng. *Record Modeling*)

### 4. Puštanje u pogon i upravljanje

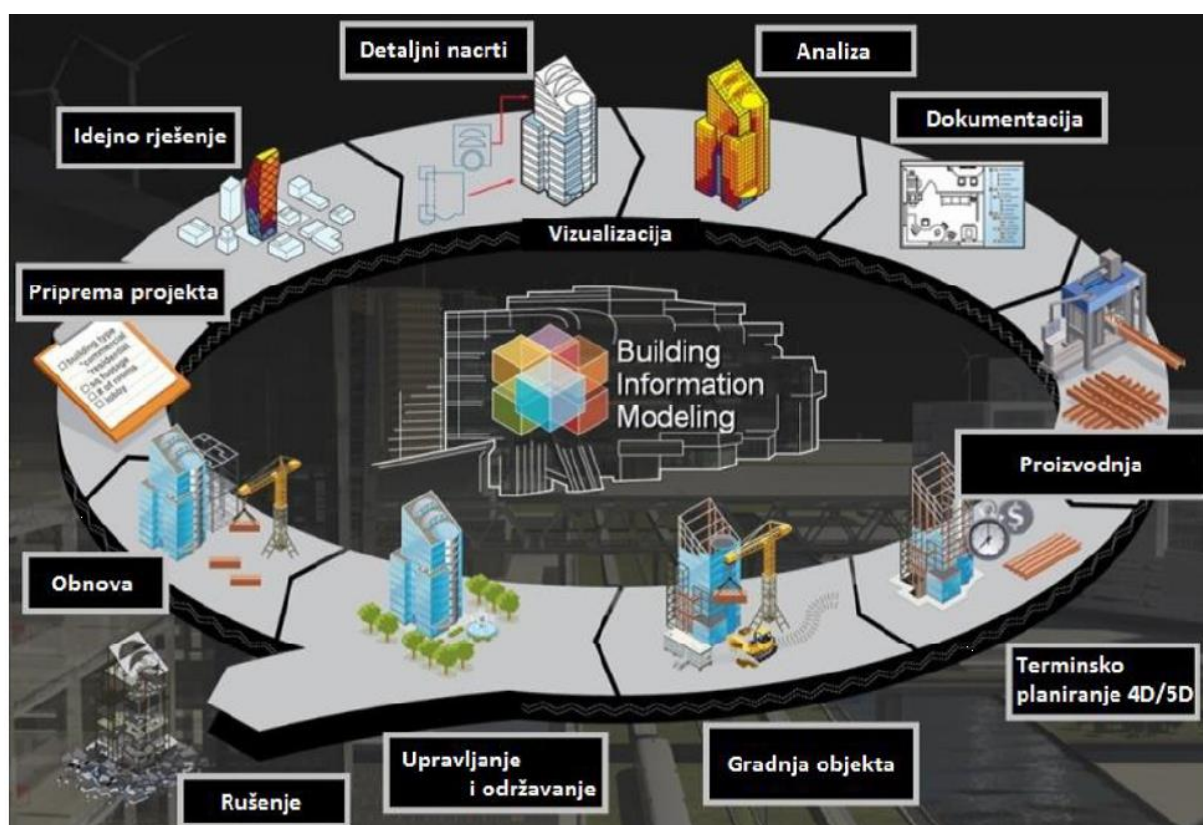
- Planiranje održavanja (eng. *Building (Preventative) Maintenance Scheduling*)
- Upravljanje imovinom (eng. *Asset Management*)
- Upravljanje i praćenje prostora (eng. *Space Management and Tracking*)
- Planiranje izvanrednih nepogoda (eng. *Disaster Planning*).

Drugim riječima, BIM pristup koristi se tijekom cjeloživotnog ciklusa projekta, te se time omogućuje [6]:

- Podršku tijekom predlaganja odluka u projektnim procesima.
- Rano sprječavanje pogrešaka na projektu.
- Detaljne analize.
- Efikasne izmjene i jednostavnije upravljanje promjenama.
- Vidljivije shvaćanje projektnih ciljeva i odnosa sa svim sudionicima na projektu.
- Vizualizaciju projektnih rješenja.
- Unaprjeđenja u projektiranju i koordinaciji projekata.
- Povećanje i osiguravanje kvalitete građevinskih procesa i konačnog proizvoda.
- Učinkovitost procesa tijekom faze građenja.

- Povećanje sigurnosti kroz sve životne faze građevine.
- Podrška tijekom analize životnog ciklusa i projektnih troškova.

Temeljem prethodno navedenog proizlazi da se BIM pristup može opisati kao tehnologija i kao metodologija. BIM kao tehnologija predstavlja virtualnu simulaciju fizičkih i konfiguracijskih atributa objekta za projektiranje, razne proračune, analize, simulacije i izvješća pomoću informativnog 3D BIM model. Osim toga, BIM kao metodologija pruža suradnju multidisciplinarnim timovima oslanjajući se na dostupan podacima bogat model kroz različite životne faze objekta [6, 7].



Slika 1. Uporaba BIM-a tijekom životnog vijeka objekta [4]

### 2.1.1. Razine zrelosti BIM-a

Različite razine zajedničke suradnje, odnosno razine zrelosti BIM-a koje su prikazane na slici 2, predstavljaju raspon razina razmjene informacija između sudionika na konkretnom modelu projekta. Upotreba jednog BIM modela za sve projektne sudionike ostvaraju se postupno.

Sve atraktivniji koncept suradničkog načina rada inzistira na poznavanju 4 stupnja zrelosti BIM-a [8]:

#### 1. Razina 0 BIM

Predstavlja nisku razinu suradnje gdje se razmjena informacija određenog projekta odvija pomoću papirnatih oblika tehničke dokumentacije.

#### 2. Razina 1 BIM

Na ovoj djelomičnoj razini zrelosti koristi se podatkovno okruženje (eng. *Common Data Environment* – CDE) sa zajedničkom bazom podataka projekta gdje su definirane određene uloge i odgovornosti sudionika procesa.

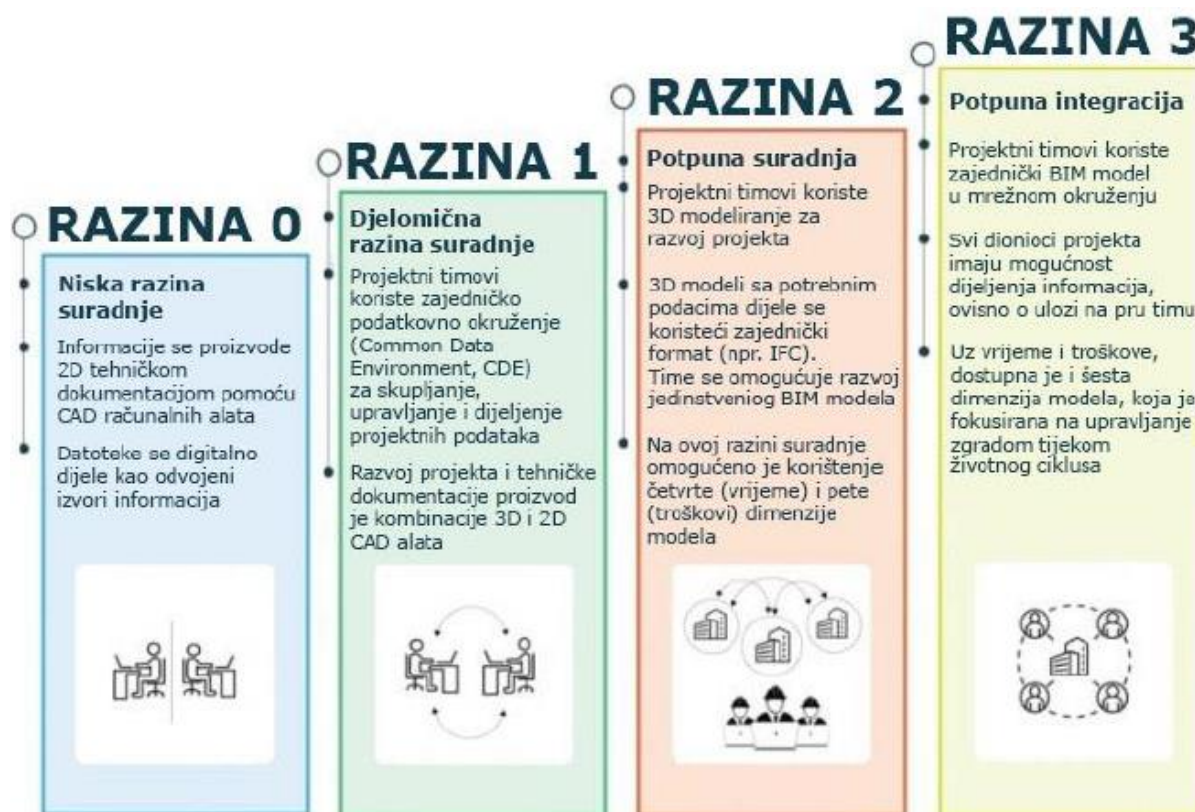
#### 3. Razina 2 BIM

Ova razina predstavlja potpunu razinu na kojoj se provodi BIM suradnja. Svim sudionicima omogućena je upotreba vlastitog modela uz mogućnost prijenosa podatka pomoću zajedničkog projektnog formata, poput IFC (*Industry Foundation Class*) ili COBie (*Construction Operations Building Information Exchange*) te drugi.

#### 4. Razina 3 BIM

Implementacija ove potpune integracijske razine omogućena je u naprednijim kompanijama. Karakterizira je potpuna kolaboracija svih struka pomoću zajedničkog centralnog modela. Na taj je način moguće upravljati podacima o projektu tijekom cjeloživotnog vijeka trajanja.

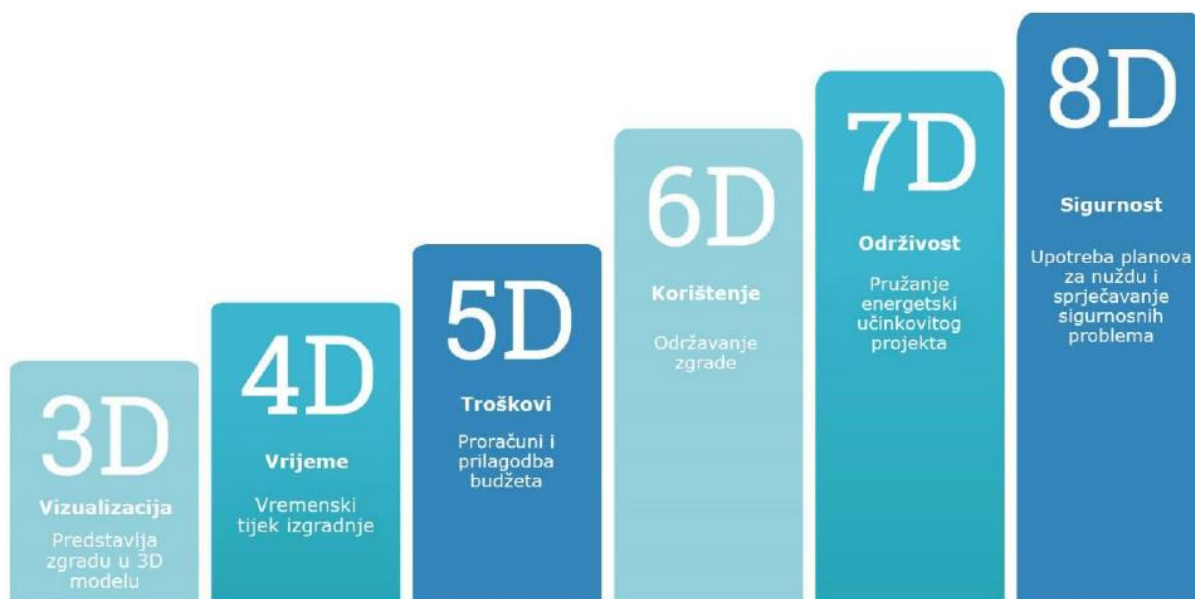




Slika 2. Razvojne razine zrelosti BIM-a [8]

### 2.1.2. BIM model

BIM modeli sadrže sve potrebne informacije, pomoću BIM elemenata, koje se odnose na projektne faze projektiranja, izgradnje, upravljanja i održavanja. Zamjena CAD tehnologije BIM tehnologijom rezultat je konstantnog razvoja tehnologije. Kako BIM tehnologija podrazumijeva nadopunjavanje informacija kroz sve faze BIM procesa, postoji nekoliko dimenzija informacija BIM model koje će se pobliže objasniti u nastavku teksta [6, 9, 10].



Slika 3. Dimenzije informacija BIM modela [11]

Najčešći tip BIM modela je zajednički vizualni informacijski model 3D BIM koji osim stvaranja grafičkih i drugih projektnih informacija poboljšava multidisciplinarnu suradnju korisnika. Nadopunjavanje 3D BIM modela s vremenskim komponentama, kojim se olakšava upravljanje rasporedom zadataka na projektu, dodaje se dodatna 4D dimenzija. S druge strane, peta dimenzija BIM modela se odnosi na definiranje količine materijala potrebna za modeliranje BIM elemenata, analize proračuna i procjenu troškova tijekom razvoja projekta. Osim toga, šesta dimenzija BIM modela značajna je za održavanja objekta uz osiguravanje niže potrošnje energije provođenjem detaljne analize energetske učinkovitosti, dok je sedma dimenzija BIM modela zadužena za ekološku, socioekonomsku i tehničku održivost projekta. Najnoviji 8D BIM model služi za prevenciju sigurnosnih problema uz upotrebu planova za hitne slučajeve [9, 12, 13].

#### 2.1.2.1. Elementi BIM modela

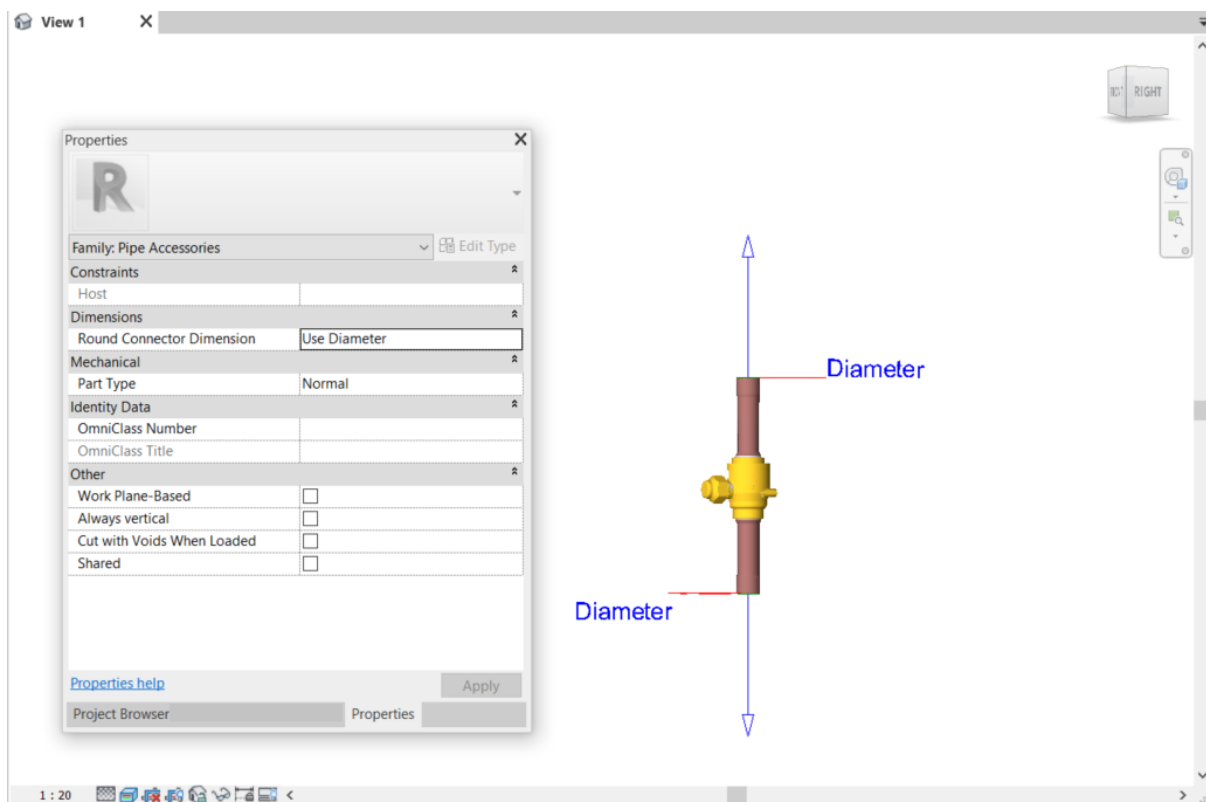
BIM modeli stvoreni su od skupine BIM elemenata (Familija) koji su prikazani kao 2D i 3D geometrija. BIM elementi mogu imati više tipova koji sadrže geometrijske i vizualne karakteristike, svojstva elemenata, detaljne informacije o proizvodu (cijena, ID proizvoda, kategorija, materijal, proizvođač itd.) i mnoge druge potrebne projektne informacije za izradu

tehničke dokumentacije i izgradnju samog objekta. U Revitu se nalazi baza s postojećim familijama, međutim, mogu se izraditi i vlastite familije. Neki od primjera BIM elemenata su strojarske komponente poput ventila, koljena, redukcija, strojarskih cijevi i mnoge druge komponente [6, 14].

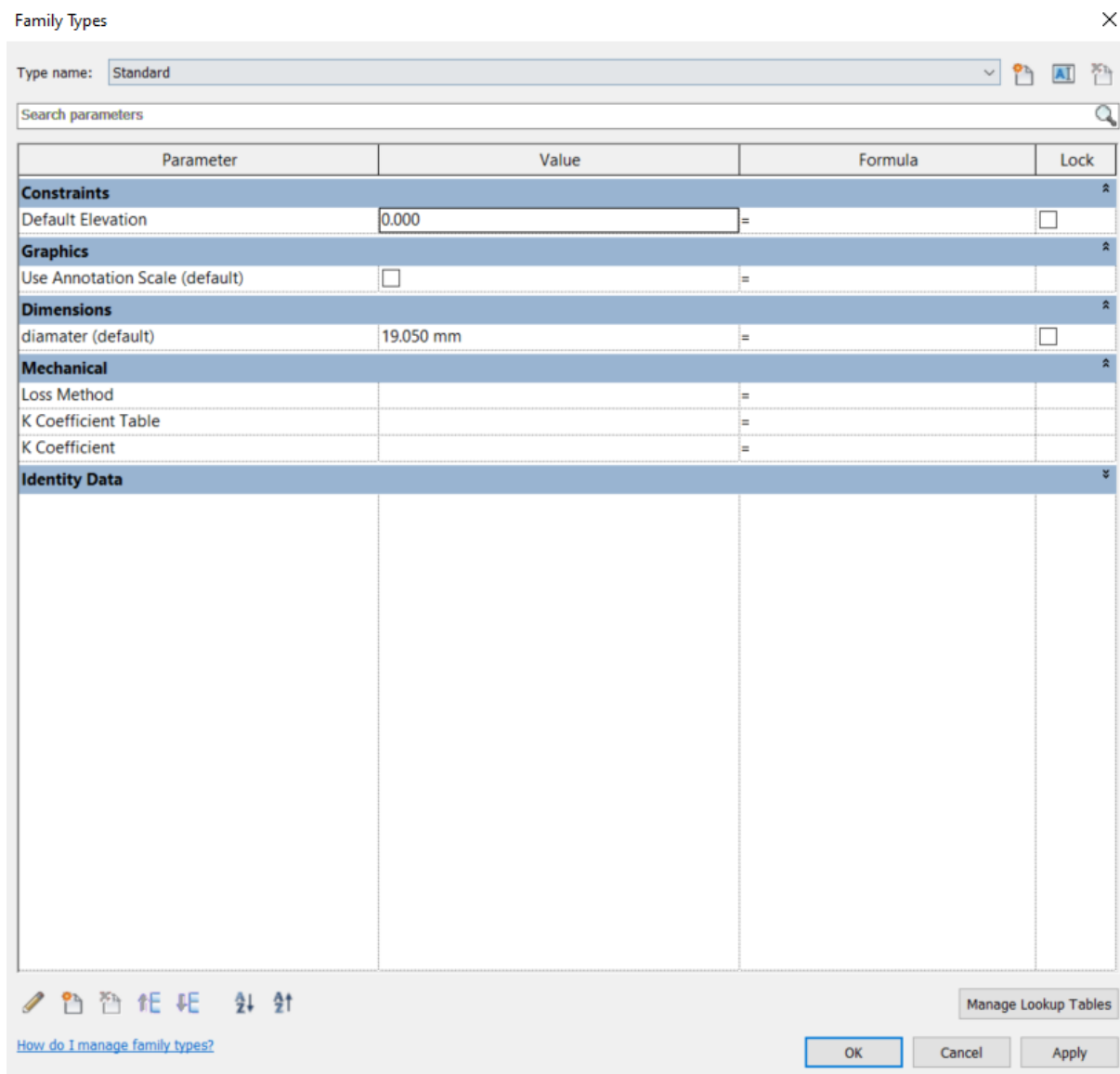
Minimalna struktura podataka BIM elemenata [6]:

- 3D geometrija
- 2D detalji
- zahtjevi za manipulativan prostor
- materijali i ostali parametri
- logički spojevi.

Na slici 4 prikazan je ventil sa svojim svojstvima u 3D obliku kao primjer familije u Revitu. Također, parametri familije mogu se uređivati, dodavati i povezivati pomoću formula u sučelju *Family Types*, koje je prikazano na slici 5.



Slika 4. Ventil familija u Revitu

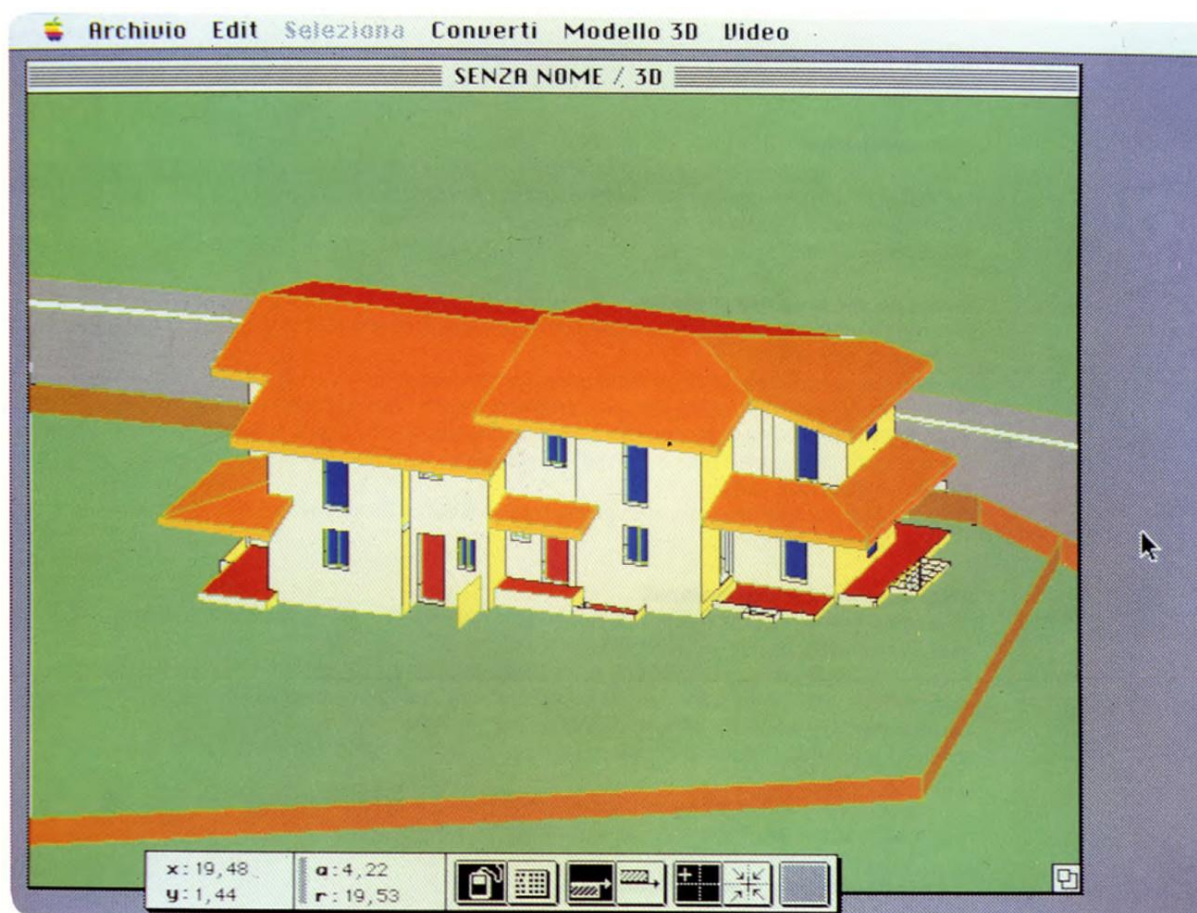
Slika 5. Sučelje *Family Types* u Revitu

## 2.2. Razvoj BIM-a

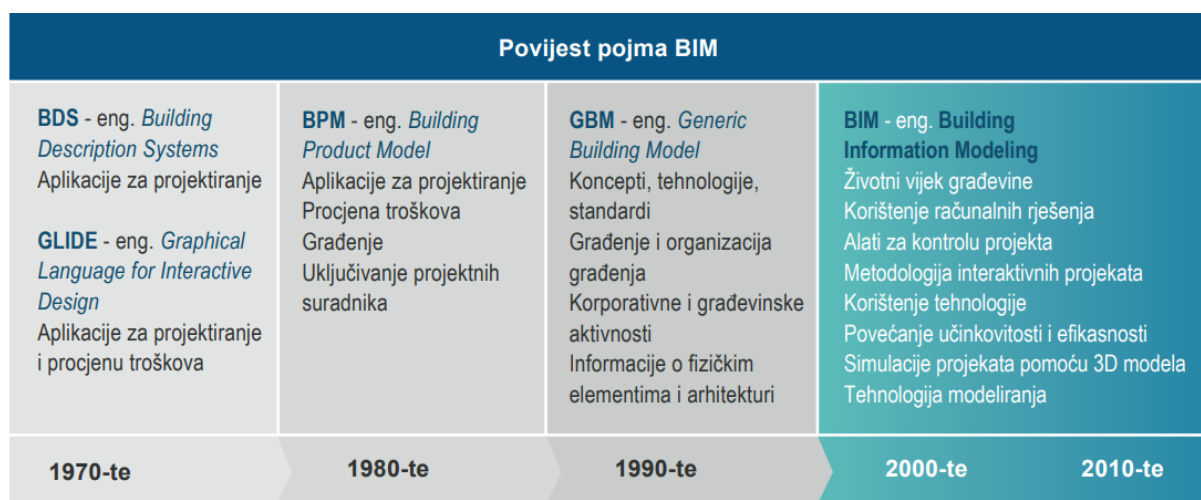
Današnje predstavljanje BIM-a kao cjeloviti pristupa koji mijenja i unapređuje multidisciplinarno objektno poslovanje premašuje početno značenje usmjereno na objekt. Povijesno gledajući, idejni koncepti BIM-a doživio je značajne promjene drugom polovicom 20-og stoljeća. U članku pod naslovom *Building Description System* (BDS), prvi put je 1975. Charles Eastman raspravljao o idejama parametarskog dizajna pomoću 3D prikaza koji sadrže integrirane objektnu bazu podataka. Također, dvije godine kasnije je razvio *Graphical Language for Interactive Design* (GLIDE), projekt koji sadrži većinu karakteristika modernog

BIM alata. Štoviše, najveće zasluge razvoju BIM alata pripadaju dvama programskim genijima, idejnim začetnicima Revita i Archicada, Leonid Raiz i Gábor Bojár [6, 15].

Prvi BIM alat dostupan na osobnom računalu je Archicad, čije je sučelje prikazano na slici 6, dok je razvoj BIM pojma prikazano na slici 7.



Slika 6. Archicad sučelje iz 1984. godine [15]



Slika 7. Povijest BIM definicije [6]

### 2.2.1. BIM softver

BIM softver sadržava prikaz fizičkih i funkcionalnih atributa građevine kao objektno orijentirani model povezan s bazom informacija uz mogućnost komunikacije s modelom prikazanim u 2D i 3D geometriji. Također, omogućuje prilagodbu nacrtu u skladu s razvojem modela prema potrebama projekta [6].

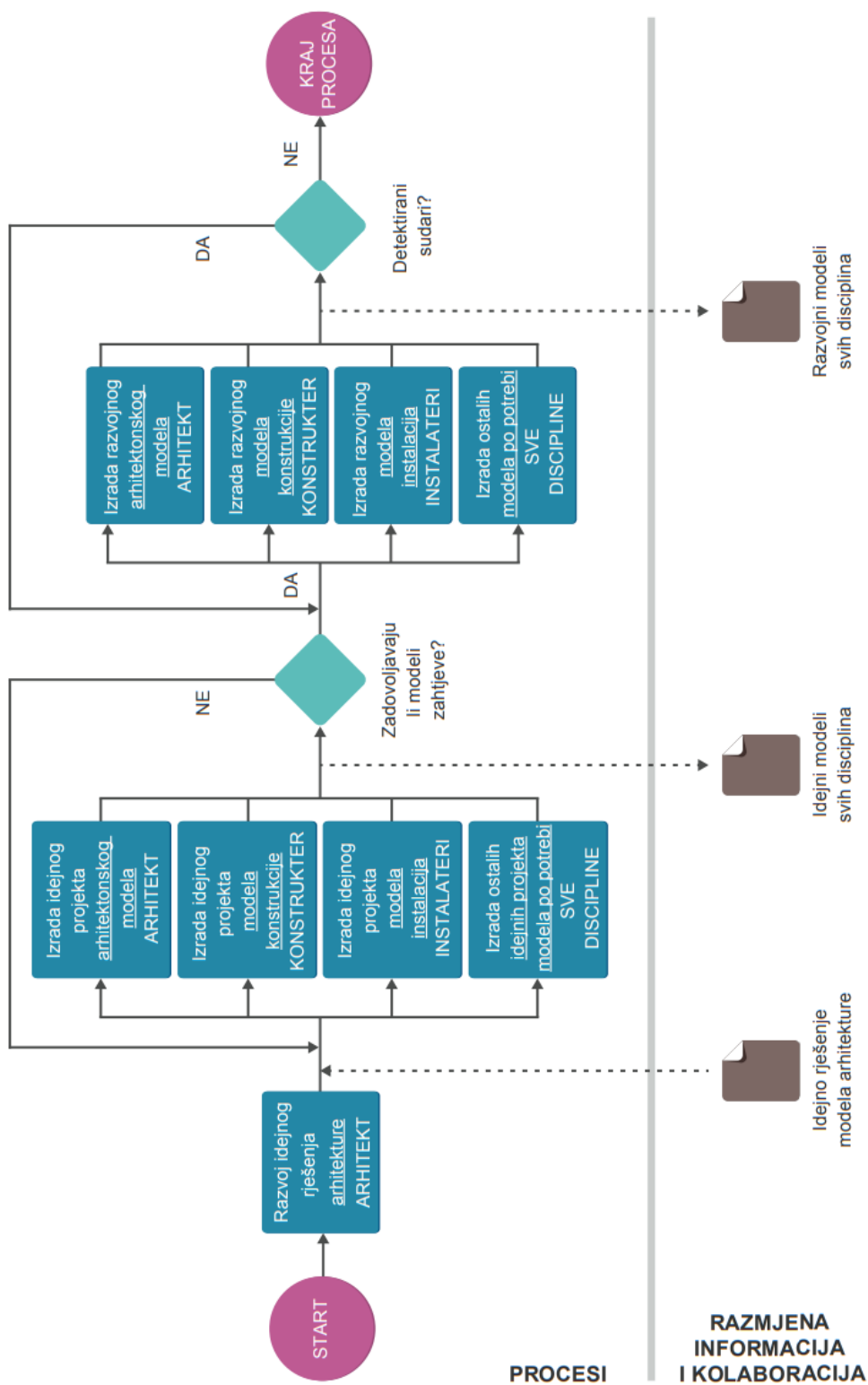
Kako bi BIM softverski alati isporučili integrirani model koji treba biti u skladu sa određenim standardima koje implementira BIM, mora se zadovoljiti sljedećih šest uvjeta BIM tehnologije [6, 7]:

- Pristup arhitektonskim, građevinskim, strojarskim i drugim disciplinama.
- Primjenjivost kroz sve objektne faze.
- Prostornost (3D).
- Poboľšane mogućnosti uvoza i izvoza dokumenata/modela korištenjem protokola i formata za razmjenu podataka.
- Udaljavanje od desktop-aplikacija.
- Uporaba alata koji podržavaju prikaz i izradu kompleksnijih detalja.

Jedan od BIM softvera koji pokriva većinu navedenih kriterija, uz još prostora za poboljšanjem uvjeta, je Revit [16].

### ***2.2.2. Razmjena informacija i kolaboracija***

Projektne podaci koji se mogu primjenjivati i ažurirati, dostupni su svim projektnim skupinama, te su pohranjeni na istoj lokaciji. Informacije unutar BIM projekta su ključan faktor koji utječe na efikasan i kontinuiran razvoj projekta. Razumijevanje korištenja projektnih informacija olakšava kolaboraciju između multidisciplinarnih timovima uz konstantno razvijanja baze podataka kroz određene cikluse projekta. Dijagram toka procesa koordinacije prikazan je na slici 8.

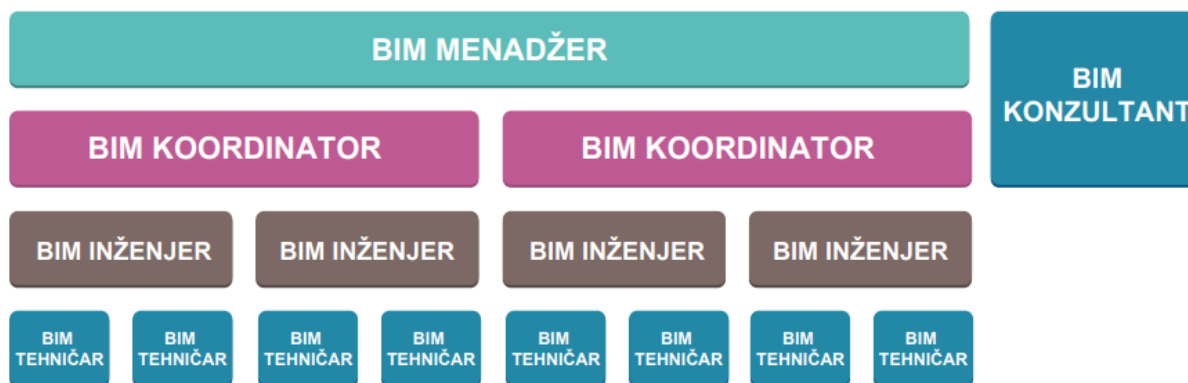


Slika 8. Dijagram toka razmjene informacija i kolaboracija [6]

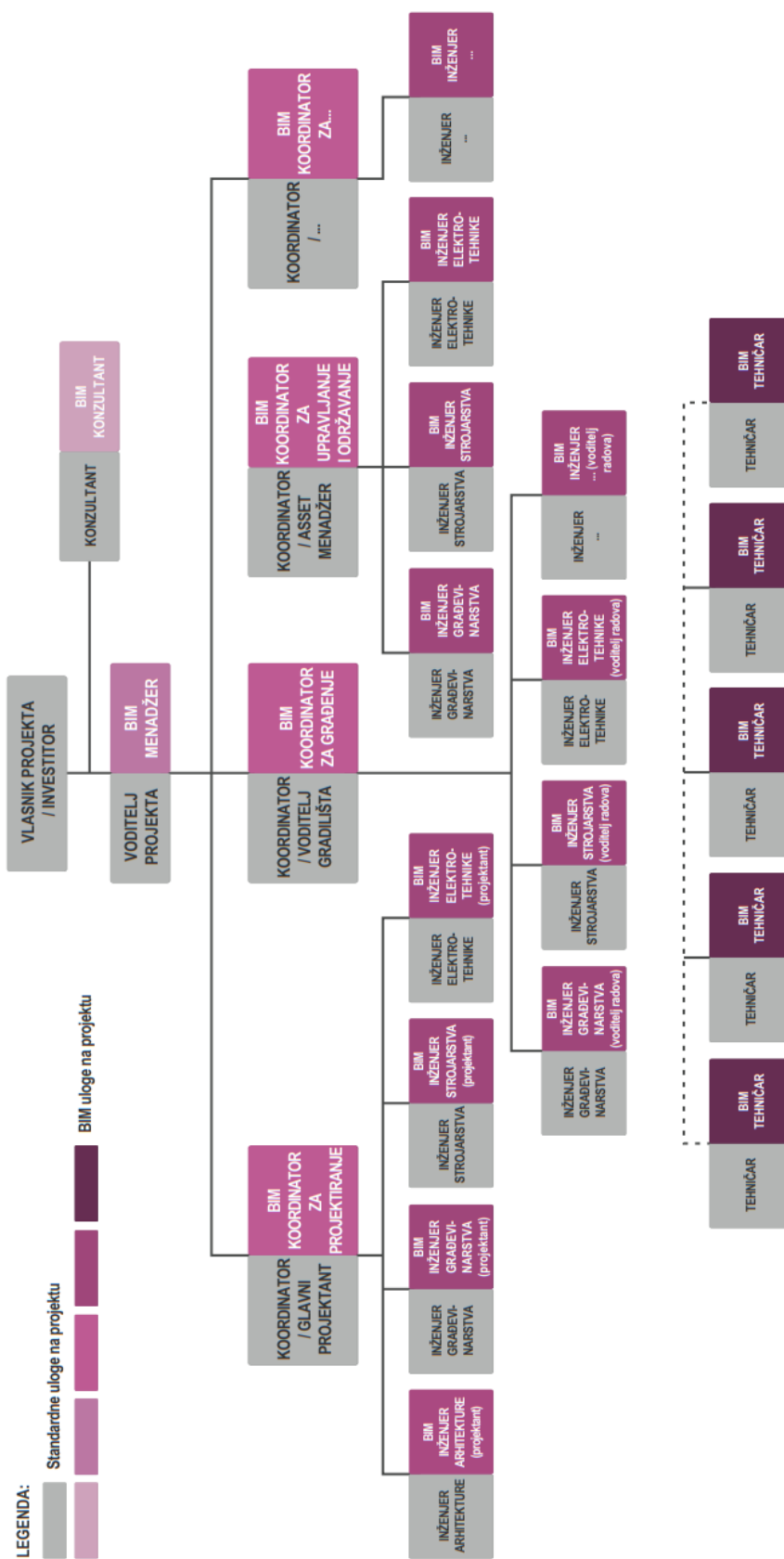


### 2.2.3. BIM projektne sudionici

Primjenom BIM pristupa zahtijevaju se nove projektne dužnosti koje donose specifične odgovornosti, detaljnije opis i prikaz slijedi u nastavku. BIM menadžer definira ciljeve i zahtjeve projekta, uz skupljanje i razmjenjivanje projektnih informacija. Također određuje pravila kojih se treba pridržavati tijekom cijelog životnog vijeka građevine. S druge strane BIM konzultant vodi i konzultira različite timove koji u svojim timovima nemaju BIM stručnjaka, te tek usvajaju BIM pristup. Također, BIM koordinatori, koji su predodređeni za uže područje određene struke, su izravna veza između BIM menadžera i ostalih sudionika. BIM koordinatori su stručnjaci za upravljanje i modeliranje informacija u pravom obliku pomoću odgovarajućih programa. Osim njih, BIM inženjeri razvijaju model i tehničku dokumentaciju projekta uporabom odgovarajućeg BIM softvera, dok je BIM tehničar stručnjak za modeliranje tehničkih i funkcionalnih sklopova u BIM softverima uz odlično poznavanje pojedinih strukovnih područja. Hijerarhijska struktura BIM uloga prikazana je na slici 9, dok je usporedba standardnih i BIM uloga na projektu prikazana na slici 10.



Slika 9. Hijerarhijski prikaz BIM uloga [6]



Slika 10. Usporedba standardnih uloga s BIM ulogama na projektu [6]

### 3. VIZUALNO PROGRAMIRANJE

Vizualno programiranje je metoda programiranja u kojoj se korisniku omogućuje stvaranje algoritma grafičkim elementima, od kojih svaki ima određene ulazne i izlazne parametre, umjesto tekstualnog unosa koda. Zbog svoje preglednosti, pristupačnosti i smanjenju apstrakcije kodiranja, vizualno programiranje postaje alat koji omogućava ubrzano svladavanje programiranja, u svrhu postizanja određene razine automatizacije u projektu. Stoga se u okviru ovog diplomskog rada koristi Dynamo BIM, integrirani alat u računalnom alatu Autodesk Revit. Nadalje, neke naredbe je komplicirano ostvariti u grafičkom sučelju, te se zbog toga omogućuje upotreba tekstualnih programskih jezika poput Pythona, koji se primjenjuje u ovom radu unutar Dynama. Programi za vizualno programiranje služe za izvoz, pohranu te konverziju podataka iz matičnog softvera koje se mogu upotrijebiti u drugim inženjerskim alatima [17].

#### 3.1. Primijenjeni alati

##### 3.1.1. Revit

Revit je tehnološki najnapredniji BIM alat razvijen od strane Autodesk. Prva verzija Revita izdana 2000. godine, služila je kao alat arhitektima za izgradnju trodimenzionalnih objekata uz definirane informacije. Nakon što ga je 2002. kupio Autodesk, uslijedila su razna poboljšanja samog alata poput dodavanja *Worksetova*, odnosno specifičnih područja rada, čime se omogućuje istovremena suradnja različitih inženjerskih disciplina na zajedničkom BIM modelu u realnom vremenu. BIM model u Revitu čine familije, parametarski elementi koji mogu biti odabrani i automatizmom, ispunjene informacijama kojima se može upravljati u svrhu prilagodbe zahtijevanim uvjetima projekta [18, 19].

Revit zajedno čine [20]:

- Revit Arhitektura (eng. *Revit Architecture*)

Program baziran na stvaranje i održavanje dizajna modela uz suradnju s drugim sudionicima u projektiranju tijekom razvoja projekta. Jedna od glavnih prednosti programa je automatsko ažuriranje cijelog projekta i projektne dokumentacije prilikom lokaliziranih promjena na

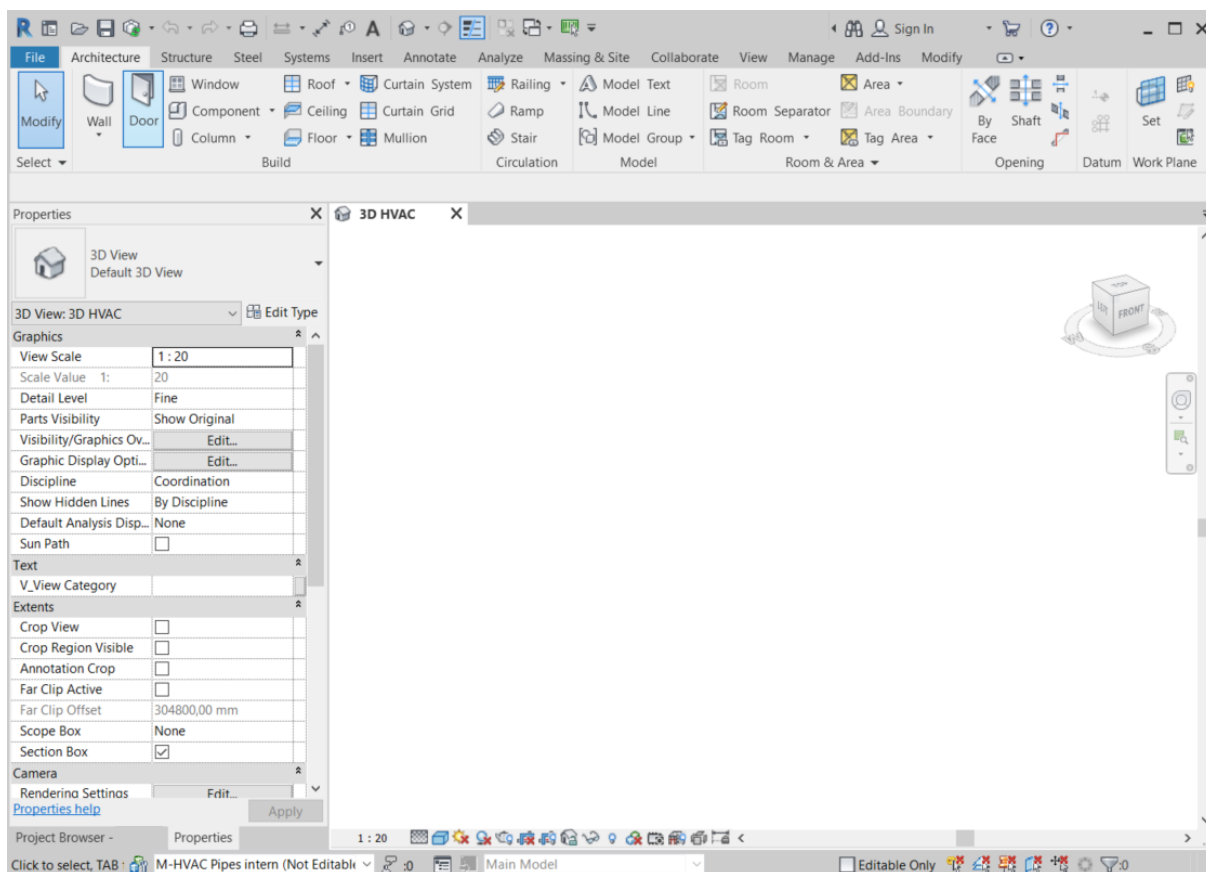
modelu. Time se olakšava koordinacija, poboljšava preciznost i ubrzava projektiranje konstrukcije u Revitu okruženju.

- Revit Statika (eng. *Revit Structure*)

Uporabom BIM sustava Revit Statika povlači tehničku dokumentaciju iz zajedničkog centralnog modela. Osim toga, Revit Statika istovremeno omogućuje konstruiranje fizičkog modela objekta i provođenje analize istog. Zbog toga osigurava uštedu vremena prilikom projektiranja i analiziranja konstrukcije, te poboljšava suradnju između drugih inženjera na projektu.

- Revit Strojarsstva i Elektrotehnike (eng. *Revit MEP*)

Program namijenjen za projektiranje strojarskih i električne instalacije. Također, omogućuje provjeru kolizija elemenata između svih tehničkih struka. Stoga se smanjuje broj pogrešaka na modelu što rezultira ekonomskim uštedama tijekom faze izvođenja projekta.



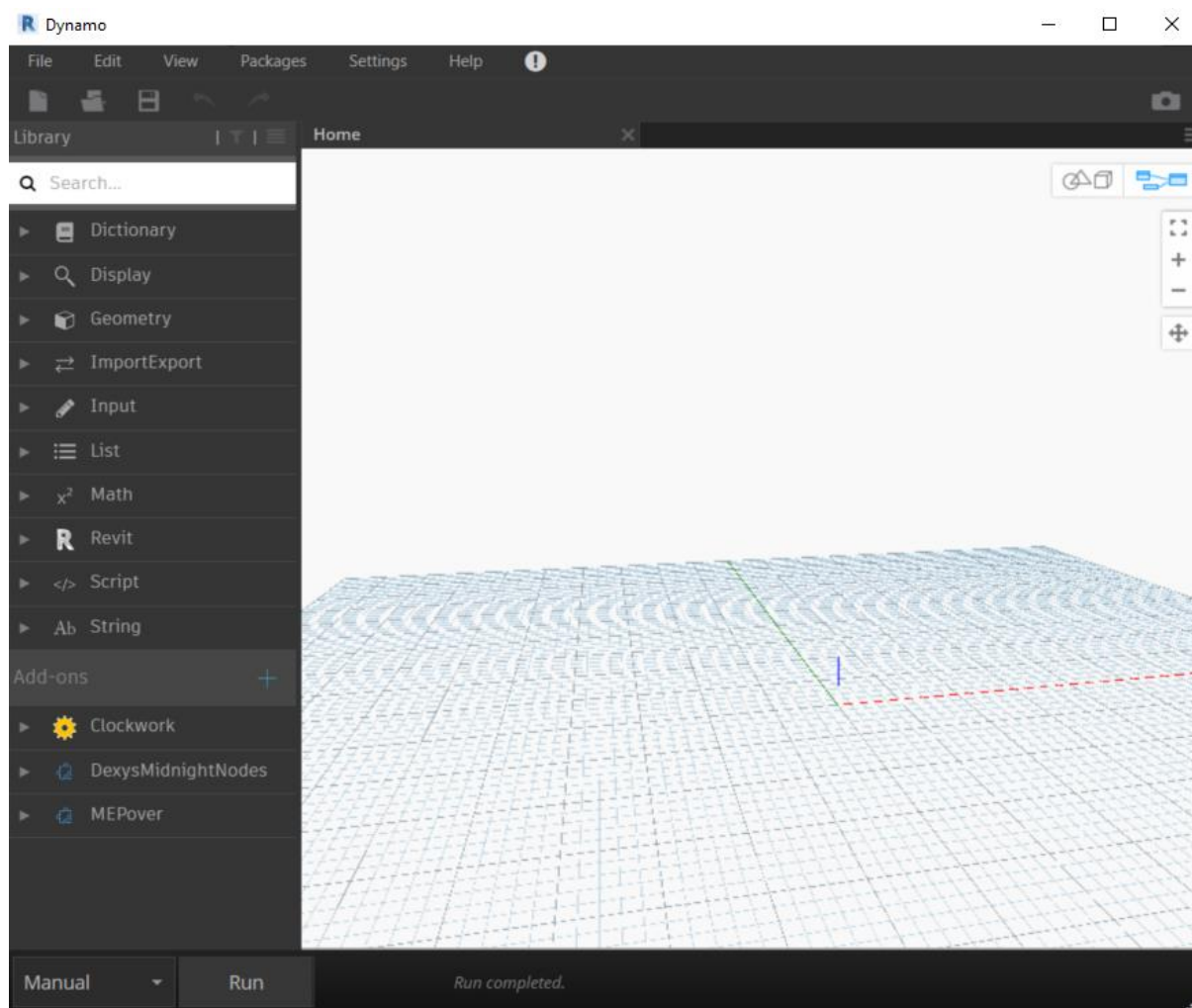
Slika 11. Korisničko sučelja u Revitu

### 3.1.2. *Dynamo*

Dynamo je vizualni programski jezik razvijen unutar Autodesk, koji se koristi kao samostalni alat ili se implementira u alate poput Civil 3D, Maya, Revit itd. Dynamo i Revit zajedno se upotrebljavaju za modeliranje i analizu kompleksnih geometrija i automatizaciju repetitivnih procesa. Korisnicima omogućuje rad unutar procesa vizualnog programiranja u kojem se povezuju elementi modela, kombinacijom *nodeova*, kako bi se definirali odnosi i redoslijed radnji koje čine pojednostavljene algoritme. Osim toga, Dynamo dozvoljava izvlačenje geometrije i ostalih tehničkih parametara elemenata iz Revit BIM modela. Veliki broj gotovih paketa, *nodeova*, čine Dynamo jednostavnim i efikasnim alatom za ubrzavanje i usavršavanje BIM radnog procesa [21, 22]. Korisničko sučelje Dynama, prikazano na slici 12, sastoji se od pet glavnih područja, od kojih je najveće radno područje gdje se razvijaju vizualni programi i pregledava razrađena geometrija.

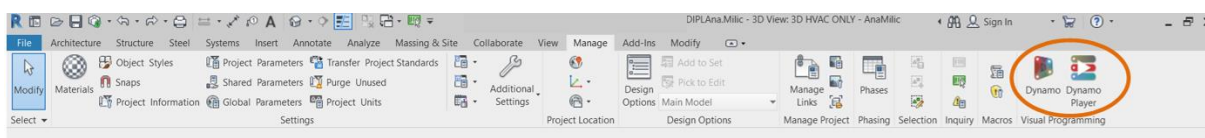
Korisničko sučelje sadrži [21]:

- glavni izbornik
- alatna traka
- knjižnica
- radni prostor
- izvršna traka.



Slika 12. Dynamo sučelje

U sklopu Dynama nalazi se dodatak Dynamo Player, koji se koristi kao produžetak Dynama direktno u Revitu. Algoritam napravljen u Dynamu moguće je pokrenuti preko Dynamo Playera bez otvaranja algoritma u Dynamo sučelju što ubrzava primjenu gotovih algoritama. Osim toga, unutar algoritma u Dynamu omogućuje se odabir potrebnih *nodeova* kao ulaznih i/ili izlaznih podataka koji se u Dynamo Playeru mogu mijenjati sukladno potrebama korisnika.

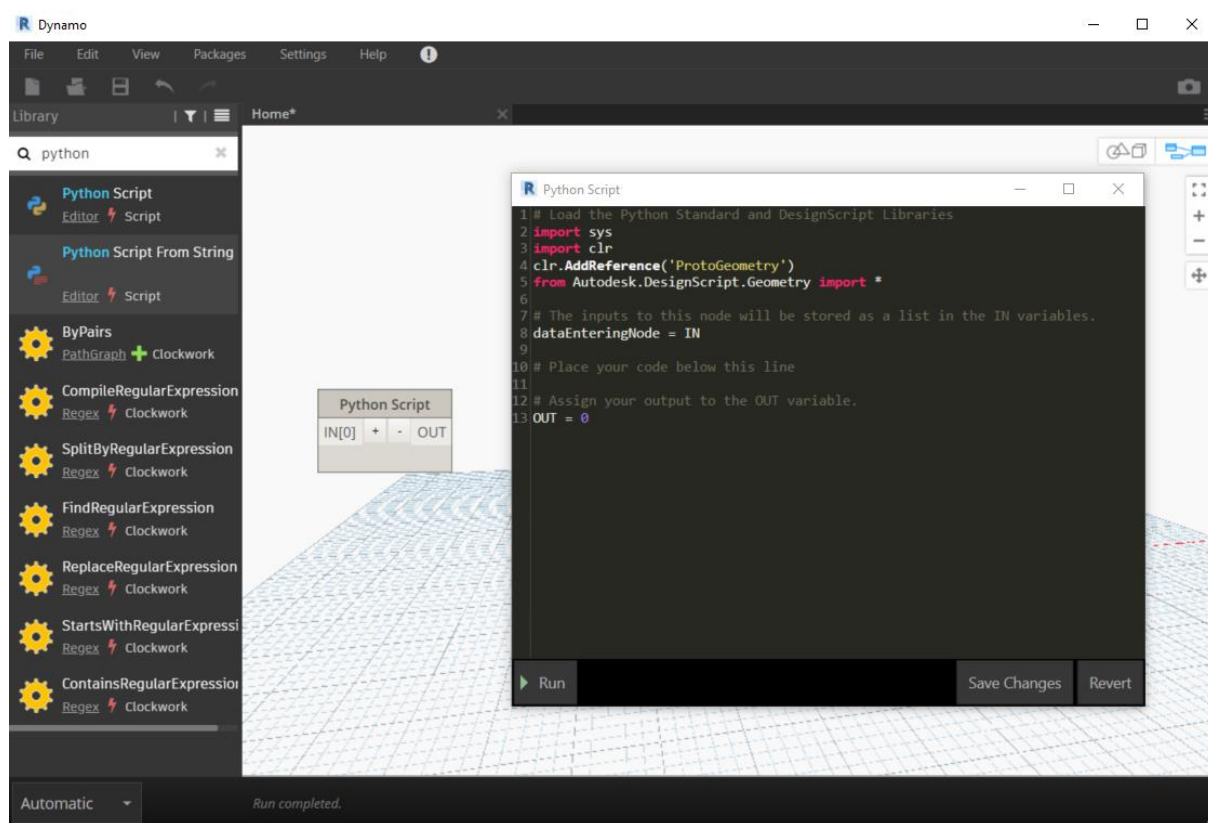


Slika 13. Prikaz Dynama i Dynamo Playera unutar alatne trake Revita

### 3.1.3. Python

Python je programski jezik opće namjene i visoke razine kojeg je stvorio Guido van Rossum 1990. godine, dok je javno objavljen 1991. godine. Python dopušta programerima korištenje stilova kao što su objektno orijentirano, strukturno i aspektno orijentirano programiranje. Zbog svoje fleksibilnosti, raznolikost i široke podrške u mnogim dostupnim bazama funkcija za različite namjene, Python je najkorišteniji programski jezik suvremenog doba [23].

U okviru ovog diplomskog rada u Dynamo sučelju koristit će se Python, putem *nodea Python script*, prikazano na slici 14, kako bi grafovi algoritma bili sljedivi i koncizni.



Slika 14. Python skripta u Dynamu

#### 4. PROGRAMSKO RJEŠENJE

Podatkovni centar je objekt za pohranu podataka koji sadrži IT infrastrukturu. Sastavni dio podatkovnog centra čine sustavi za hlađenje, napajanje, upravljanje i sigurnost za podršku primjene IT tehnologije u kritičnim uvjetima. Konkretno, razradit će se hlađenje jer se oprema za kritično napajanje jako zagrijava, te iziskuje određene ambijentalne uvjete poput temperature, zraka i vlažnosti, a održava se HVAC opremom. HVAC oprema ovog projektnog zadatka proizvedena je prema standardima DIN EN 9001:2000 i ISO/TS 16949:2002, te se uz uputstva koristi za adekvatnu primjenu. Stoga je moguće automatizirati rashladni sustav prema određenim ulaznim podacima u svrhu izbjegavanja repetitivnih procesa i uštedom projektnih resursa.

Automatizacija procesa obuhvaća rashladni sustav podatkovnog centra od unutarnje jedinice do izlaza iz objekta. Razmotreni sustav su plinska i tekuća strana rashladnog kruga, dobava sanitarne vode za napajanje ovlaživača, te odvod kondenzata. Na cijevi plinske i tekuće faze, te dobave sanitarne vode ugrađeni su kuglasti ventili za potrebe zatvaranja sustava tijekom održavanja ili uslijed nesreće. Na grani cijevi plinske faze kao i na cijevi kondenzata, na dijelu gdje vertikala prelazi u horizontalu, izvode se uzvišenja koja stvaraju efekt sifona. Modeliranje rashladnog sustava podatkovnog centra odvija se u strojarskom modelu objekta unutar Revita, verzija 2020. Uporabom Dynama, točnije verzije Dynamo Revit 2.1.0.7733, omogućuje se izvoz potrebnih podataka mreže cjevovodnih instalacija iz Revita, programske okoline za modeliranje predmetnog projektnog rješenja. Štoviše, nakon definiranja i sortiranja potrebnih podataka provođenjem algoritma podaci se uvoze natrag u Revit u obliku parametara i/ili elemenata te se generira 3D BIM model. Stoga se programsko rješenje ovog diplomskog rada sastoji od dva programska algoritma međusobno povezana Excelom. U nastavku su objašnjena oba algoritma unutar kojih se koriste paketi prikazani u tablici 1. Zbog sljedivosti oba algoritma detaljno je prikazana i objašnjena jedna vrsta cjevovodnih instalacija, dok su ostale tri vrste cjevovodnih instalacija prikazane u prilogu ovog diplomskog rada.

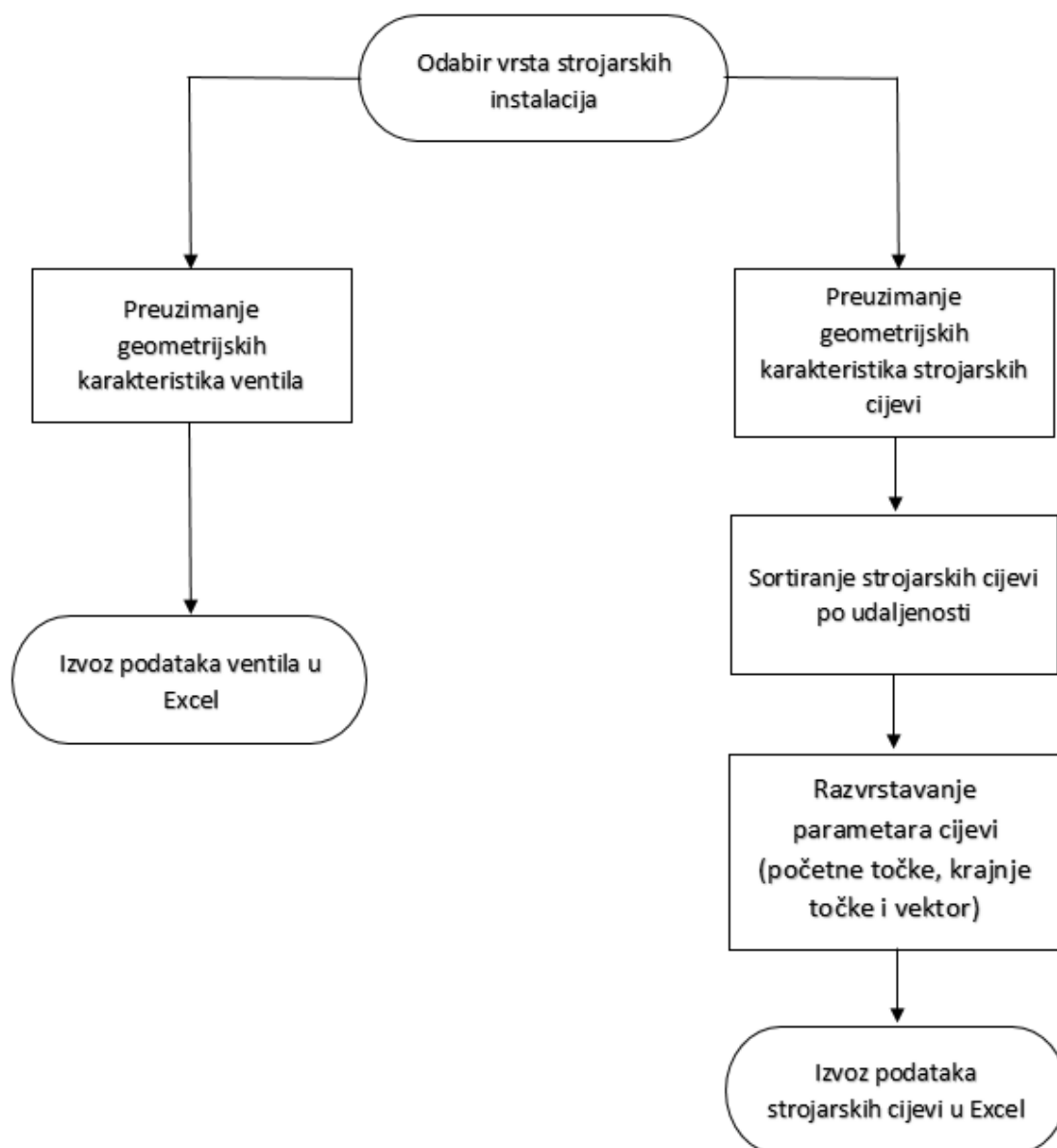


Tablica 1. Korišteni paketi u Dynamu

Naziv paketa	Autor	Opis	Datum objave
Clockwork for Dynamo 2.x	andydandy	Kolekcija od 400+ nodeova za razne svrhe poput upravljanja listama, matematičkim operacijama, geometrijskim operacijama itd.	15. siječnja 2020.
MEPover	tpover	Kolekcija nodeova za manipulaciju strojarskih i elektrotehničkih elemenata	8. lipnja 2020.

#### 4.1. Definiranje i sortiranje podataka modela

Prvi programski algoritam, prikazan u dijagramu toka na slici 15, služi za izvlačenje definiranih i sortiranih ulaznih podataka standardnog modela iz Revita u Excel, pomoću Dynamu. Odabirom vrsta strojarskih odnosno cjevovodnih instalacija rashladnog sustava slijedi preuzimanje njihovih geometrijskih karakteristika koje se nakon sortiranja i razvrstavanja izvoze u Excel tablicu. U nastavku slijedi opis prvog algoritma za slučaj kad je varijabla *pipeSystem* jednaka dobavi sanitarne vode, jednoj o četiri varijable Revit modela, jer je algoritam isti za sve četiri varijable. Otvaranjem standardnog modela rashladnog sustava u Revitu, omogućuje se pokretanje Dynamu.



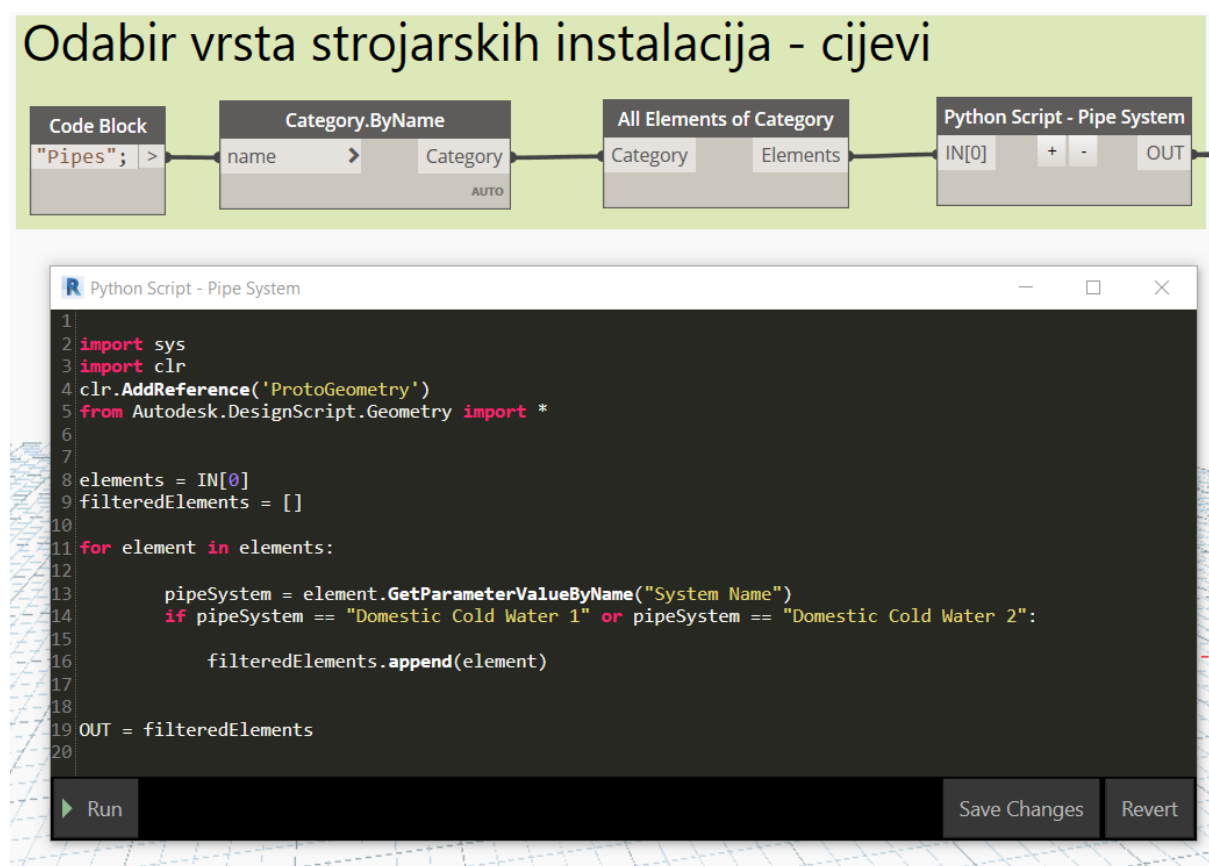
Slika 15. Prikaz algoritma za definiranje ulaznih podataka modela u dijagramu toka

Programski algoritam koji služi za definiranje i sortiranje izlaznih podataka standardnog modela sastoji se od nekoliko povezanih koraka:

1. Odabir vrste strojarskih instalacija – cijevi

Nakon otvaranja Dynamo sučelja potrebno je definirati vrstu strojarskih instalacija zbog jednostavnijeg izvlačenja i definiranja njihovih geometrijskih karakteristika. Uporabom *Pipes* kategorije izdvajaju se samo elementi cijevi u standardnom modelu iz Revita koji se pomoću Python skripte *Pipe System* dodatno definiraju. Radi jednostavnijeg snalaženja u algoritmu

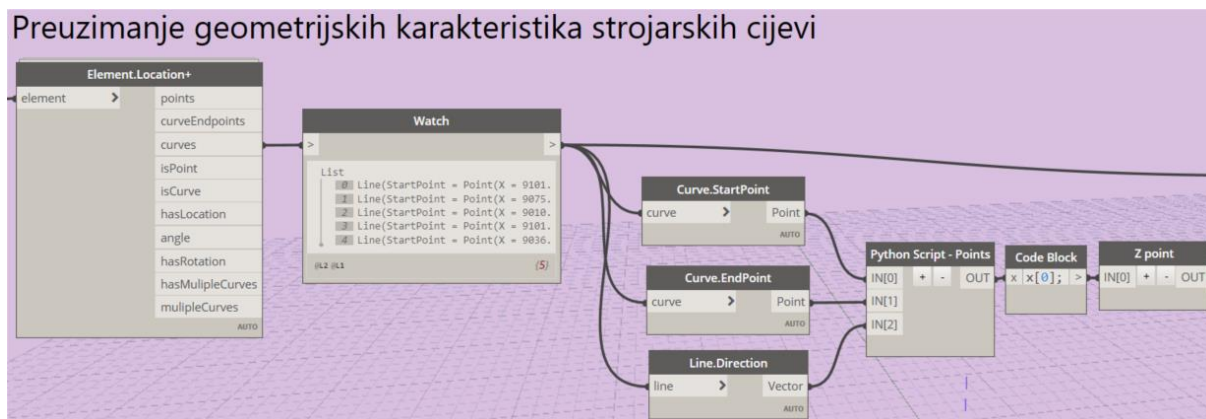
prikazan je slučaj kad je *pipeSystem* varijabla jednaka dobavi sanitarne vode (*Domestic Cold Water*) za napajanje ovlaživača, dok je ostatak varijabli programskog algoritma prikazano u prilogu. Ostatak varijabli programskog algoritma koje se koriste kao *pipeSystem* varijabla su plinska (*Refrigerant Gas*) i tekuća (*Refrigerant Liquid*) strana rashladnog kruga, te odvod kondenzata (*Condensate Drain*). Nakon definiranja cjevovodnih instalacija kod se istovremeno nastavlja na „Preuzimanje geometrijskih karakteristika strojarskih cijevi“ i „Preuzimanje geometrijskih karakteristika ventila“.



Slika 16. Definiranje strojarskih instalacija u Dynamu

## 2. Preuzimanje geometrijskih karakteristika strojarskih cijevi

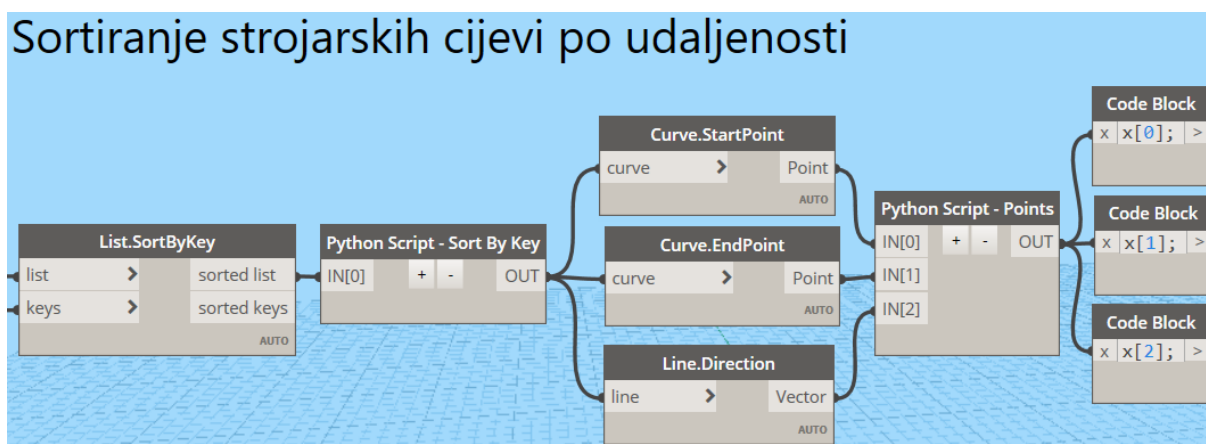
Izdvajanje geometrijskih karakteristika cijevi vrši se temeljem nodea *Element.Location+* koji nakon definiranja elemenata cijevi kreira listu s njihovim geometrijskim karakteristikama preuzetu iz Revit standardnog modela. Nakon toga, izdvajaju se početne i krajnje točke uz vektor smjera odabranih cijevi u zasebnu listu kao što je vidljivo na slici 17.



Slika 17. Algoritam za preuzimanje geometrije cijevi

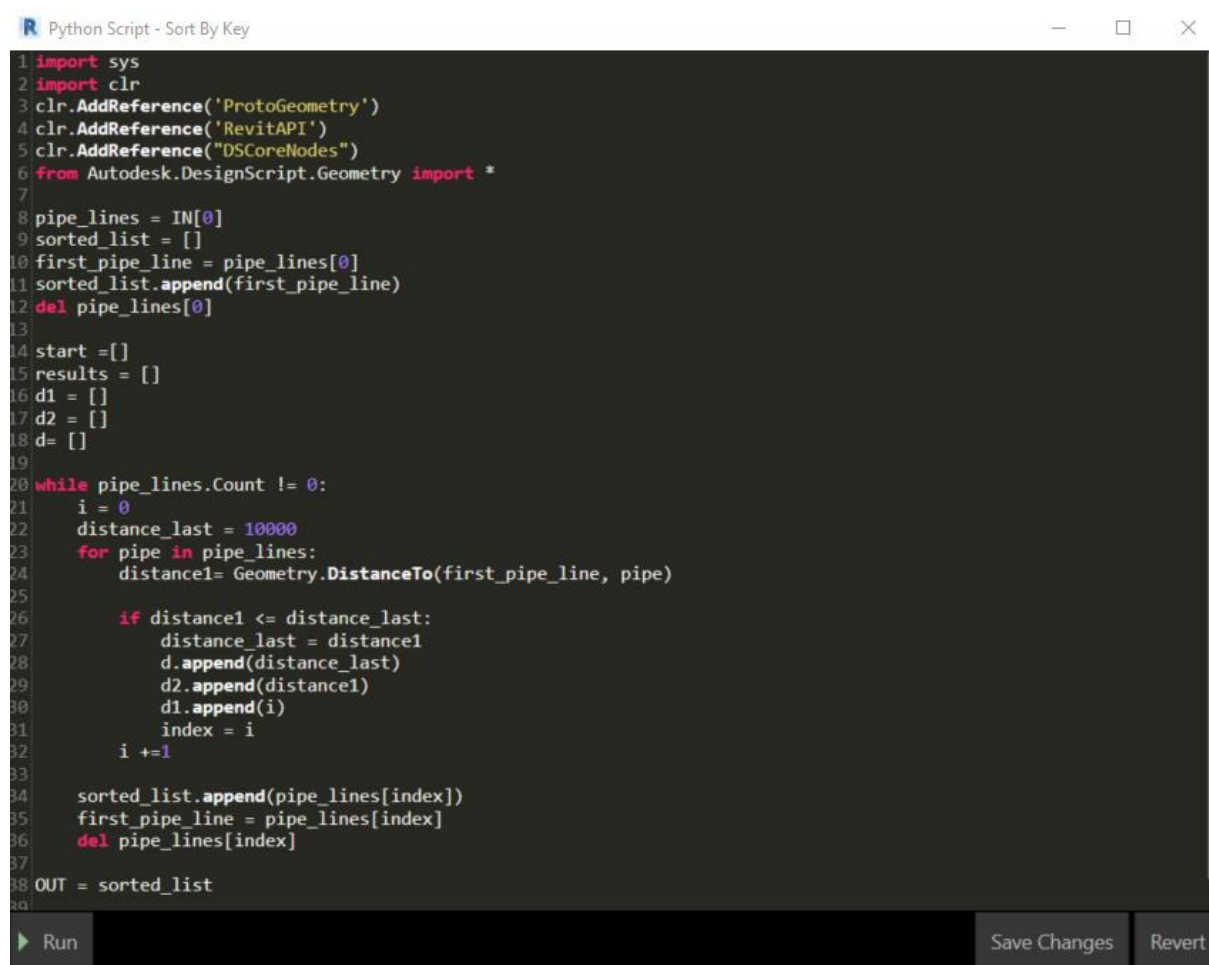
### 3. Sortiranje strojarskih cijevi po udaljenosti

Kako bih pozicije cijevi ostale kao u standardnom modelu, koriste se dvije Python skripte radi pojednostavljenog prikaza grafova algoritma. Za početak se cijevi sortiraju po osi z temeljem nodea *List.SortByKey*. Tako sortirana lista je ulazni podataka za Python skriptu *Sort By Key* u kojoj se filtriraju po najmanjoj udaljenosti, tj. po standardnom redoslijedu cijevi modela. Nakon toga se pomoću Python skripte *Points* razvrstavaju iznosi geometrijskih karakteristika cijevi na tri liste. Prvu listu čine iznosi geometrijskih karakteristika početne točke cijev, drugu listu čine iznosi geometrijskih karakteristika krajnje točke cijevi, dok treću listu čine iznosi geometrijskih karakteristika vektora smjera cijevi.



Slika 18. Algoritam za sortiranje strojarskih cijevi po udaljenosti

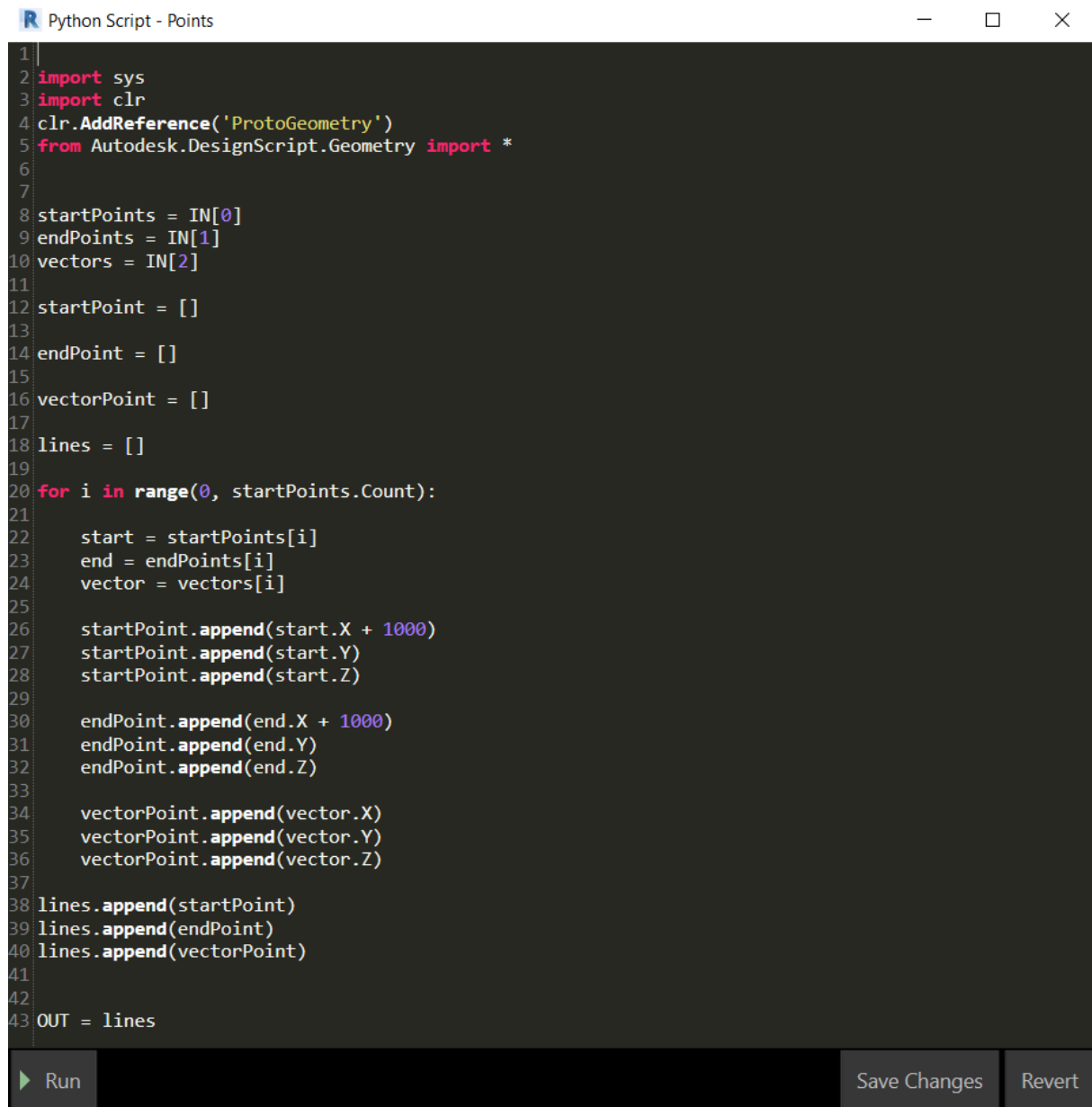
Ulazna varijabla, Python skripte *Sort By Key* koja je prikazana na slici 19, sastoji se od liste poredanih cijevi po osi z. Prva cijev na listi ulazne varijable ujedno je i početna cijev u *while* petlji, kojom se provjera udaljenost s ostalim cijevima poredane liste ulazne varijable. Cijev koja ima najmanju udaljenost s početnom cijevi se dodaje u novu sortiranu listu koja predstavlja izlaznu varijablu Python skripte *Sort By Key*. Osim toga, cijev s najmanjom udaljenosti se briše iz poredane liste, te postaje početna cijev *while* petlje. Postupak *while* petlje se ponavlja sve dok poredana lista nije prazna. Upotrebom *while* petlje kreira se nova lista *sorted\_list* koju čini točan poredak cijevi po standardnom modelu.



```
1 import sys
2 import clr
3 clr.AddReference('ProtoGeometry')
4 clr.AddReference('RevitAPI')
5 clr.AddReference("DSCoreNodes")
6 from Autodesk.DesignScript.Geometry import *
7
8 pipe_lines = IN[0]
9 sorted_list = []
10 first_pipe_line = pipe_lines[0]
11 sorted_list.append(first_pipe_line)
12 del pipe_lines[0]
13
14 start = []
15 results = []
16 d1 = []
17 d2 = []
18 d= []
19
20 while pipe_lines.Count != 0:
21     i = 0
22     distance_last = 10000
23     for pipe in pipe_lines:
24         distance1= Geometry.DistanceTo(first_pipe_line, pipe)
25
26         if distance1 <= distance_last:
27             distance_last = distance1
28             d.append(distance_last)
29             d2.append(distance1)
30             d1.append(i)
31             index = i
32         i +=1
33
34     sorted_list.append(pipe_lines[index])
35     first_pipe_line = pipe_lines[index]
36     del pipe_lines[index]
37
38 OUT = sorted_list
```

Slika 19. Sortiranje cijevi po udaljenosti Python skriptom

Python skripta *Points* sastoji se od tri ulazne varijable cijevi: početne točke, krajnje točke i vektor smjera. Kako bi se razvrstali iznosi geometrijskih karakteristika cijevi na tri liste koristi se *for* petlja, prikazana na slici 20.

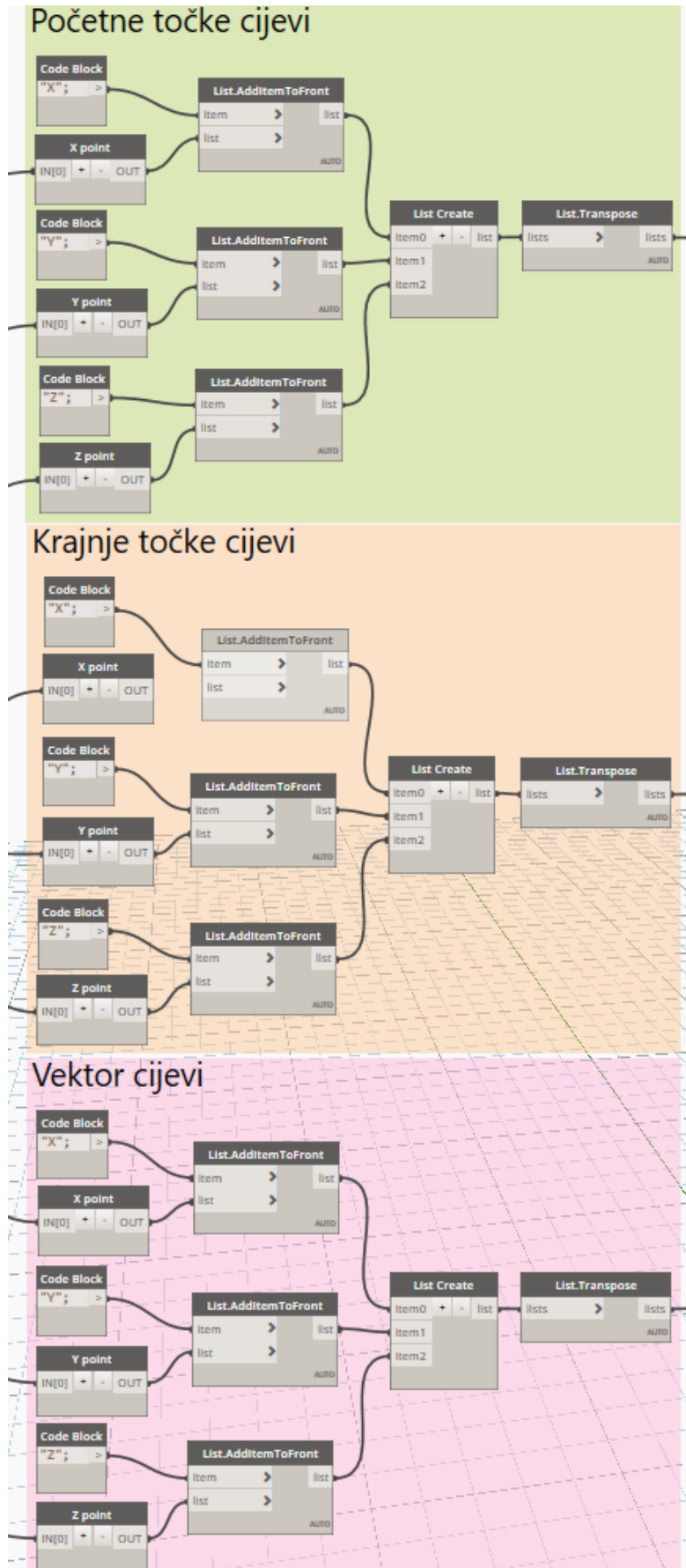


```
1
2 import sys
3 import clr
4 clr.AddReference('ProtoGeometry')
5 from Autodesk.DesignScript.Geometry import *
6
7
8 startPoints = IN[0]
9 endPoints = IN[1]
10 vectors = IN[2]
11
12 startPoint = []
13
14 endPoint = []
15
16 vectorPoint = []
17
18 lines = []
19
20 for i in range(0, startPoints.Count):
21
22     start = startPoints[i]
23     end = endPoints[i]
24     vector = vectors[i]
25
26     startPoint.append(start.X + 1000)
27     startPoint.append(start.Y)
28     startPoint.append(start.Z)
29
30     endPoint.append(end.X + 1000)
31     endPoint.append(end.Y)
32     endPoint.append(end.Z)
33
34     vectorPoint.append(vector.X)
35     vectorPoint.append(vector.Y)
36     vectorPoint.append(vector.Z)
37
38 lines.append(startPoint)
39 lines.append(endPoint)
40 lines.append(vectorPoint)
41
42
43 OUT = lines
```

Slika 20. Python skripta za razvrstavanje iznosa cijevi

#### 4. Razvrstavanje parametara cijevi (početne točke, krajnje točke i vektor cijevi)

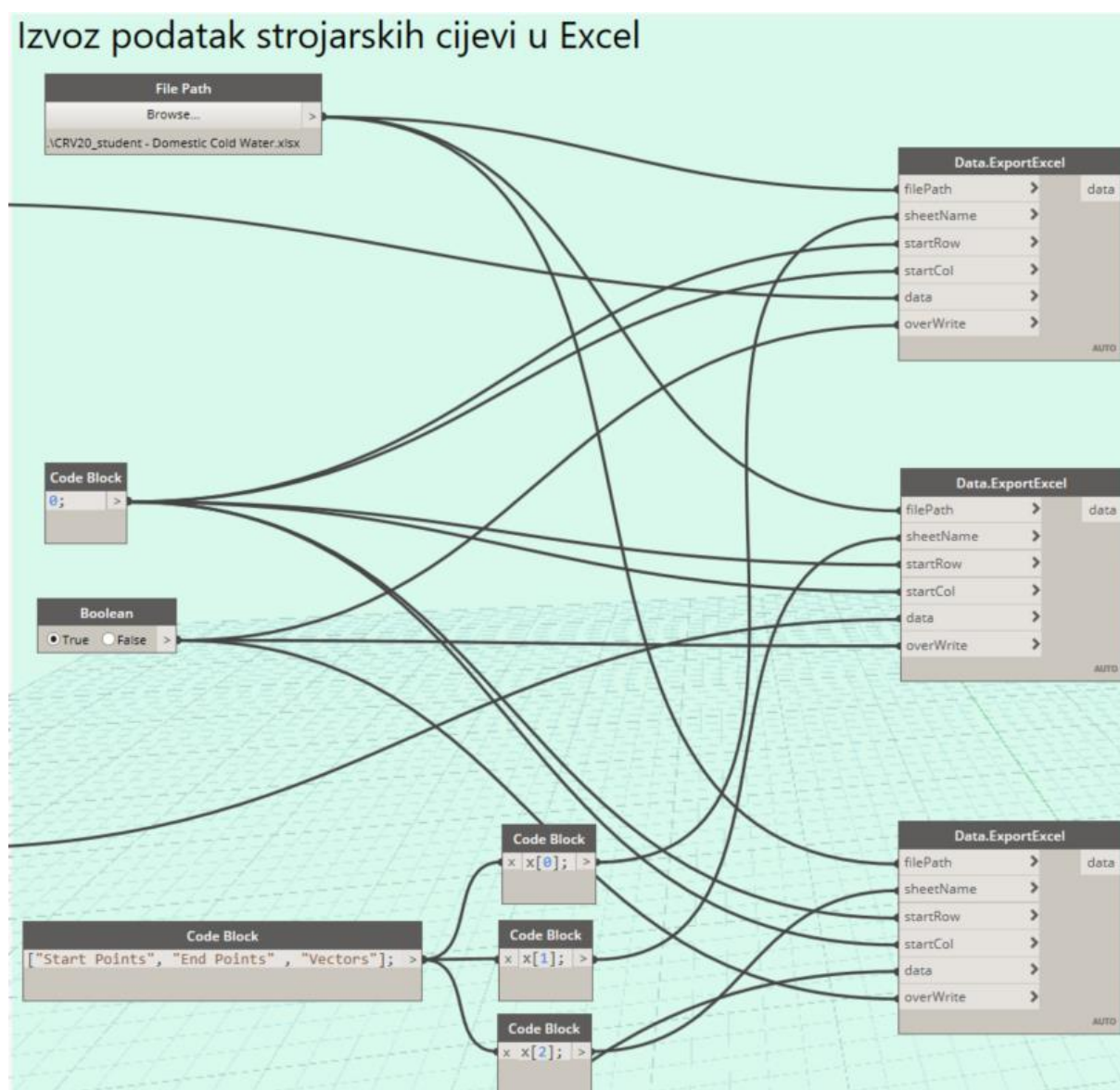
Korak 4 odvija se nakon sortiranja izlaznih podataka modela. Lista poredanih cijevi se dijeli na tri liste: početne točke cijevi, krajnje točke cijevi i vektor cijevi, zbog jednostavnijeg izvoza podataka iz Revita u Excel. Sve tri liste se sastoje od vlastitih iznosa pozicija točki cijevi u osima  $x$ ,  $y$  i  $z$ .



Slika 21. Prikaz algoritma izlaznih podataka modela

## 5. Izvoz podataka strojarskih cijevi u Excel

Izvoz definiranih i sortiranih podataka modela iz Revita u Excel se vrši *nodeom Data.ExportExcel*, kako je prikazano na slici 22. Ulazni podaci *nodea Data.ExportExcel* su: *filePath*, *sheetName*, *startRow*, *startCol*, *data* i *overWrite*. Lokacija Excel tablice definira se *nodeom File Path*, dok se ulaznim podatkom *overWrite* koristi Booleova logika za kontrolu podataka. Izgled tablice definira se ostalim ulaznim podacima *nodea Data.ExportExcel*. Pomoću *Code Blocka* definirani su nazivi listova (*sheetName*) tablice: *Start Points*, *End Points* i *Vectors*. Nakon toga, početak retka i stupca u tablici označuju ulazni podaci *startRow* i *startCol*, dok ulazna varijabla *data* sadrži podatke koji se izvoze u Excel.

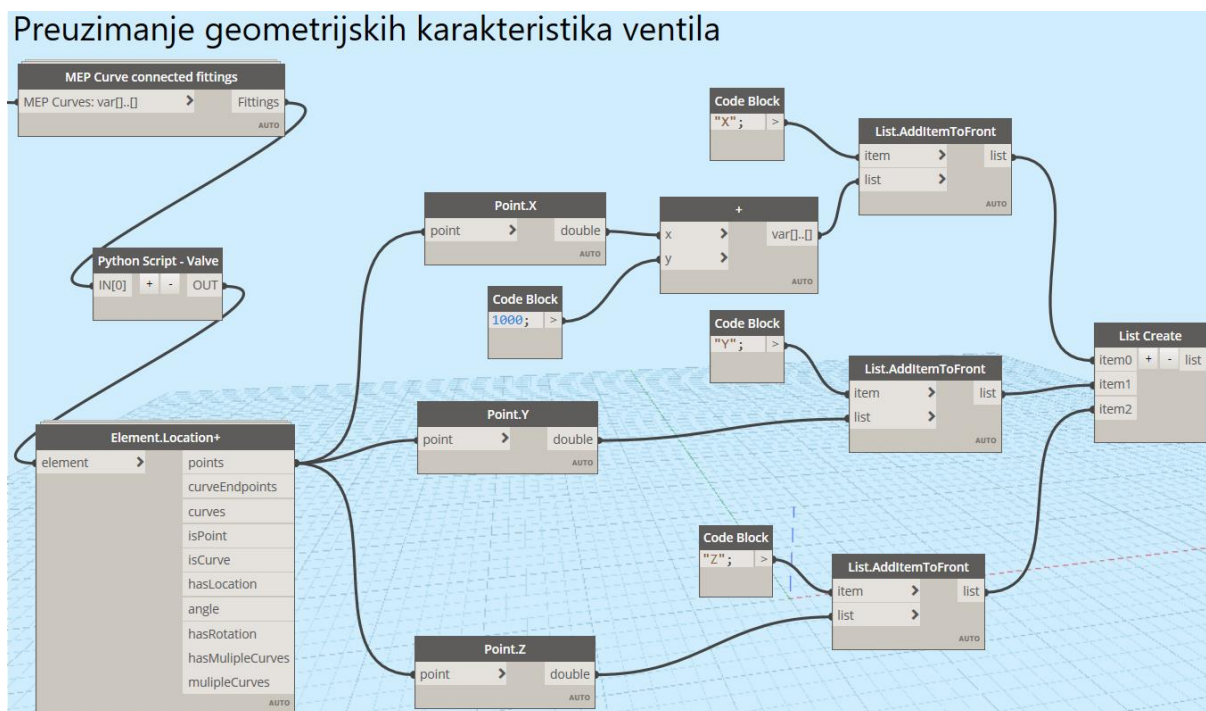


Slika 22. Algoritam izvoza podataka modela u Excel



## 6. Preuzimanje geometrijskih karakteristika ventila

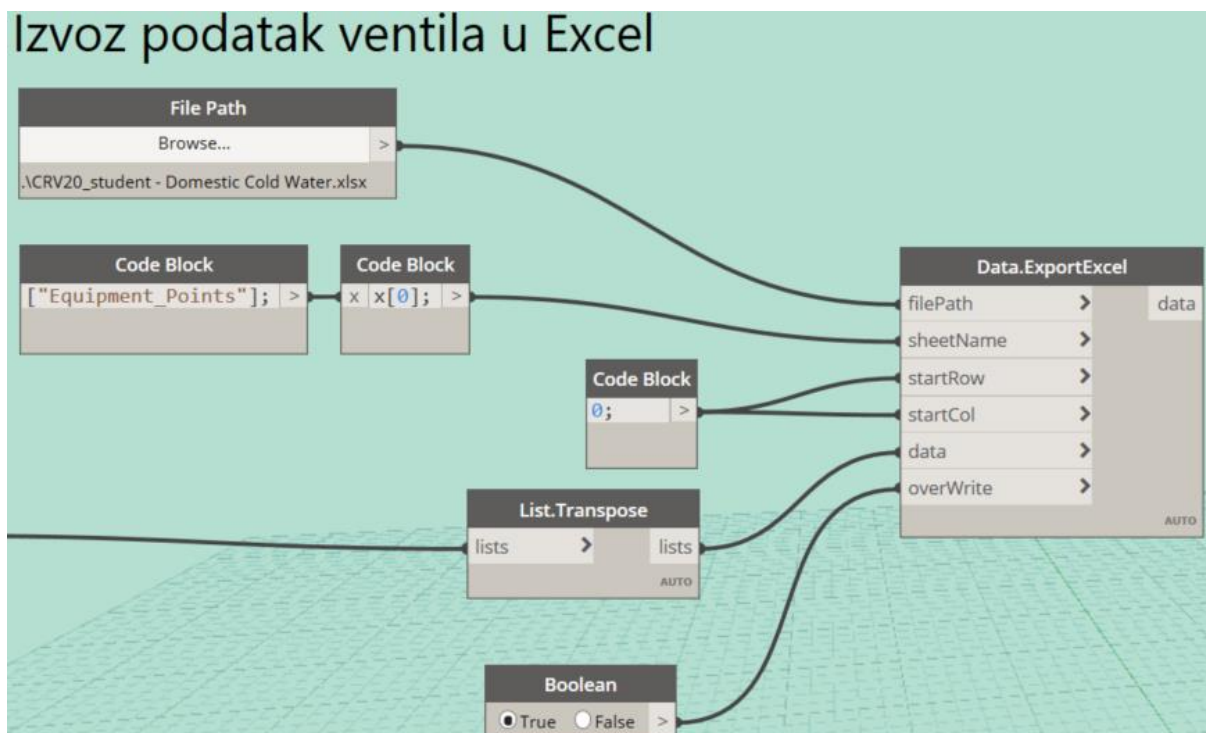
Definiranjem cjevovodnih instalacija paralelno se uz preuzimanje geometrijskih pozicija strojarskih cijevi preuzimaju geometrijske karakteristike ventila iz Revita. Geometrija ventila preuzima se pomoću *nodea MEP Curve connected fittings* i *Element.Location+*. Potom se kreira lista s geometrijom ventila, prikazano na slici 23.



Slika 23. Algoritam preuzimanja geometrije ventila

## 7. Izvoz podataka ventila u Excel

Geometrijski podaci ventila izvoze se pomoću *nodea Data.ExportExcel* iz Revita u Excel istim postupkom kao pod korakom 5, kako je prikazano na slici 24. U ovom slučaju su ulazni podaci geometrijske karakteristike ventila, dok su u koraku 5 ulazni podaci jednaki geometrijskim karakteristikama cjevovodnih instalacija, što predstavlja razliku između koraka 5 i 7.



Slika 24. Algoritam izvoza podataka ventila u Excel

Izlazni podatci: početne točke, krajnje točke, vektori i ventili za sve četiri vrste strojarskih cijevi izvoze se iz Revit standardnog modela u Excel tablicu. Prikaz Excel tablica izlaznih podataka, geometrijskih karakteristika strojarskih instalacija, vidljivi su na sljedećim slikama.

Refrigerant Gas							Condensate Drain						
A	B	C	D	E	F	G	A	B	C	D	E	F	G
X	Y	Z					X	Y	Z				
10541.06	-2091.48	2009.072					10107.06	-2159.75	2029.9				
10541.06	-2122.06	2101.6					10076.48	-2159.75	2720				
10541.05	-2231.83	2132.175					10017.06	-2159.75	2689.425				
10541.19	-2231.83	2689.425					9986.485	-2159.75	2594.154				
10510.63	-2231.82	2720											
10451.21	-2231.81	2689.425											
10420.63	-2231.81	2590											

Refrigerant Liquid							Domestic Cold Water						
A	B	C	D	E	F	G	A	B	C	D	E	F	G
X	Y	Z					X	Y	Z				
10541.06	-2159.75	2035.788					10101.06	-2091.48	2281.028				
10541.06	-2185.56	2061.6					10101.06	-2091.48	2524.188				
10541.06	-2281.81	2087.412					10075.25	-2091.48	2550				
10541.06	-2281.81	2564.188					10036.06	-2117.3	2550				
10515.25	-2281.81	2590					10010.25	-2191.81	2550				

Slika 25. Izlazni podaci početnih točki cijevi standardnog modela u Excelu

Refrigerant Gas							Condensate Drain						
A	B	C	D	E	F	G	A	B	C	D	E	F	G
X	Y	Z					X	Y	Z				
10541.06	-2091.48	2071.025					10107.06	-2159.75	2689.425				
10541.05	-2201.26	2101.6					10047.63	-2159.75	2720				
10541.05	-2231.83	2218.288					10017.06	-2159.75	2625.035				
10541.05	-2231.83	2403.288					8821.058	-2159.75	2582.5				
10481.78	-2231.81	2720											
10451.21	-2231.81	2620.575											
8821.058	-2231.81	2590											

Refrigerant Liquid							Domestic Cold Water						
A	B	C	D	E	F	G	A	B	C	D	E	F	G
X	Y	Z					X	Y	Z				
10541.06	-2159.75	1985.4					10101.06	-2091.48	2038.95				
10541.06	-2256	2061.6					10101.06	-2091.48	2386.676				
10541.06	-2281.81	2214.2					10061.87	-2091.48	2550				
10541.06	-2281.81	2399.2					10036.06	-2166	2550				
8806.206	-2281.81	2590					8821.058	-2191.81	2550				

Slika 26. Izlazni podaci krajnjih točki cijevi standardnog modela u Excelu

Refrigerant Gas							Condensate Drain									
	A	B	C	D	E	F	G		A	B	C	D	E	F	G	
1	X	Y	Z					1	X	Y	Z					
2		0	0	61.95299				2		0	7.11E-13	659.525				
3	-0.00649	-79.1995	-4.11E-05					3		-28.85	-2.39E-07	3.55E-13				
4	0.001919	0.001066	86.11324					4		0	3.61E-08	-64.39				
5	-0.13746	-0.00255	-286.137					5		-1165.43	-3.31E-06	-11.6543				
6	-28.8477	0.007056	1.56E-06					6								
7	-7.84E-07	-0.00288	-68.85					7								
8	-1599.57	0.007011	1.78E-12					8								
9								9								
10								10								
11								11								
12								12								

Refrigerant Liquid							Domestic Cold Water									
	A	B	C	D	E	F	G		A	B	C	D	E	F	G	
1	X	Y	Z					1	X	Y	Z					
2		0	0	-50.3875				2		0	-1.39E-11	-242.078				
3	7.11E-12	-70.4416	-7.11E-13					3		0	1.63E-11	-137.512				
4	-7.98E-05	1.06E-05	126.7875					4		-13.375	0	0				
5	-8.01E-06	-2.58E-05	-164.987					5		0	-48.7041	0				
6	-1709.04	-0.00168	7.11E-13					6		-1189.19	1.42E-12	0				
7								7								
8								8								
9								9								
10								10								
11								11								
12								12								

Slika 27. Izlazni podaci vektora cijevi standardnog modela u Excelu

	A	B	C	D	E	F
1	X	Y	Z			
2	10540.92	-2231.83	2314.088			
3						

	A	B	C	D	E	F
1	X	Y	Z			
2	10540.92	-2281.81	2310			
3						

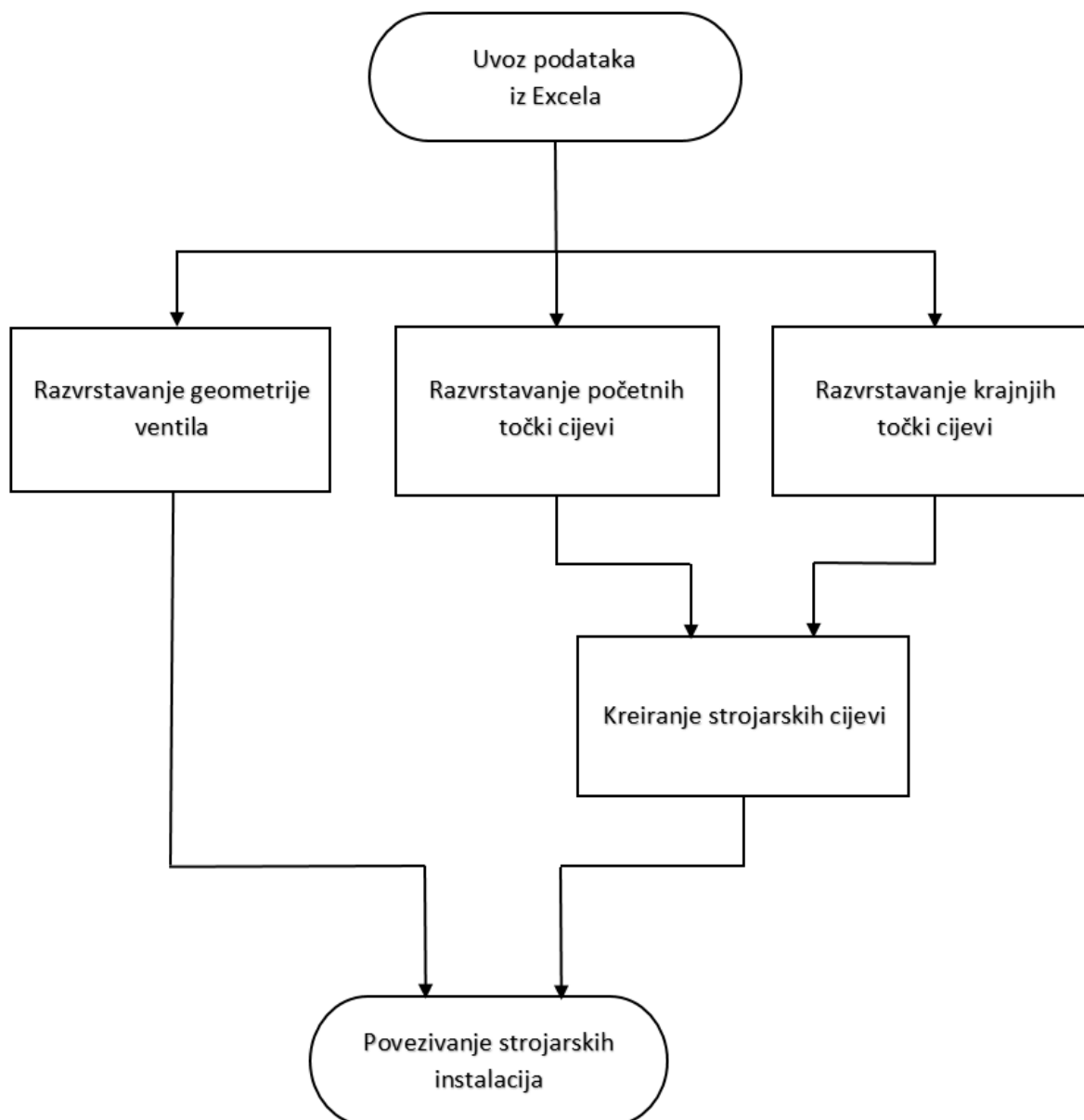
  

	A	B	C	D	E	F
1	X	Y	Z			
2	10101.06	-2091.48	2318.528			
3						

Slika 28. Izlazni podaci ventila standardnog modela u Excelu

#### 4.2. Obrada i izrada modela

Drugi programski algoritam, prikazan u dijagramu toka na slici 29, služi za obradu i izradu standardnog modela u novom Revit modelu. Nakon izvlačenja potrebnih podataka standardnog modela prethodnim algoritmom, podaci se provođenjem algoritma za obradu i izradu modela uvoze iz Excel tablice natrag u Revit u obliku parametara i elemenata te se generira novi 3D BIM model.



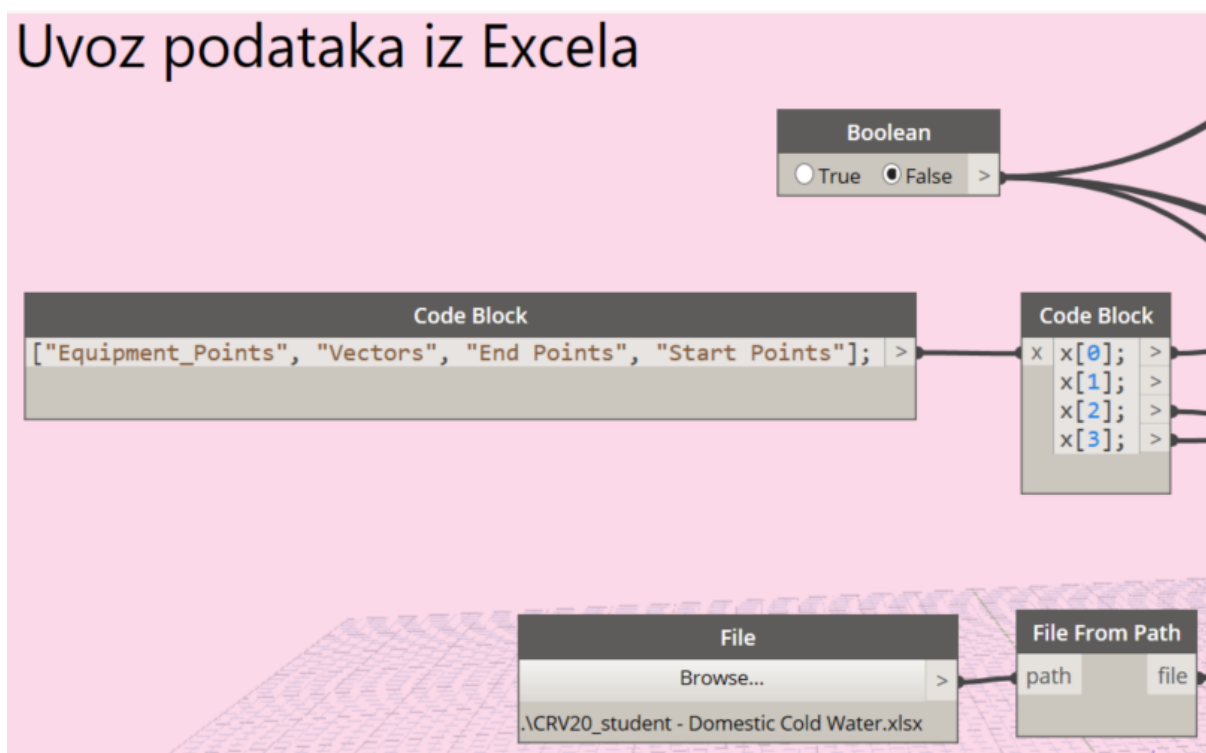
**Slika 29. Prikaz algoritma za obradu i izradu modela u dijagramu toka**

Radi jednostavnijeg snalaženja u programskom algoritmu za obradu i izradu novog modela, kao i u prethodnom algoritmu, prikazan je slučaj za jednu vrstu cjevovodnih instalacija, dobava sanitarne vode (*Domestic Cold Water*) za napajanje ovlaživača. Preostale tri vrste cjevovodnih instalacija rashladnog sustava: plinska (*Refrigerant Gas*) i tekuća (*Refrigerant Liquid*) strana rashladnog kruga, te odvod kondenzata (*Condensate Drain*) prikazane su u prilogu ovog diplomskog rada.

Programski algoritam za obradu i izradu novog modela u Revitu sastoji se nekoliko koraka koji se nadovezuju na prethodni algoritam:

### 1. Uvoz podataka iz Excela

Podaci strojarskih instalacija se uvoze iz Excel tablice koju smo koristili u prethodnom algoritmu za izvoz podataka postojećeg standardnog modela iz Revita. *Nodeom File Path* odabire se lokacija Excel datoteke iz kojeg se podaci unose u Dynamo. Na slici 30 prikazan je uvoz podataka dobave sanitarne vode (*Domestic Cold Water*) iz Excela.

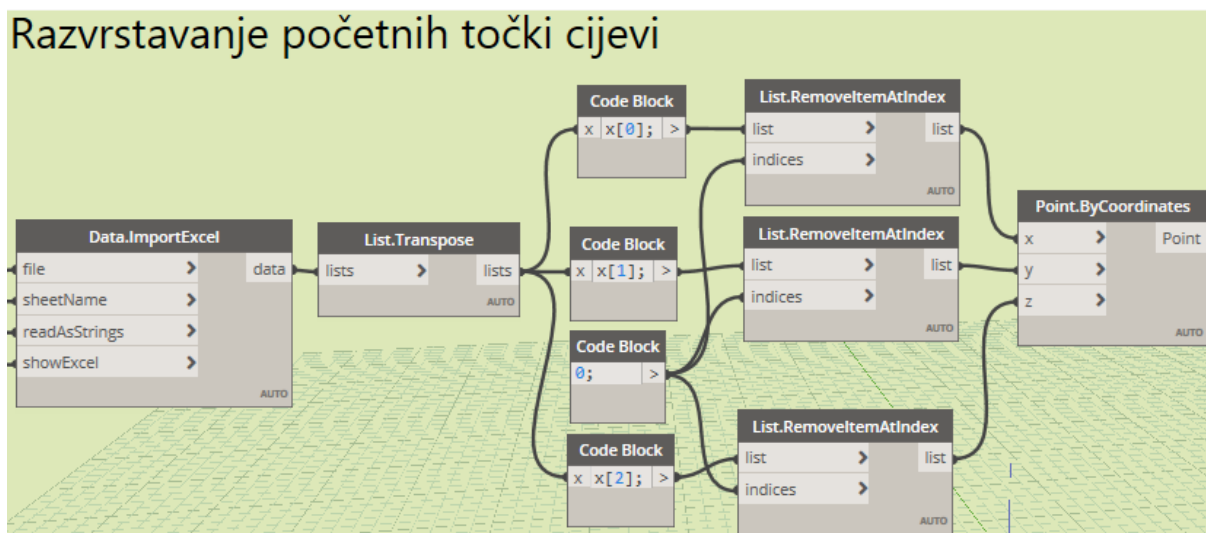


Slika 30. Prikaz algoritma za uvoz podatka iz Excela

### 2. Razvrstavanje početnih točki cijevi

Podaci početnih točki strojarskih cijevi koji se uvoze iz Excel tablice razvrstavaju se u zasebnu listu, jer se koriste kao jedna od dvije ulazne varijable koje skupa čine liniju potrebnu za kreiranje putanje cijevi u koraku 5. Uvoz podataka se vrši *nodeom Data.ImportExcel* pomoću definiranja ulaznih podataka: *filePath*, *sheetName*, *readAsString* i *showExcel*. Lokacija Excel tablice definira se *nodeom File Path*, dok se iznosi koji se žele uvesti definiraju odabirom naziva lista Excel tablice pod *sheetName*. U ovom slučaju uvoze se podaci početnih točki strojarskih cijevi odabirom *Start Points* pod *sheetName nodea*

*Data.ImportExcel*. Ulazni podaci *readAsString* i *showExcel* koriste Booleovu logiku za kontrolu željenih podataka. Nakon toga, slijedi razvrstavanje podataka početnih točki u zasebne liste kako bi se kreirala putanja cijevi u koraku 5.



Slika 31. Algoritam za razvrstavanje početnih točki cijevi

### 3. Razvrstavanje krajnjih točki cijevi

Krajnje točke cijevi razvrstavaju se paralelno s korakom 2, jer se koriste kao druga ulazna varijabla koja skupa sa početnim točkama čine liniju potrebnu za kreiranje putanje cijevi. Stoga se u ovom slučaju uvoze podaci krajnjih točki strojarških cijevi odabirom *End Points* pod *sheetName* nodea *Data.ImportExcel*. Zatim slijedi razvrstavanje podataka krajnjih točki u zasebne liste kao i u prethodnom koraku.

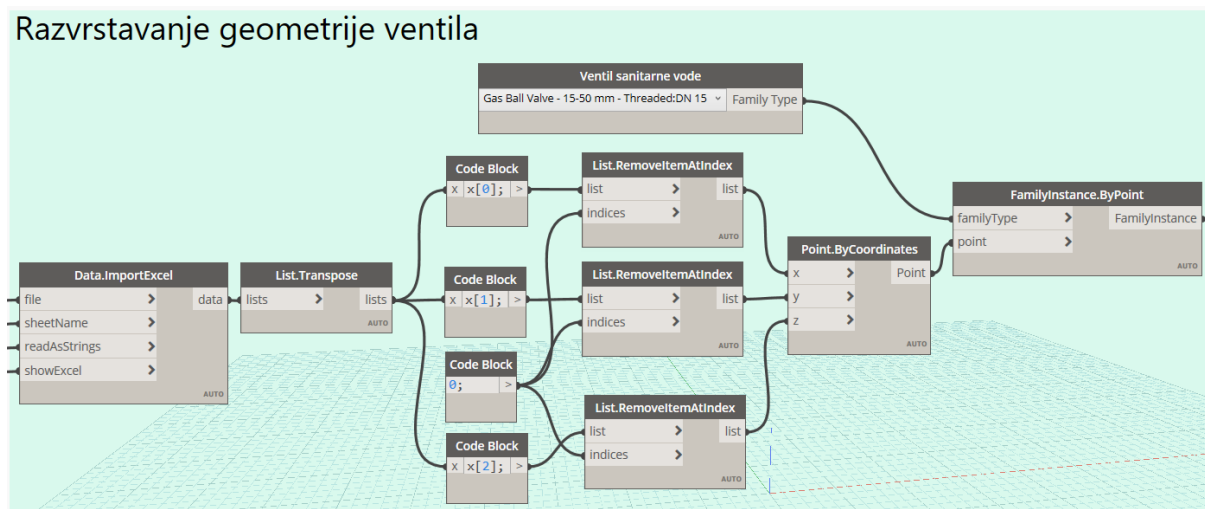




Slika 32. Algoritam za razvrstavanje krajnjih točki cijevi

#### 4. Razvrstavanje geometrije ventila

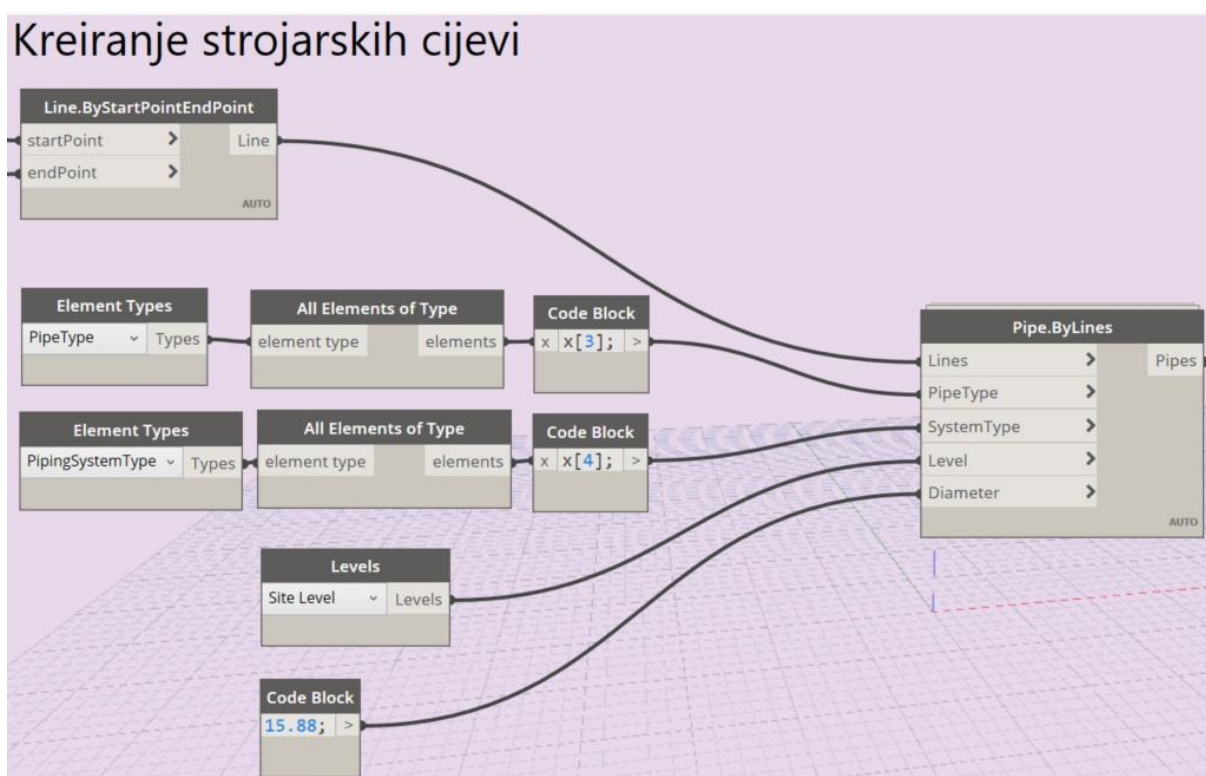
Geometrija ventila razvrstava se istodobno s koracima 2 i 3. Podaci geometrijskih karakteristika ventila se uvoze odabirom *Equipment\_Points* pod *sheetName* nodea *Data.ImportExcel*. Zatim slijedi razvrstavanje podataka geometrije ventila u zasebne liste kao i u prethodna dva koraka. Osim toga, odabire se vrsta ventila u padajućem *nodeu FamilyType*. U ovom slučaju prikazan je odabrani kuglasti ventil (*Gas Ball Valve*) za dobavu sanitarne vode.



Slika 33. Algoritam za razvrstavanje geometrijskih karakteristika ventila

## 5. Kreiranje strojarskih cijevi

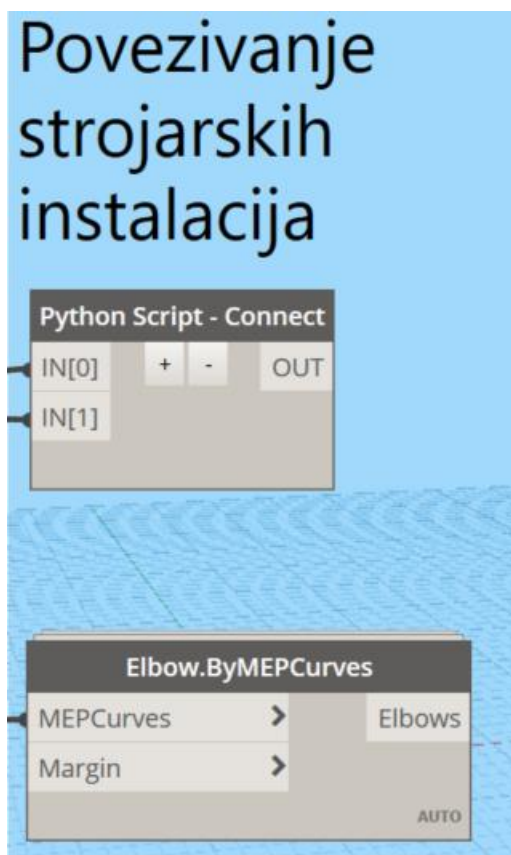
Prva ulazne varijabla, za kreiranje putanje strojarskih cijevi, unutar *nodea Pipe.ByLines* se sastoji od linija cijevi. Liniju cijevi čine razvrstane početne i krajnje točke iz koraka 2 i 3. Osim toga, kako bi se generirale strojarske cijevi u novom modelu unutar Revita potrebno je definirati i druge ulazne varijable unutar *nodea Pipe.ByLines*, a to su: *PipeType*, *SystemType*, *Level* i *Diameter*, kako je prikazano na slici 34.



Slika 34. Algoritam za kreiranje strojarskih cijevi

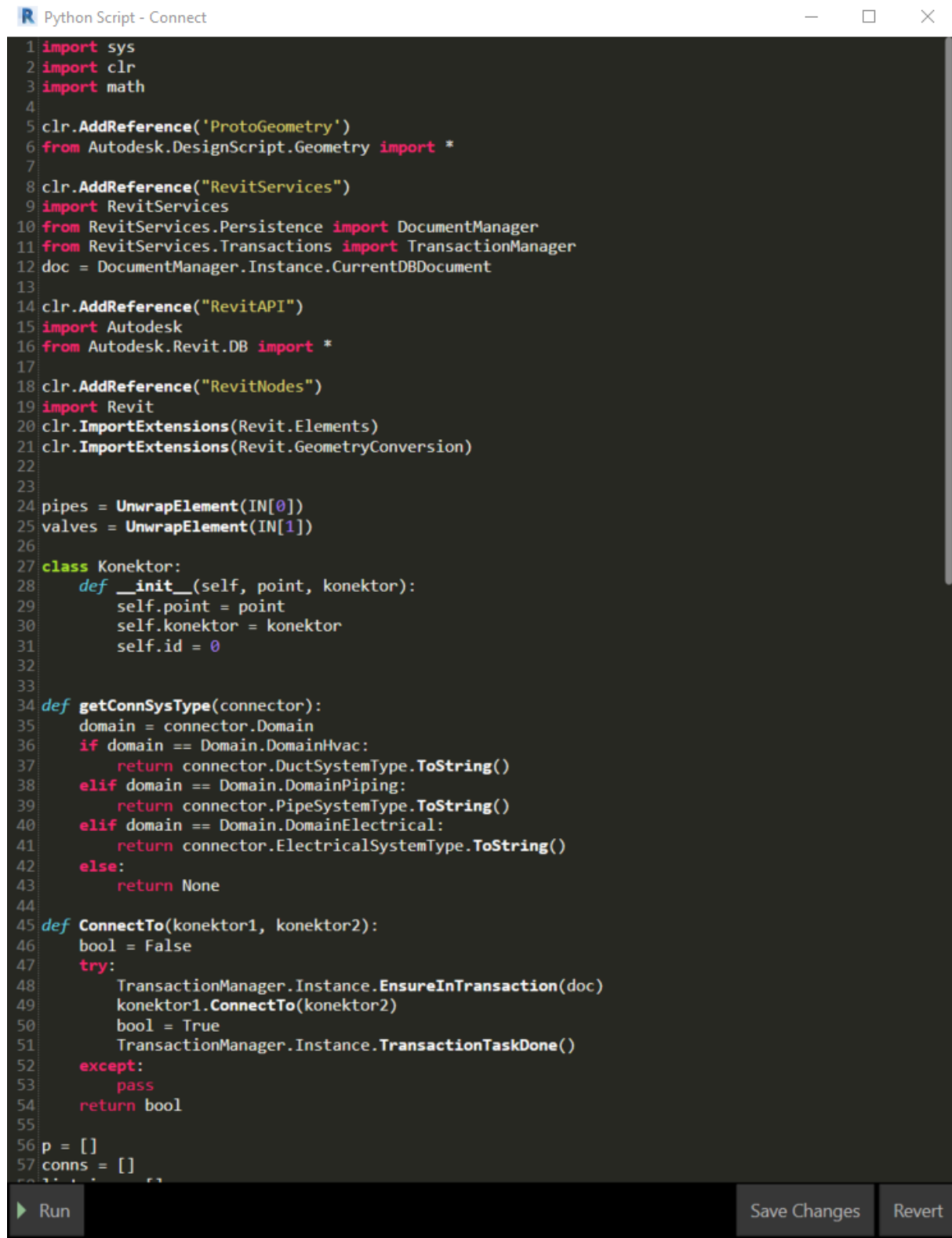
## 6. Povezivanje strojarskih instalacija

Nakon kreiranja geometrijskih pozicija strojarskih instalacija u modelu unutar Revita, strojarske cijevi i ventili povezuju se pomoću Python skripte *Connect*, dok se strojarske cijevi spajaju s koljenima pomoću *nodea Elbow.ByMEPCurves*, kako je prikazano na slici 35.



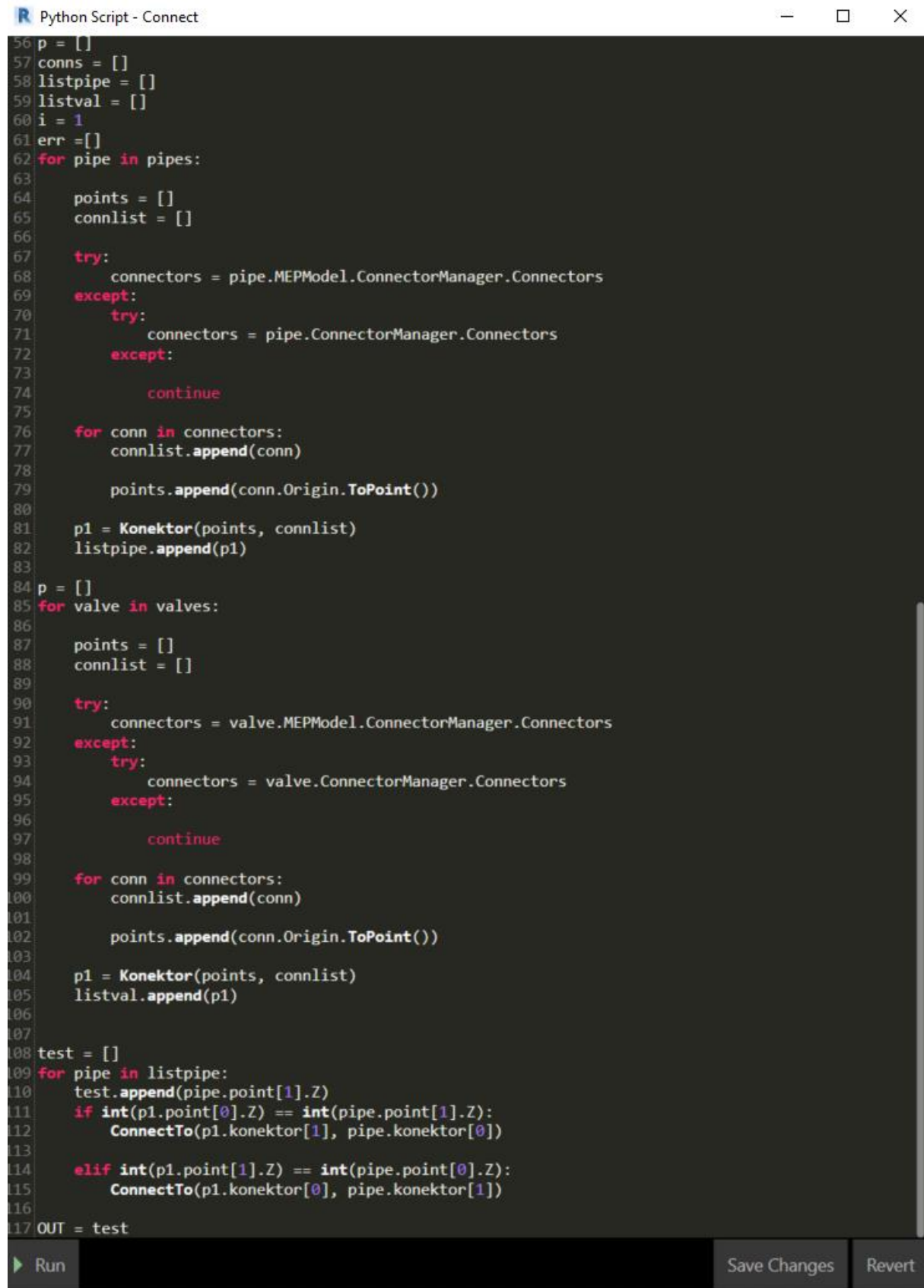
Slika 35. Algoritam za povezivanje strojarskih instalacija

U Python skripti *Connect* prvu ulaznu varijablu čine strojarske cijevi, dok drugu ulaznu varijablu čini ventil odabrane vrste cijevi. U nastavku skripte definira se klasa konektor radi spajanja više podataka. Slijedi definiranje funkcija konektora: tip sistema konektora *getConnSysType* i spajanje konektora *ConnectTo*. Nakon toga se pomoću *for* petlje kreiraju dvije liste. Prva lista *listpipe* sadrži geometrijske karakteristike konektora cijevi, dok druga lista *listval* sadrži geometrijske karakteristike konektora ventila. Zatim se korištenjem *if* uvjeta u novoj *for* petlji spajaju strojarske cijevi s ventilima pomoću definiranih konektora kako bi se formirala putanja mreže cjevovodnih instalacija. Python skripta *Connect* prikazana je na slikama 36 i 37.



```
1 import sys
2 import clr
3 import math
4
5 clr.AddReference('ProtoGeometry')
6 from Autodesk.DesignScript.Geometry import *
7
8 clr.AddReference("RevitServices")
9 import RevitServices
10 from RevitServices.Persistence import DocumentManager
11 from RevitServices.Transactions import TransactionManager
12 doc = DocumentManager.Instance.CurrentDBDocument
13
14 clr.AddReference("RevitAPI")
15 import Autodesk
16 from Autodesk.Revit.DB import *
17
18 clr.AddReference("RevitNodes")
19 import Revit
20 clr.ImportExtensions(Revit.Elements)
21 clr.ImportExtensions(Revit.GeometryConversion)
22
23
24 pipes = UnwrapElement(IN[0])
25 valves = UnwrapElement(IN[1])
26
27 class Konektor:
28     def __init__(self, point, konektor):
29         self.point = point
30         self.konektor = konektor
31         self.id = 0
32
33
34 def getConnSysType(connector):
35     domain = connector.Domain
36     if domain == Domain.DomainHvac:
37         return connector.DuctSystemType.ToString()
38     elif domain == Domain.DomainPiping:
39         return connector.PipeSystemType.ToString()
40     elif domain == Domain.DomainElectrical:
41         return connector.ElectricalSystemType.ToString()
42     else:
43         return None
44
45 def ConnectTo(konektor1, konektor2):
46     bool = False
47     try:
48         TransactionManager.Instance.EnsureInTransaction(doc)
49         konektor1.ConnectTo(konektor2)
50         bool = True
51         TransactionManager.Instance.TransactionTaskDone()
52     except:
53         pass
54     return bool
55
56 p = []
57 conns = []
```

Slika 36. Prvi dio Python skripte za spajanje strojarskih instalacija



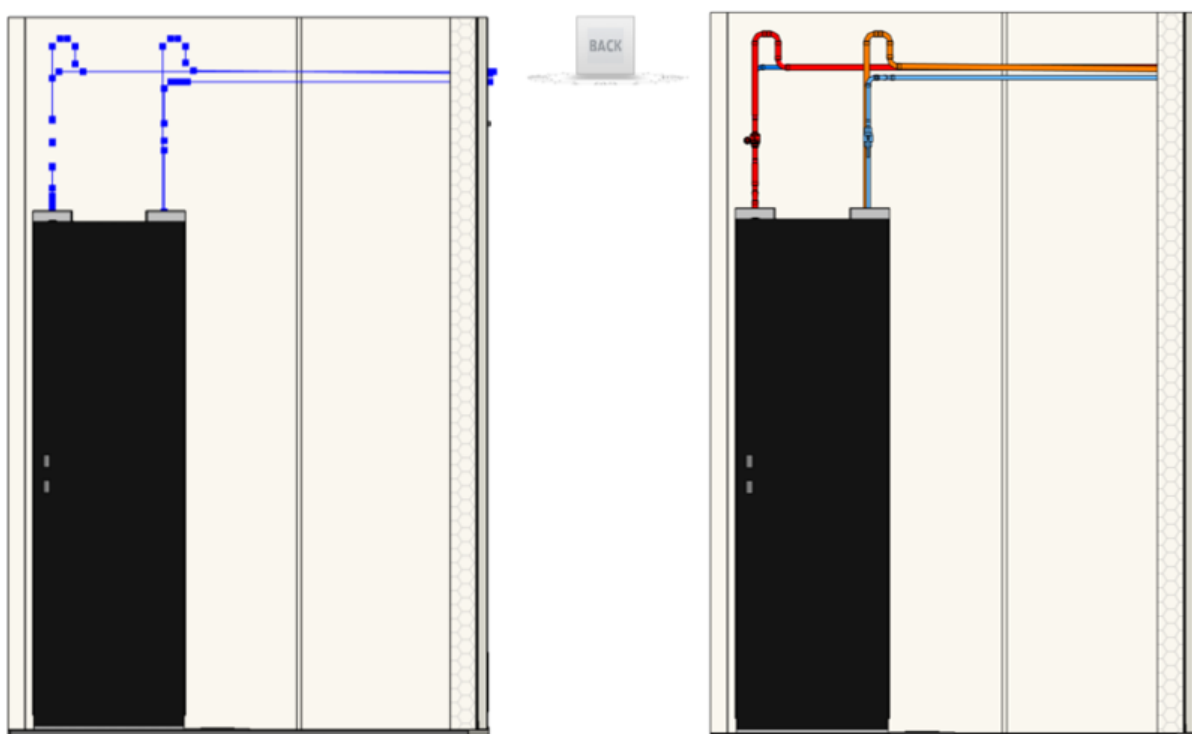
```
Python Script - Connect
56 p = []
57 conns = []
58 listpipe = []
59 listval = []
60 i = 1
61 err = []
62 for pipe in pipes:
63
64     points = []
65     connlist = []
66
67     try:
68         connectors = pipe.MEPModel.ConnectorManager.Connectors
69     except:
70         try:
71             connectors = pipe.ConnectorManager.Connectors
72         except:
73
74             continue
75
76     for conn in connectors:
77         connlist.append(conn)
78
79     points.append(conn.Origin.ToPoint())
80
81     p1 = Konektor(points, connlist)
82     listpipe.append(p1)
83
84 p = []
85 for valve in valves:
86
87     points = []
88     connlist = []
89
90     try:
91         connectors = valve.MEPModel.ConnectorManager.Connectors
92     except:
93         try:
94             connectors = valve.ConnectorManager.Connectors
95         except:
96
97             continue
98
99     for conn in connectors:
100         connlist.append(conn)
101
102     points.append(conn.Origin.ToPoint())
103
104     p1 = Konektor(points, connlist)
105     listval.append(p1)
106
107
108 test = []
109 for pipe in listpipe:
110     test.append(pipe.point[1].Z)
111     if int(p1.point[0].Z) == int(pipe.point[1].Z):
112         ConnectTo(p1.konektor[1], pipe.konektor[0])
113
114     elif int(p1.point[1].Z) == int(pipe.point[0].Z):
115         ConnectTo(p1.konektor[0], pipe.konektor[1])
116
117 OUT = test
```

Run Save Changes Revert

Slika 37. Drugi dio Python skripte za spajanje strojarskih instalacija

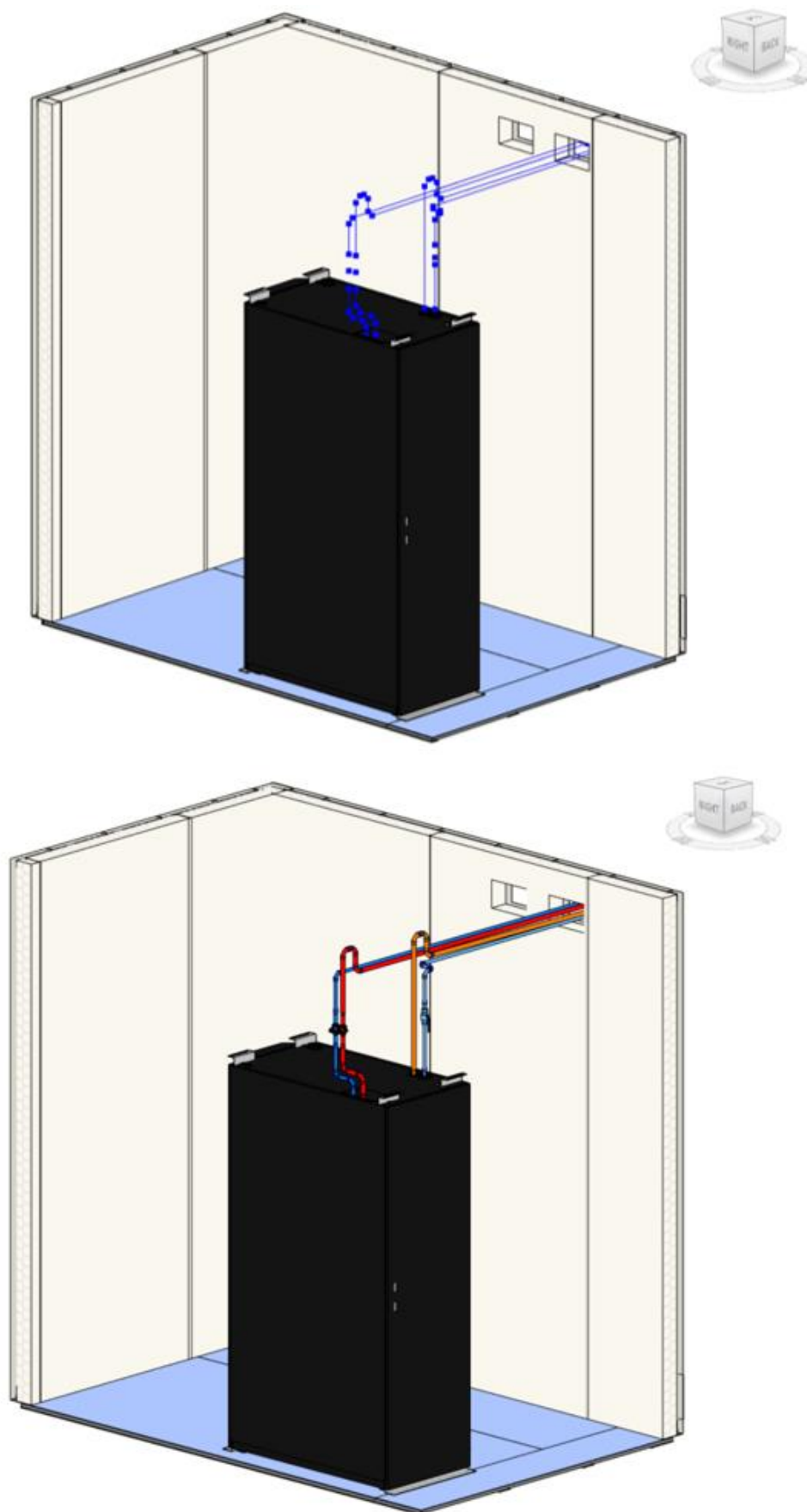
### 4.3. Rezultat programskog rješenja

Izlazne pozicije strojarskih instalacija s lijeve strane slike, dok su strojarske instalacije sa danim parametrima podatke programskog rješenja dobiju se unutar Revita nakon što se tipkom *Run* u Dinamu izvrsti cijeli algoritam za obradu i izradu modela. Generiranju se geometrijske pozicije strojarskih instalacija, tj. sve četiri vrste rashladnog sustava. Na slici 38 prikazane su geometrijske u koraku 5 drugog algoritma prikazani s desne strane slike u 2D pogledu unutar Revita.



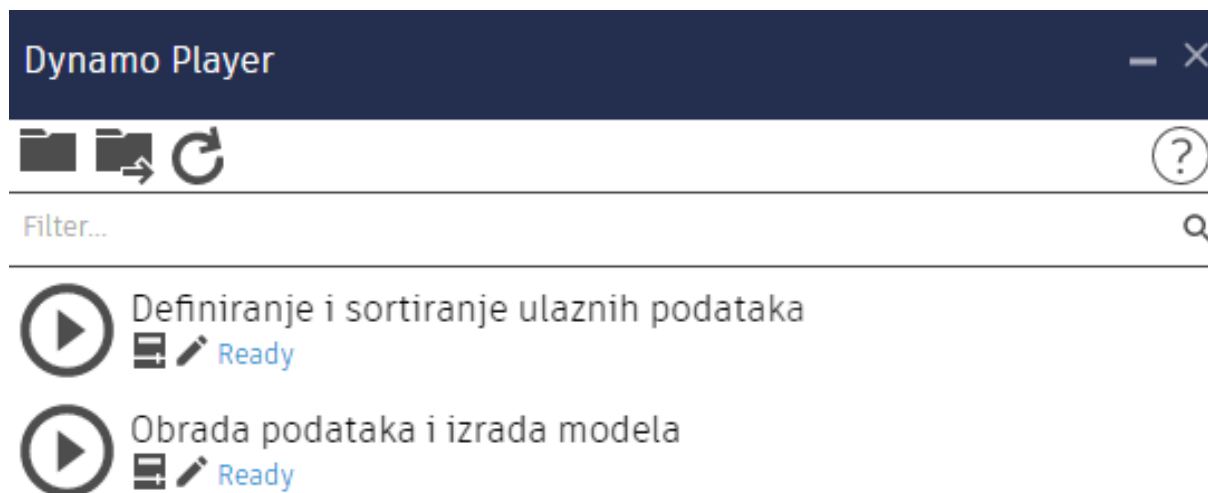
Slika 38. Prikaz generiranog modela u 2D obliku

Programsko rješenje 3D BIM modela rashladnog sustava podatkovnog centra od unutarnje jedinice do izlaza iz zgrade unutar Revita koje obuhvaća plinsku (cijevi crvene boje) i tekuću (cijevi plave boje) stranu rashladnog sustava, dobavu sanitarne vode (cijevi svjetloplave boje) i odvod kondenzata (cijevi narančaste boje) prikazano je na slici 39.



Slika 39. Prikaz programskog rješenja 3D BIM modela unutar Revita

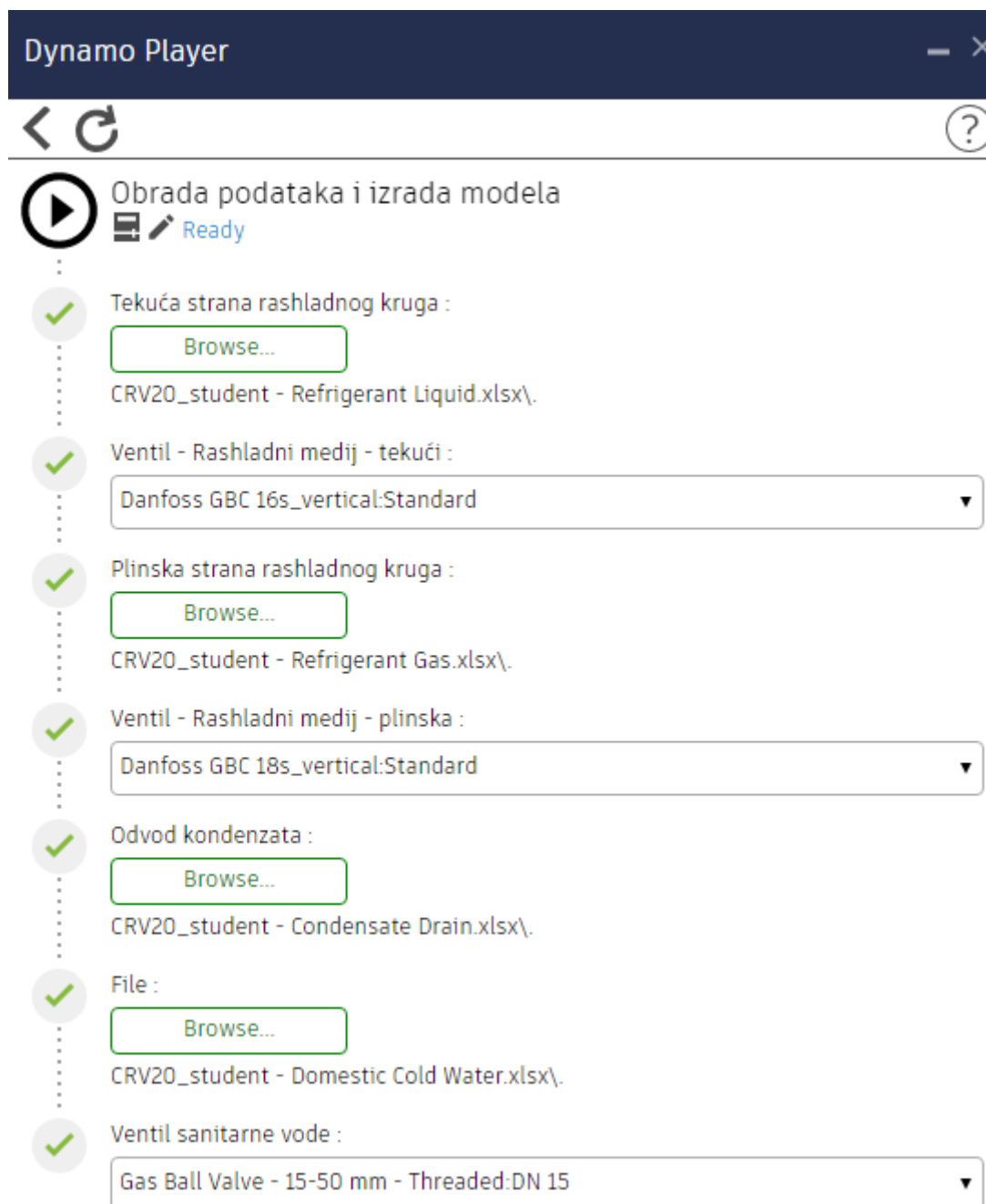
Korištenjem algoritma omogućuje se generiranje gotovog standardnog modela rashladnog sustava podatkovnog centra prilikom pokretanja novog projekta unutar Revita. Nakon uspješne implementacije u Dynamu, algoritami su prilagođeni za upotrebu u Dynamo Playeru, kao što je vidljivo na slici 40.



**Slika 40. Prikaz Dynamo Playera u sučelju Revita**

Kao ulazne varijable definirane su Excel datoteke sve četiri vrste rashladnog sustava uz mogućnost odabira potrebnog ventila odgovarajućeg sustava. Dynamo Player prikazuje listu koja sadrži odabrane ulazne podatke za generiranje 3D BIM modela rashladnog sustava koji se sastoji od unutarnje jedinice te mreže cjevovodnih instalacija do izlaza iz zgrade podatkovnog centra. Nakon toga, pokretanjem Dynamo Playera kreira se model rashladnog sustava sa svim parametrima i elementima unutar Revita, kao i na slici 39 upotrebom Dynamu. Time je ubrzana i pojednostavljena upotreba automatiziranog procesa 3D modeliranja rashladnog sustava podatkovnog centra.





Slika 41. Prikaz odabranih ulaznih podataka algoritma u Dynamo Playeru

## 5. ZAKLJUČAK

Suvremeni razvoj tehnologije uzrokuje intenzivniju potrebu za primjenom BIM-a sa svojom programskom podrškom pri projektiranju različitih struktura i građevinskih objekata. U sklopu BIM-a, programska rješenja nude fleksibilnost kod razvoja i implementiranja ideja, uz kolaboraciju projektanata različitih disciplina pomoću korištenja zajedničkog centralnog modela tijekom cijelog uporabnog vijeka objekta. Osim toga, omogućuje multidisciplinarnim projektnim timovima sprječavanje kolizija različitih projektnih sustava. Shodno tome, pristup velikoj količini podataka u 3D BIM modelu objekta, omogućuje automatizaciju dijelova procesa projektiranja. Programsko rješenje za automatizaciju procesa 3D modeliranja rashladnog sustava podatkovnog centra, od unutarne jedinice do izlaza iz zgrade, razrađena je u sklopu ovog diplomskog rada. Na primjeru rashladnog sustava prikazane su mogućnosti automatizacije procesa upotrebom Dynamo unutar Revita, BIM programske okoline za modeliranje. Tako je moguće izbjegavanje repetitivnih procesa, što rezultira povećanjem učinkovitosti projektiranja uz uštedu inženjerskih resursa. Cijeli algoritam programskog rješenja popraćen je detaljnim objašnjenjima i slikama kako bi se omogućila sljedivost i konciznost procesa. Nakon pokretanja algoritma rashladnog sustava podatkovnog centra i prikazom dobivenih rezultata modela, donio se zaključak o ispravnosti uporabe programskog rješenja za automatizaciju procesa 3D modeliranja. Međutim, upotreba algoritma pri procesu projektiranja u BIM okruženju još je uvijek na niskom stupnju primjene te postoji mnogo prostora za poboljšanje. Stoga, potrebno je primjenjivati i razvijati algoritme kako bi se postigla preciznija rješenja automatizacije procesa, te kako bi išli u korak s tehnologijom.

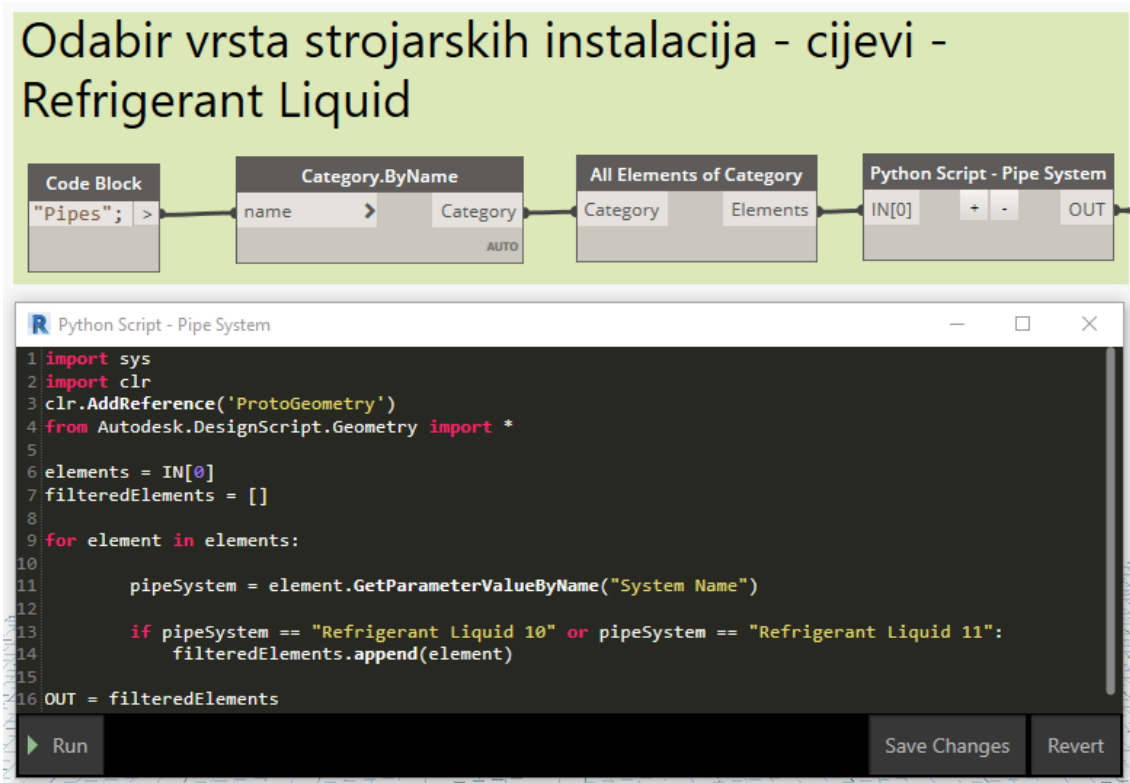
## LITERATURA

- [1] Hardin, B. BIM and Construction Management, Wiley, Indianapolis, IN. 2015.
- [2] Azhar, Salman. Building Information Modeling (BIM): Trends, Benefits, Risks, and Challenges for the AEC Industry. Leadership and management in engineering, 2011;11(3):241-252.
- [3] Bečirević D, Babić L, Cigrovski I. Od podataka laserskog skeniranja do BIM modela postojećeg stanja. Ekscentar. 2014;(17):87-92.  
Dostupno online: <https://hrcak.srce.hr/125537>
- [4] Uporaba BIM-a: <https://connect.bim360.autodesk.com/what-is-bim-in-construction>,  
Pristupljeno: 10. listopada 2021.
- [5] Kovačić I, Filzmoser M, Kiesel K, Oberwinter L, Mahdavi A. Izobrazba o primjeni BIM-a kao podrška integriranom projektiranju. Građevinar. 2015;67(6):537-546.  
Dostupno online: <https://hrcak.srce.hr/141170>
- [6] Jurčević M, Pavlović M, Šolman H. Opće smjernice za BIM pristup u graditeljstvu, Hrvatska komora inženjera građevinarstva, Zagreb, 2017.
- [7] Andabaka F, Dolaček-Alduk Z, Ecimović A, Galić D, Grošić M, Pavlović Cerinski M, Šimenić D, Šolman H. Smjernice za BIM pristup u infrastrukturnim projektima, Hrvatska komora inženjera građevinarstva, Zagreb, 2021.
- [8] Razine zrelosti BIM-a: <https://biblus.accasoftware.com/en/bim-maturity-levels-from-stage-0-to-stage-3/>,  
Pristupljeno: 14. listopada 2021.
- [9] Dimenzije BIM informacija: <https://axissteel.com/the-dimensions-of-bim-explained/>,  
Pristupljeno 20. listopada 2021.
- [10] Baroš T. Primjena BIM tehnologije i njezina pouzdanost u statičkoj analizi konstrukcija. Tehnički vjesnik. 2016; 23(4):1221-1226.  
Dostupno online: <https://hrcak.srce.hr/163820>
- [11] BIM revolucija: <https://blog.drawbotics.com/2018/11/07/the-bim-revolution-in-building-management/>,  
Pristupljeno: 20. listopada 2021.
- [12] Dimenzija BIM modela <https://www.united-bim.com/what-are-bim-dimensions-3d-4d-5d-6d-7d-bim-explained-definition-benefits/>,  
Pristupljeno: 24. listopada 2021.
- [13] BIM dimenzije: <https://biblus.accasoftware.com/en/bim-dimensions/>,  
Pristupljeno 24. listopada 2021.

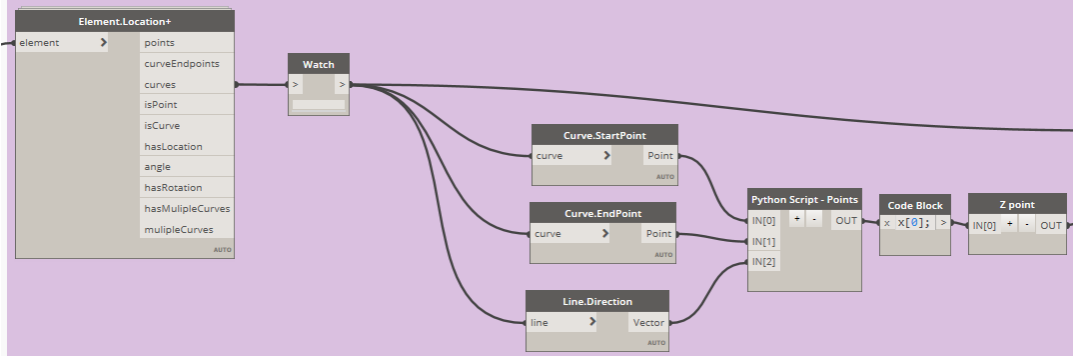
- 
- [14] BIM Management: <https://www.intelika.hr/proizvodi/bim-management>,  
Pristupljeno: 20. listopada 2021.
- [15] Razvoj BIM-a kroz povijest: <https://www.archdaily.com/302490/a-brief-history-of-bim>,  
Pristupljeno: 30. listopada 2021.
- [16] BIM tehnologija: <https://bim-hrvatska.hr/definicija-bim-tehnologije/>,  
Pristupljeno: 31. listopada 2021.
- [17] Vizualno programiranje: <https://bim-hrvatska.hr/vizualno-programiranje-kao-pomoc-projektiranju/>,  
Pristupljeno: 2. studenog 2021.
- [18] Dzambazova T, Krygiel E, Demchak G. *Introducing Revit Architecture 2010: BIM for Beginners*, 2009.
- [19] Autodesk Revit: <https://www.prior.hr/programi/autodesk-revit/>,  
Pristupljeno: 2. studenog 2021.
- [20] Revit Struktura: <https://www.intelika.hr/proizvodi/autodesk/revit/65-revit-structure>,  
Pristupljeno: 3. studenog 2021.
- [21] Dynamo za Revit: <https://www.archdaily.com/945566/learn-more-about-dynamo-for-revit-features-functions-and-news>,  
Pristupljeno: 3. studenog 2021.
- [22] Dynamo: [https://primer.dynamobim.org/en/01\\_Introduction/1-2\\_what\\_is\\_dynamo.html](https://primer.dynamobim.org/en/01_Introduction/1-2_what_is_dynamo.html),  
Pristupljeno: 3. studenog 2021.
- [23] Python: <https://www.python.org/>,  
Pristupljeno: 6. studenog 2021.

## **PRILOZI**

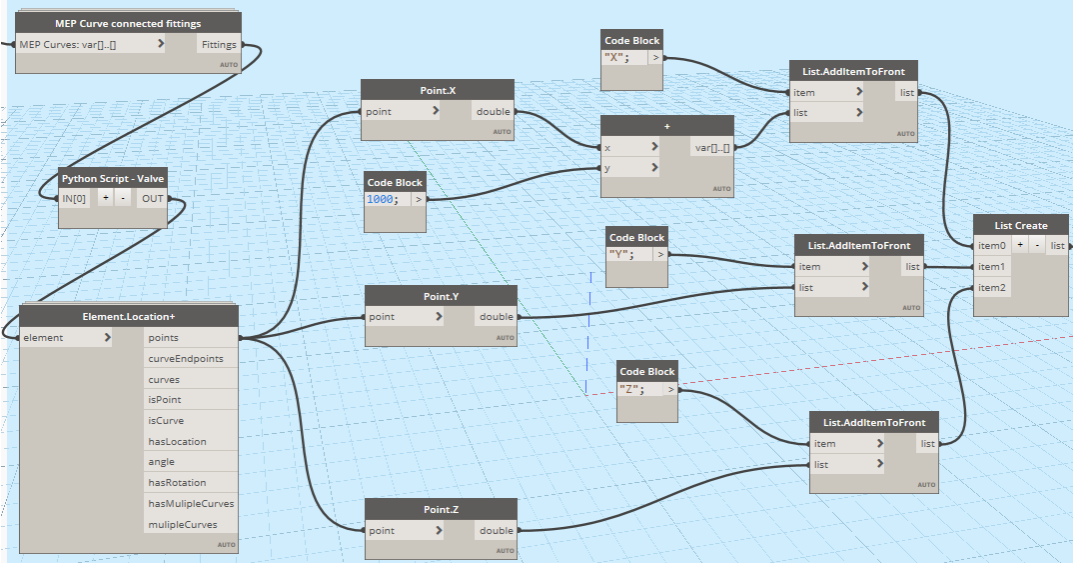
- I. CD-R disk
- II. Programski algoritam

**Programski algoritam za definiranje i sortiranje izlaznih podataka standardnog modela**Tekuća strana rashladnog kruga (*Refrigerant Liquid*)

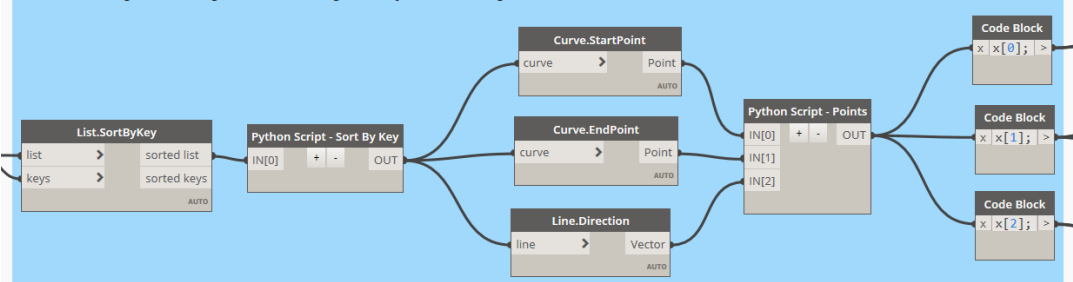
### Preuzimanje geometrijskih karakteristika strojarskih cijevi



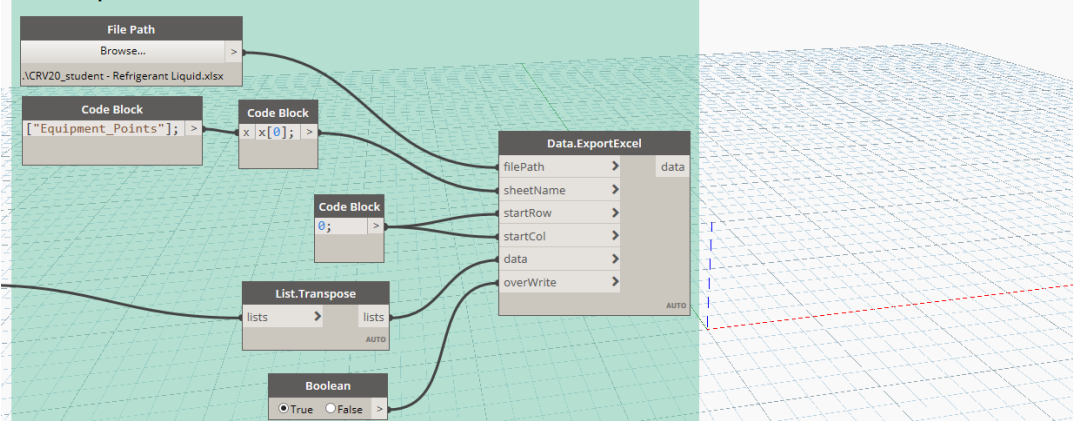
### Izvoz podatak strojarskih cijevi u Excel

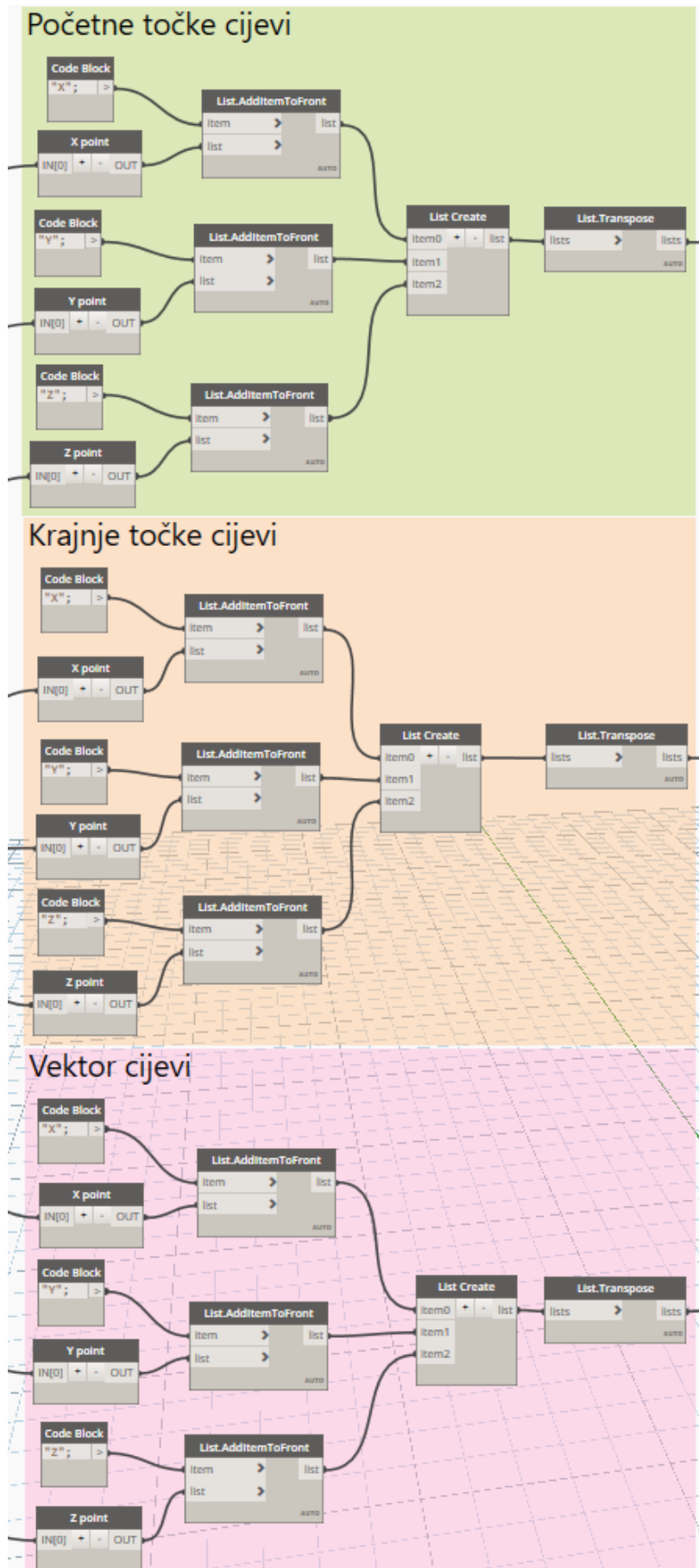


### Sortiranje strojarskih cijevi po udaljenosti



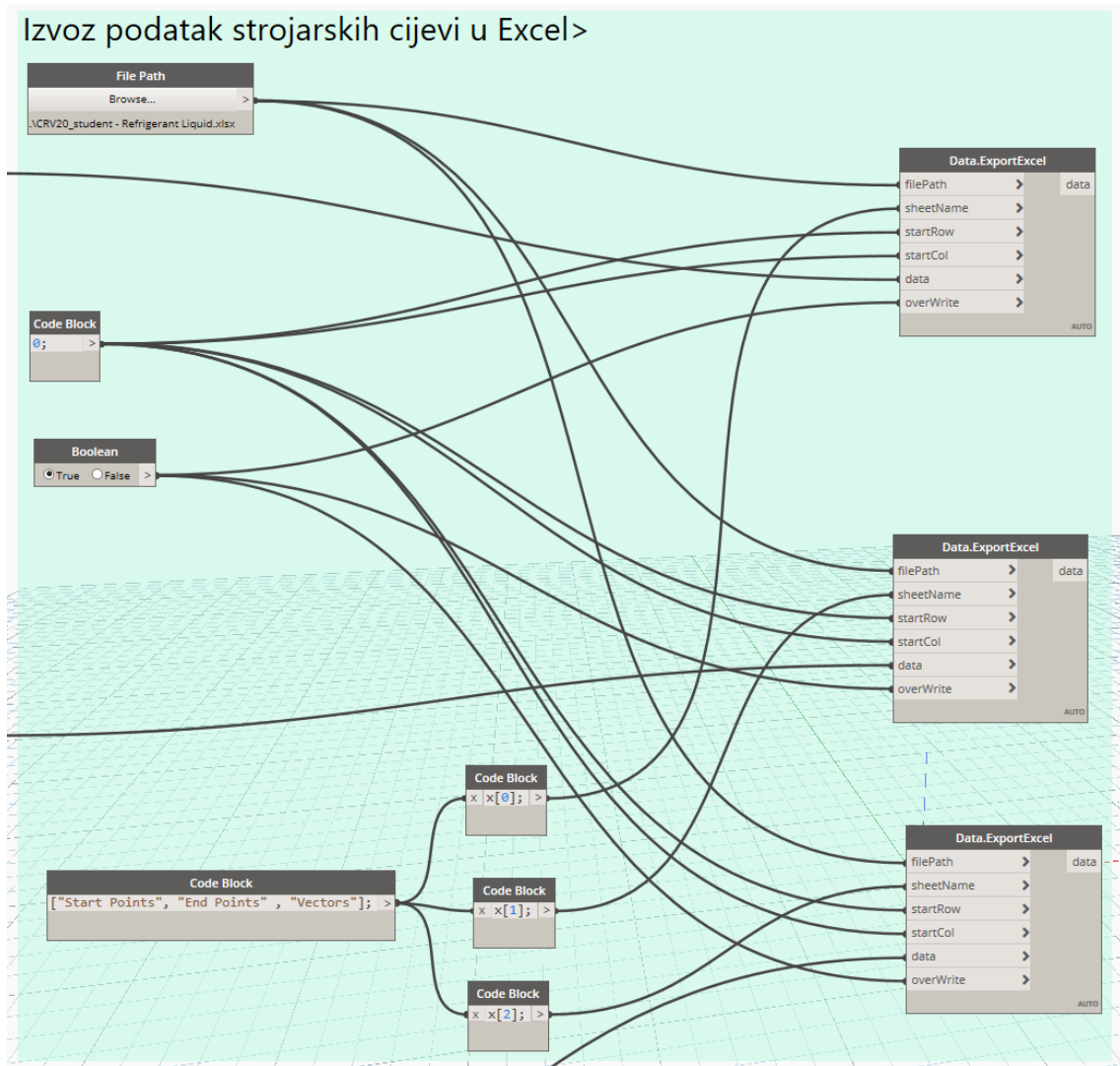
### Izvoz podatak ventila u Excel







### Izvoz podatak strojarskih cijevi u Excel >



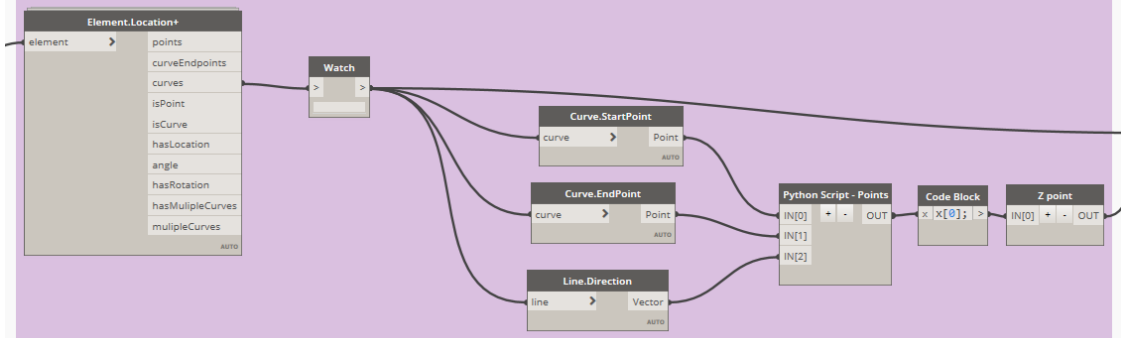
Plinska strana rashladnog kruga (*Refrigerant Gas*)

## Odabir vrsta strojarskih instalacija - cijevi Refrigerant Gas

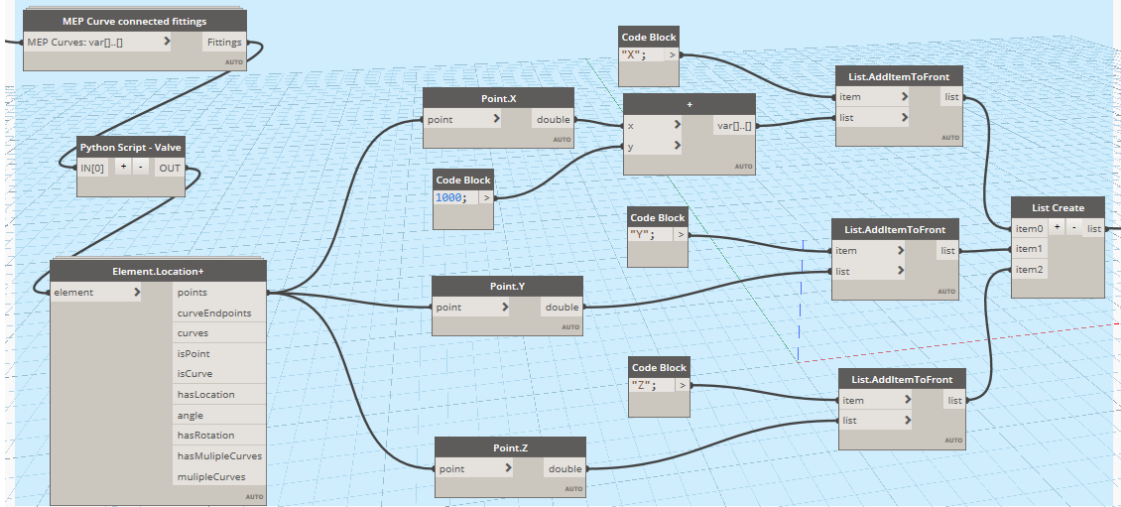
```
1 import sys
2 import clr
3 clr.AddReference('ProtoGeometry')
4 from Autodesk.DesignScript.Geometry import *
5
6
7
8 elements = IN[0]
9 filteredElements = []
10
11 for element in elements:
12
13     pipeSystem = element.GetParameterValueByName("System Name")
14     if pipeSystem == "Refrigerant Gas 8" or pipeSystem == "Refrigerant Gas 9":
15
16         filteredElements.append(element)
17
18
19 OUT = filteredElements
20
```

Run Save Changes Revert

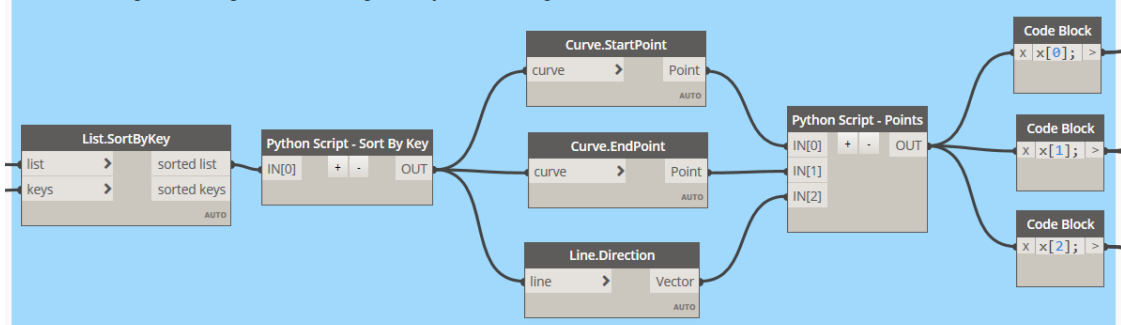
### Preuzimanje geometrijskih karakteristika strojarskih cijevi



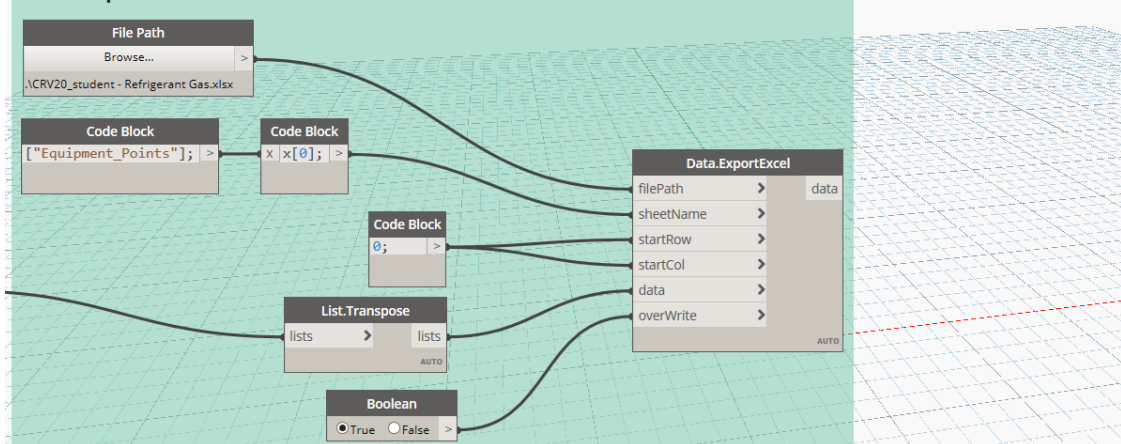
### Preuzimanje geometrijskih karakteristika ventila

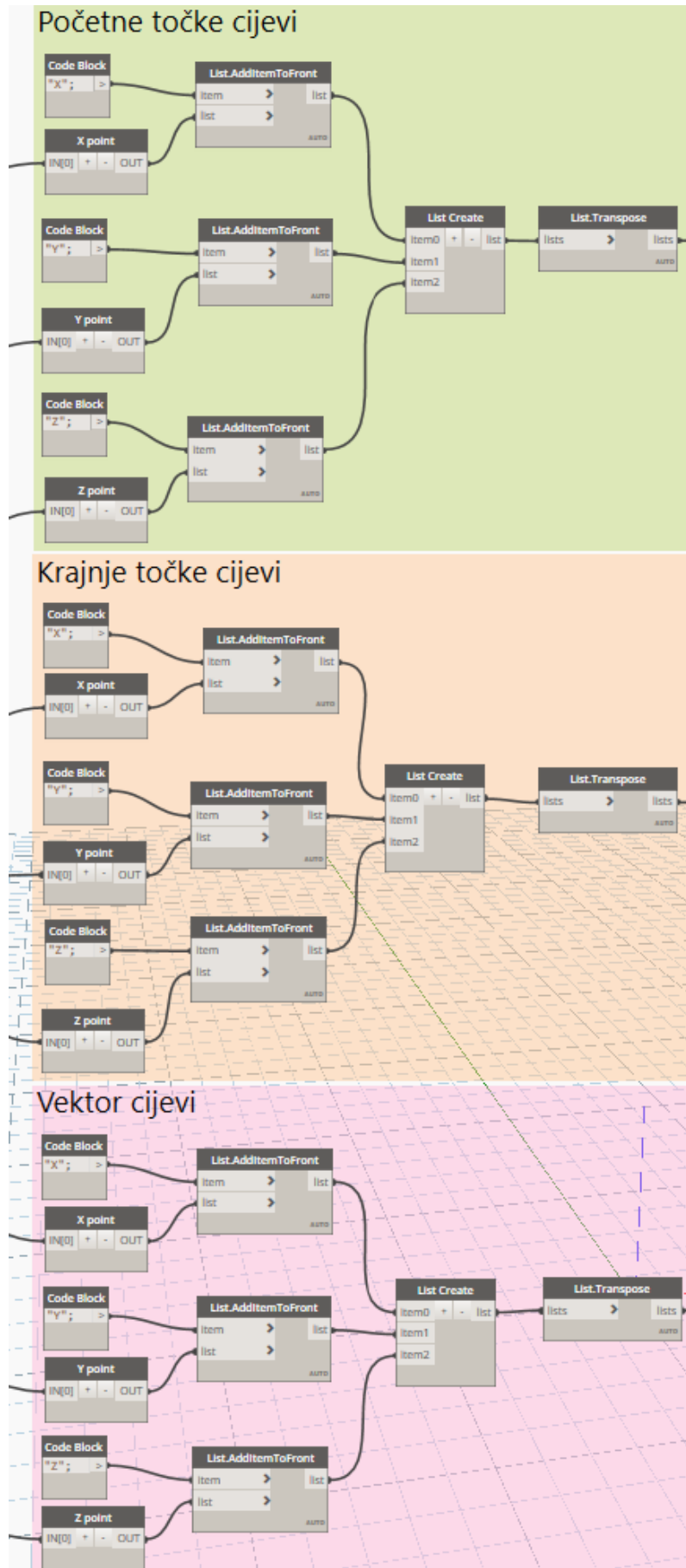


### Sortiranje strojarskih cijevi po udaljenosti

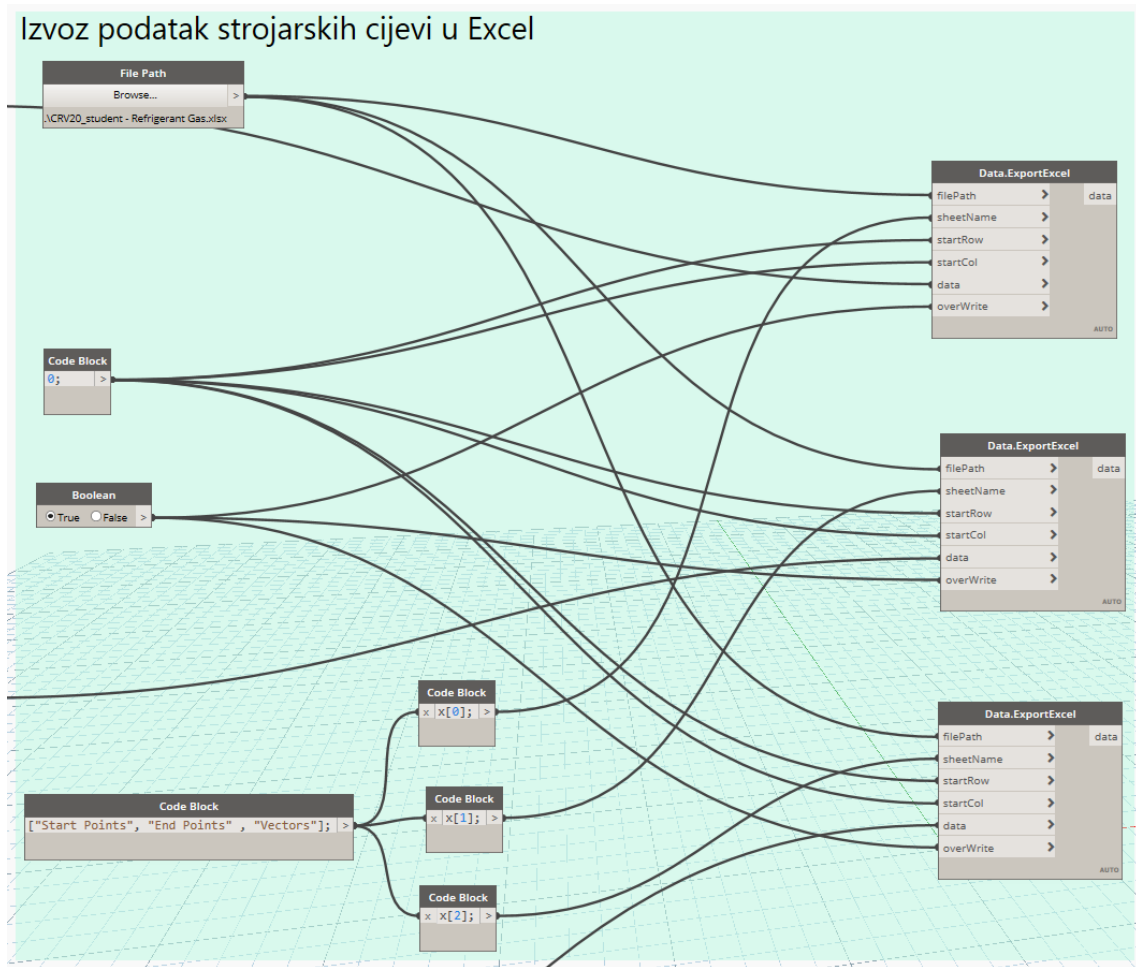


### Izvoz podatak ventila u Excel





### Izvoz podataka strojarskih cijevi u Excel



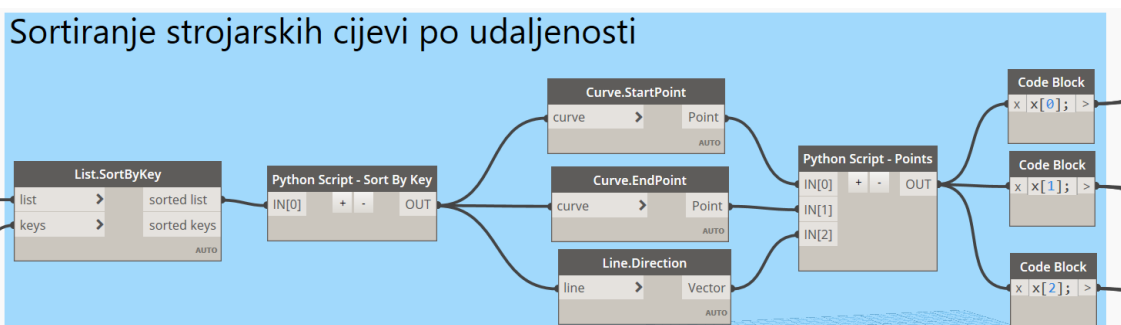
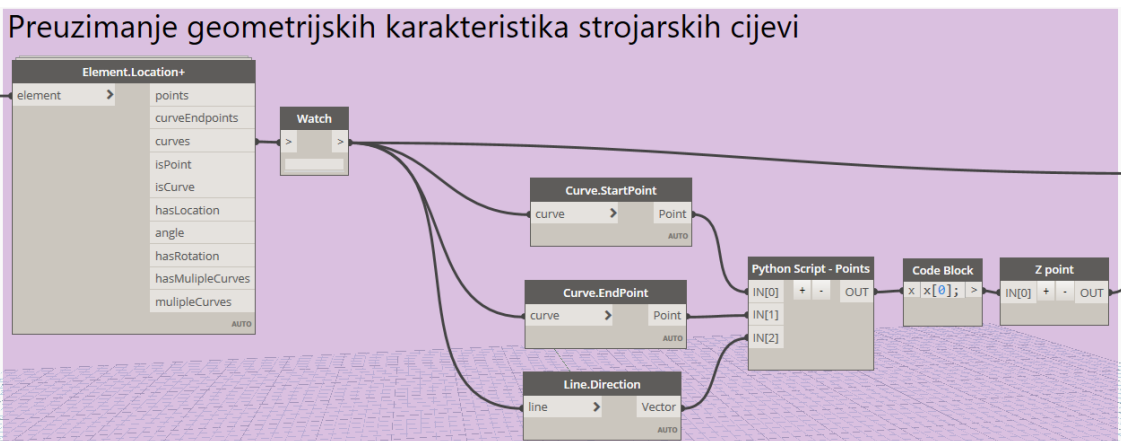
Odvod kondenzata (*Condensate Drain*)

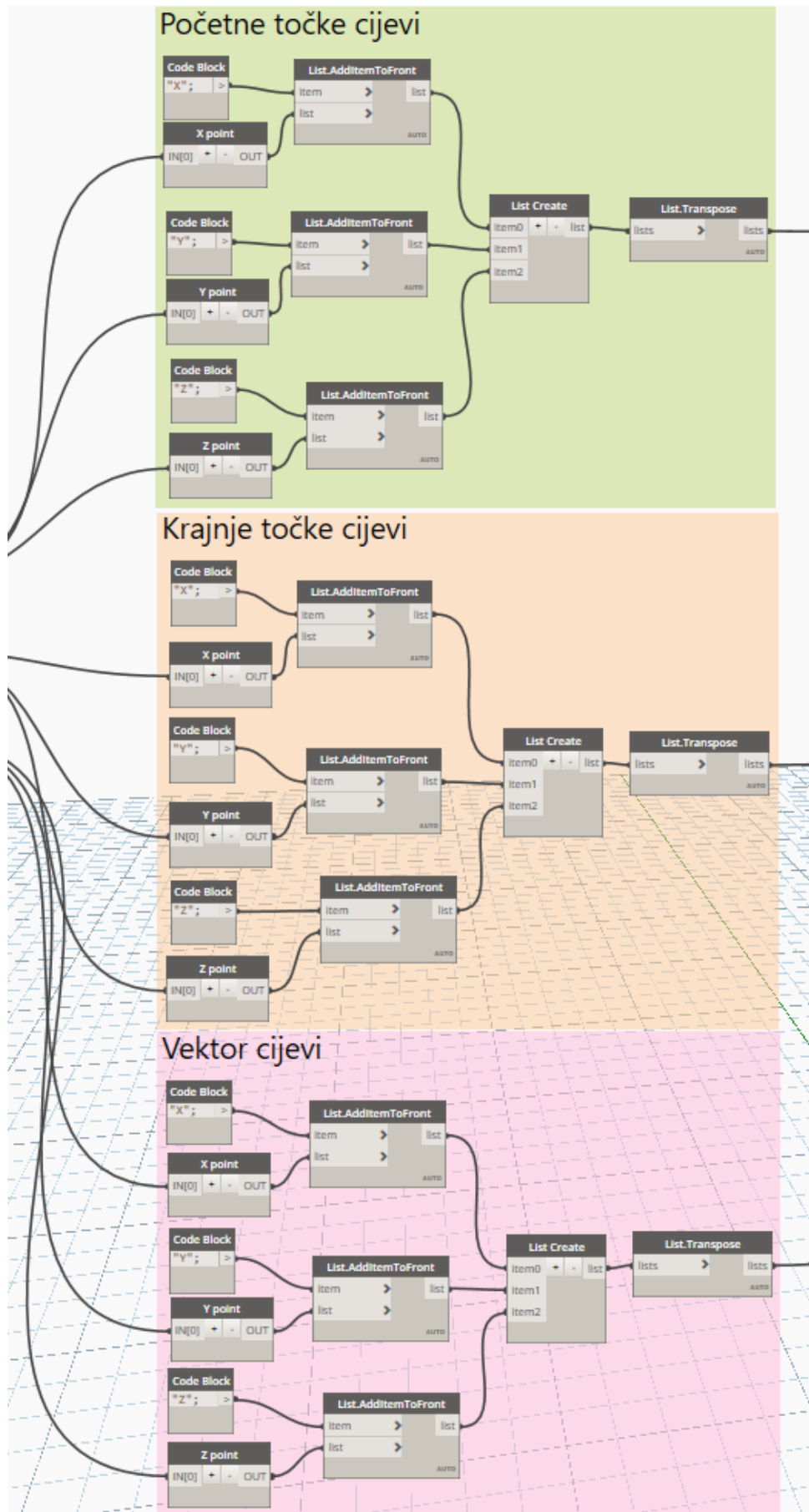


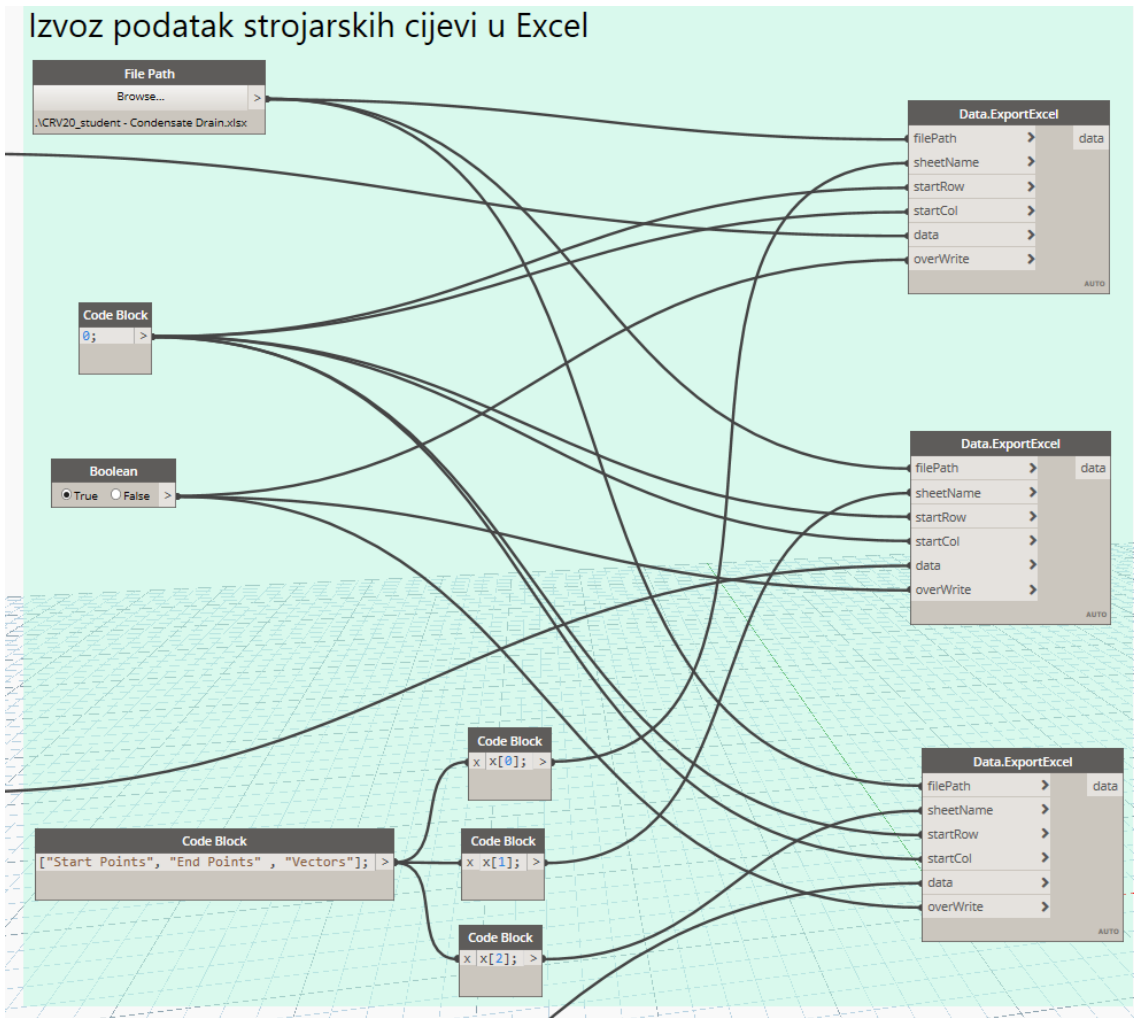
```

Python Script - Pipe System
1 import sys
2 import clr
3 clr.AddReference('ProtoGeometry')
4 from Autodesk.DesignScript.Geometry import *
5
6
7
8 elements = IN[0]
9 filteredElements = []
10
11 for element in elements:
12
13     pipeSystem = element.GetParameterValueByName("System Name")
14     if pipeSystem == "Condensate Drain 3":
15
16         filteredElements.append(element)
17
18
19 OUT = filteredElements
    
```

Run Save Changes Revert



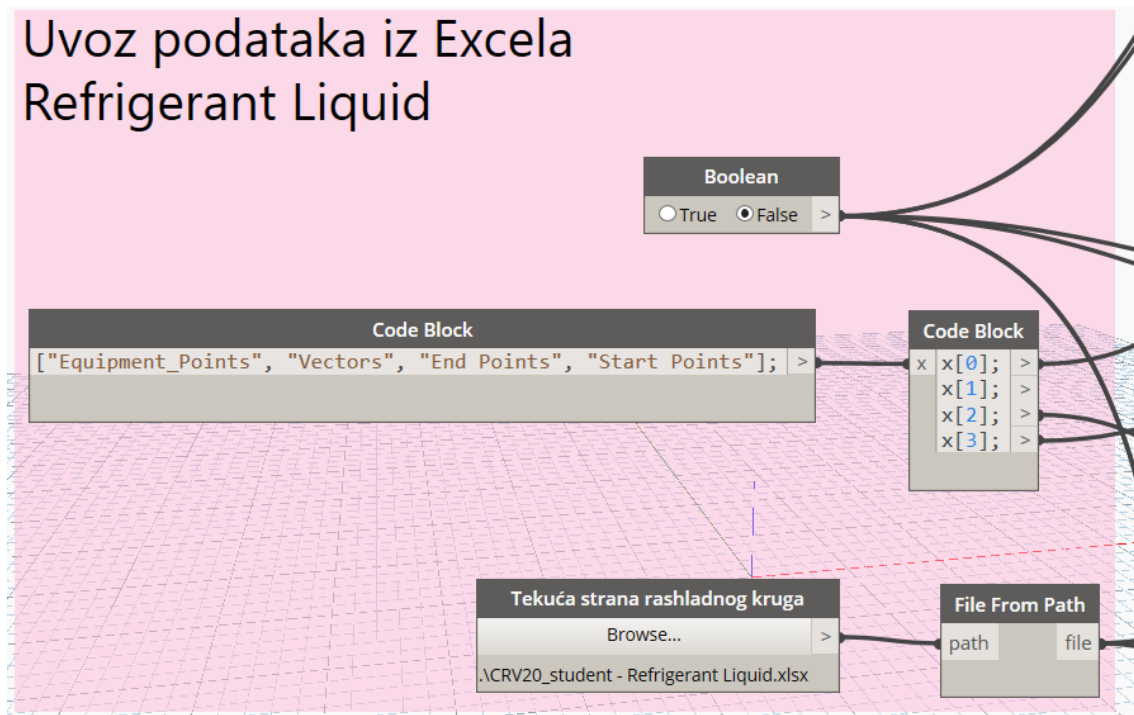




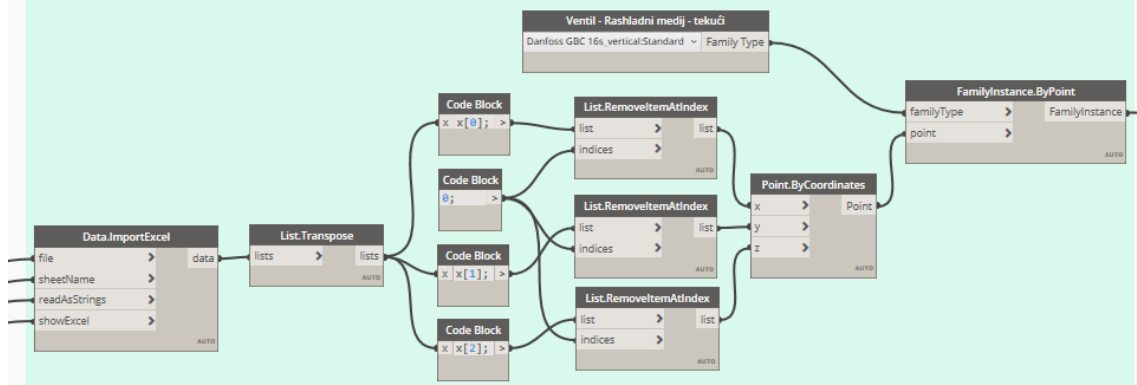


## Programski algoritam za obradu i izradu novog modela

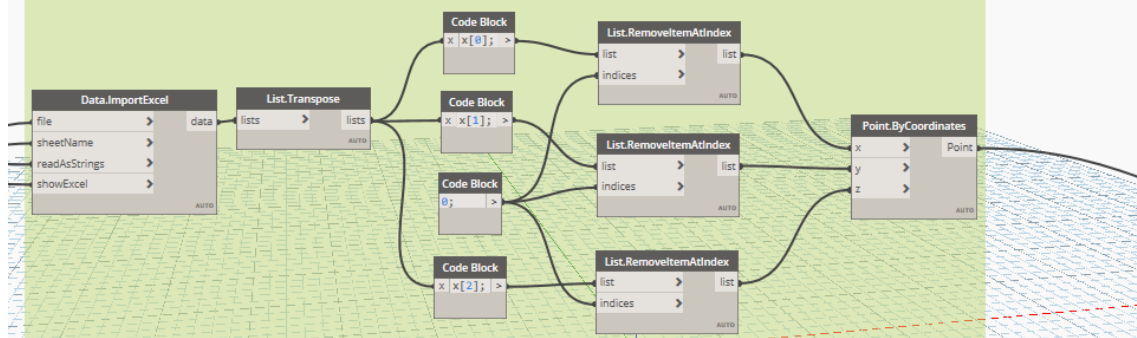
Tekuća strana rashladnog kruga (*Refrigerant Liquid*)



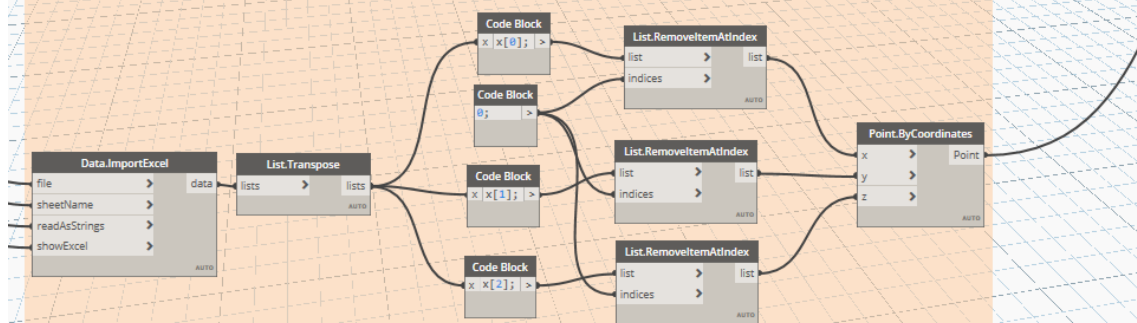
### Razvrstavanje geometrije ventila

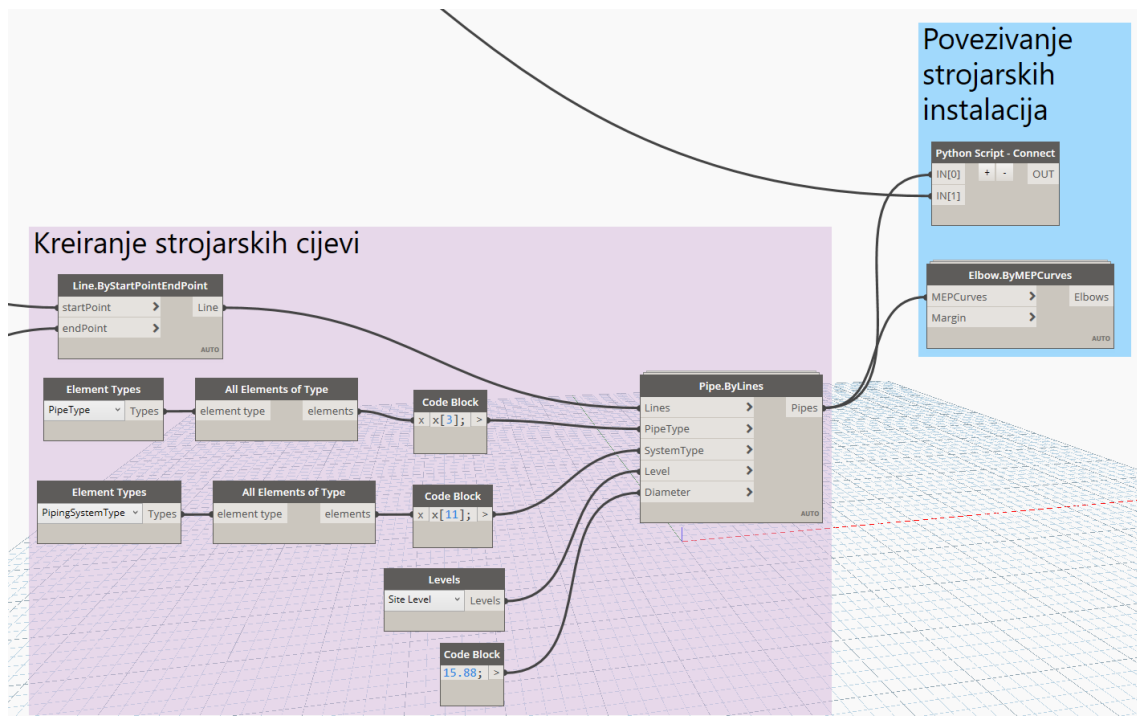


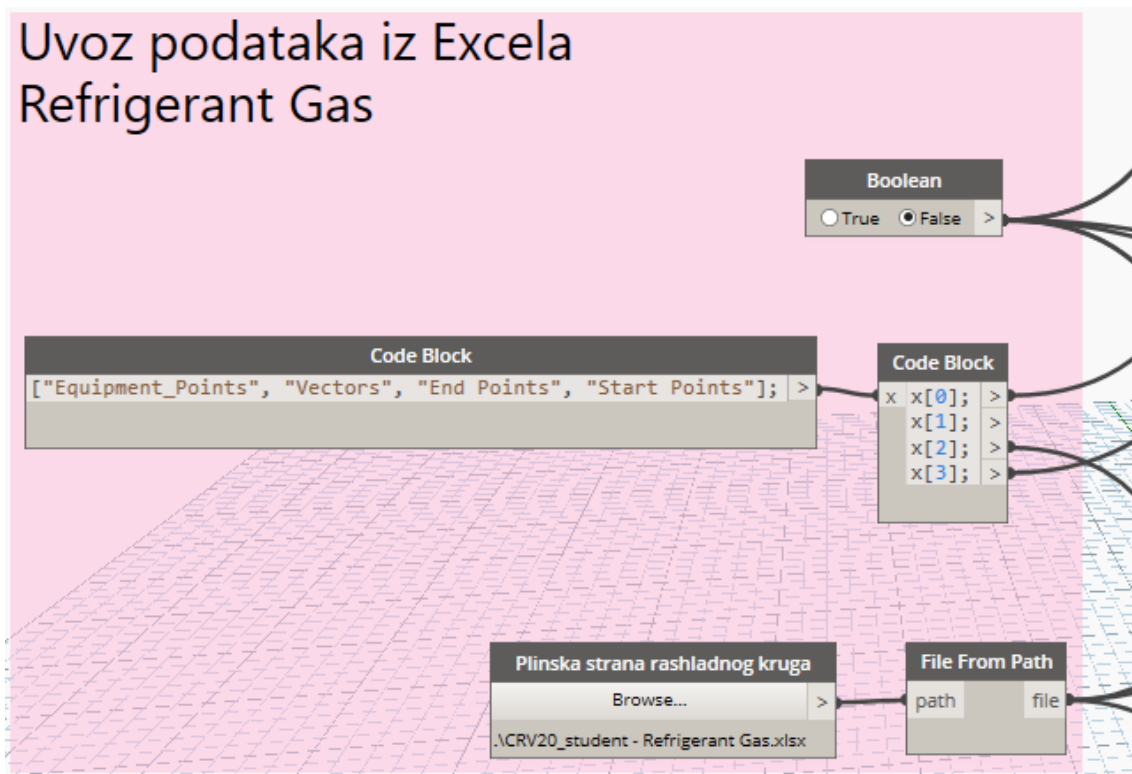
### Razvrstavanje početnih točki cijevi



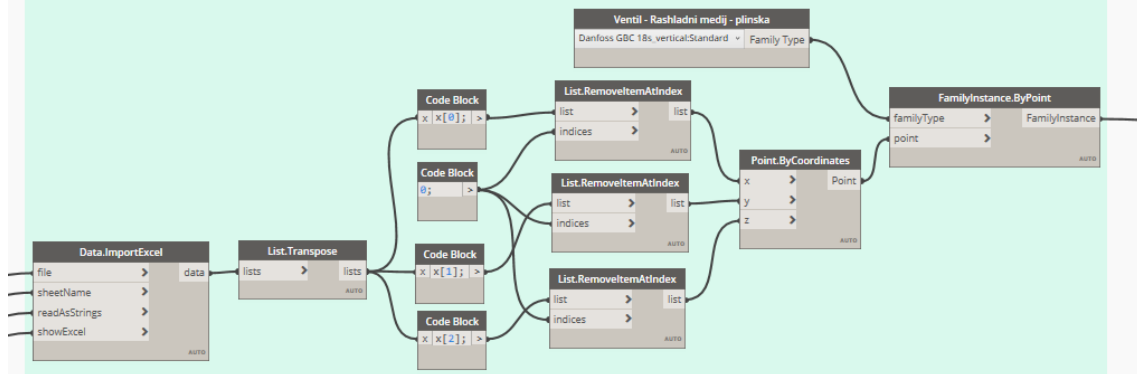
### Razvrstavanje krajnjih točki cijevi



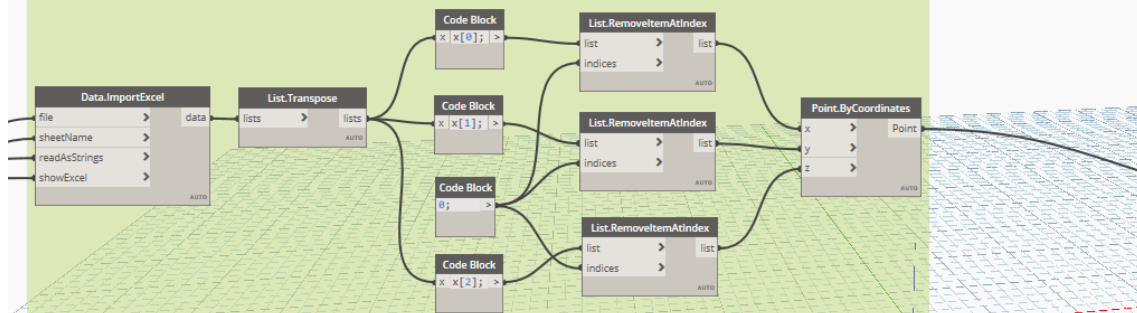


Plinska strana rashladnog kruga (*Refrigerant Gas*)

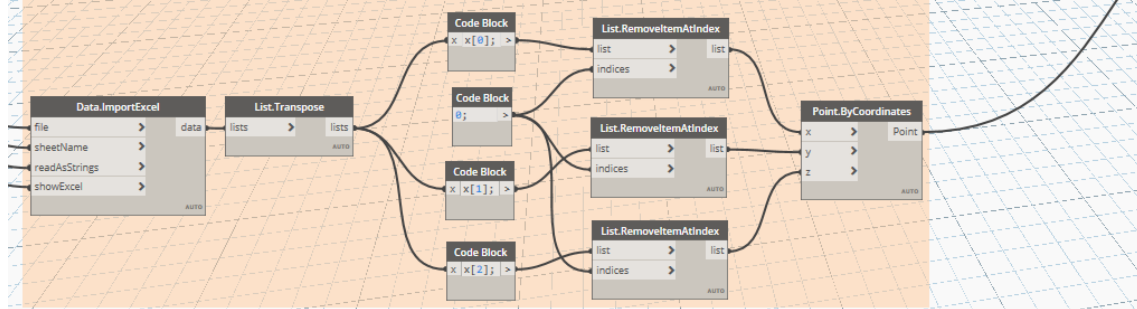
### Razvrstavanje geometrije ventila

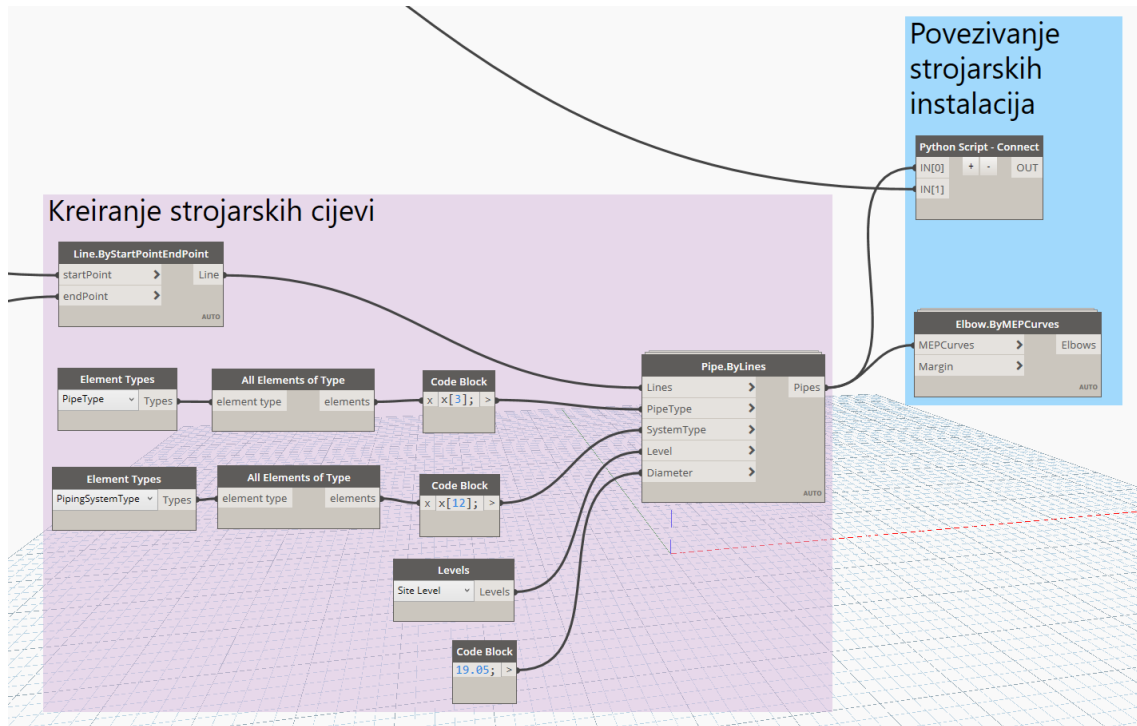


### Razvrstavanje početnih točki cijevi

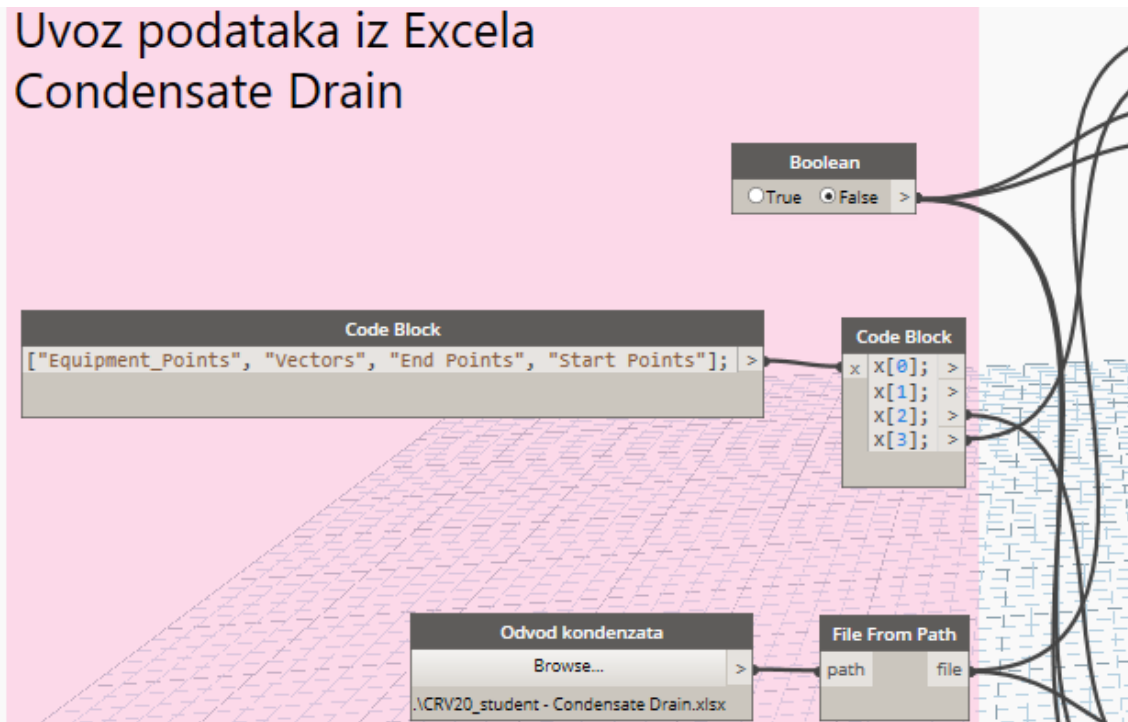


### Razvrstavanje krajnjih točki cijevi

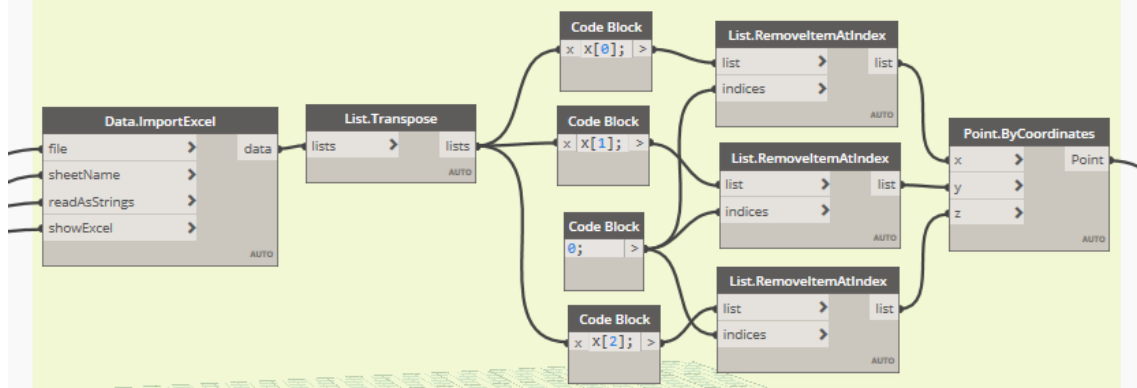




Odvod kondenzata (*Condensate Drain*)



### Razvrstavanje početnih točki cijevi



### Razvrstavanje krajnjih točki cijevi

