

UNIVERSIDADE ESTADUAL DE PONTA GROSSA  
SETOR DE ENGENHARIAS, CIÊNCIAS AGRÁRIAS E DE TECNOLOGIAS  
DEPARTAMENTO DE ENGENHARIA CIVIL

GUSTAVO VARELA MORCHE

*DESIGN* GENERATIVO BASEADO EM CONCEITOS DE BIM E *LEAN*  
*CONSTRUCTION* PARA O DESENVOLVIMENTO DE PROJETOS

PONTA GROSSA

2022

GUSTAVO VARELA MORCHE

*DESIGN* GENERATIVO BASEADO EM CONCEITOS DE BIM E *LEAN*  
*CONSTRUCTION* PARA O DESENVOLVIMENTO DE PROJETOS

Trabalho de Conclusão de Curso  
apresentado para a obtenção do título de  
Bacharel em Engenharia Civil na  
Universidade Estadual de Ponta Grossa.

Orientador: Prof. Me. Elias Pereira

PONTA GROSSA

2022

GUSTAVO VARELA MORCHE

*DESIGN* GENERATIVO BASEADO EM CONCEITOS DE BIM E *LEAN*  
*CONSTRUCTION* PARA O DESENVOLVIMENTO DE PROJETOS

Trabalho de Conclusão de Curso apresentado para obtenção do título de Bacharel  
em Engenharia Civil na Universidade Estadual de Ponta Grossa.

Ponta Grossa, 16 de dezembro de 2022.

---

Prof. Me. Elias Pereira – Orientador

Mestre em Engenharia Civil

Departamento de Engenharia Civil - Universidade Estadual de Ponta Grossa

---

Prof. <sup>a</sup> Dra. Nisiane Madalozzo Wambier

Doutora em Gestão do Território

Departamento de Engenharia Civil - Universidade Estadual de Ponta Grossa

---

Prof. <sup>a</sup> Ma. Eloise Aparecida Langaro

Mestra em Engenharia Civil

Departamento de Engenharia Civil - Universidade Estadual de Ponta Grossa

## RESUMO

Nesta monografia trabalhou-se com informações relacionadas a BIM (*Building Information Modeling*), desperdícios previstos por *Lean Construction* e como esses dois conceitos se unem através de *design* generativo utilizando programação visual para trabalhar contra desperdícios no desenvolvimento de projetos. BIM é através de tecnologia, obter em projeto, um produto idêntico ao visado na realidade, trazendo conceitos e formas de trabalhar para se chegar ao produto real, enquanto *Lean Construction* é a maneira de pensar o desenvolvimento, objetivando a obtenção do melhor produto eliminando no processo de desenvolvimento desperdícios, já *design generativo* é a maneira de desenvolvimento de tal produto através da utilização de computadores. O objetivo geral desta monografia foi unir tais conceitos e tecnologias utilizando programação visual (*design* generativo utilizando programação), focando em analisar a aplicação em projetos de engenharia civil para a obtenção de redução, eliminação e apontamento de desperdícios conhecidos e pré-determinados trazidos pelo por *Lean Construction*. Para atender a este objetivo, utilizou-se o software de projetos Revit® para a criação de projetos teste de comparação, enquanto para o desenvolvimento da programação visual, utilizou-se o software de programação visual Dynamo®. A programação visual desenvolvida em Dynamo® gerou um processo simplificado de *design* generativo, o qual permitiu recriar os projetos testes em Revit®, realizando comparações entre o produto manual e o produto desenvolvido com programação visual. Com a utilização do processo apresentado, concluiu-se que, embora a programação visual tenha se mostrado incompleta para o desenvolvimento de um processo de *design* generativo totalmente autônomo, a união de conceitos de BIM e *Lean Construction* no desenvolvimento de projetos se mostra benéfico, devido a capacidade de processos BIM (conceitos e tecnologias de trabalho) de preverem e apresentarem formas de trabalharem com os desperdícios apontados por *Lean Construction*.

**Palavras-chave:** *Design* generativo. BIM. *Lean Construction*. Projetos.

## ABSTRACT

This final paper worked with information related to BIM (Building Information Modeling), foreseen wastes by Lean Construction and how these two concepts join together through generative design using visual programming to work against waste in the development of projects. BIM is through technology, acquire in design, an identical product foreseen in reality bringing concepts and ways of working to obtain the real product, while Lean Construction the way of thinking on the development aiming to obtain the best product, eliminating wastes in the development process, and, generative design, is the way of development of the product using computers. The general objective of this final paper was to join these concepts and technology using visual programming (generative design using visual programming) with focus in analyzing the application in civil engineering projects to obtain reduction, elimination and observation of known and predetermined wastes by *Lean Construction*. To the development of such technology, it was used the project software Revit<sup>®</sup> to the creation of test projects to create comparison, while to the development of visual programming, it was used the visual programming software Dynamo<sup>®</sup>. The visual program developed in Dynamo<sup>®</sup> created a simplified process of generative design, which one allowed the recreation of the Revit<sup>®</sup> test projects, creating the comparison between the manual product and the product developed with visual programming. With the use of the presented process, it was concluded that, while the visual programming has shown an incomplete development of an autonomous process of generative design, the union of BIM and Lean Construction concepts on the development of the design have shown benefits because of the capacity of BIM process (concepts and technology of work) to foreseen and present ways to work with the wastes pointed by Lean Construction.

**Keywords:** Generative design. BIM, Lean Construction. Design.

## LISTA DE FIGURAS

Figura 01 – LOD ( <i>Level of Development</i> ) de projetos .....	13
Figura 02 – exemplo de nós de Dynamo® .....	21
Figura 03 – Exemplo de nó de IronPython .....	23
Figura 04 – Fluxograma da Metodologia .....	27
Figura 05 – Planta baixa simplificada do modelo arquitetônico do projeto teste 01 ..	32
Figura 06 – Isométrico simplificado do modelo arquitetônico do projeto teste 01 .....	32
Figura 07 – Planta baixa simplificada do modelo elétrico do projeto teste 01 .....	33
Figura 08 – Isométrico simplificado do modelo elétrico do projeto teste 01 .....	34
Figura 09 - Programação visual (modelagem) baseada no teste 01 .....	35
Figura 10 - Programação visual (documentação e organização bidimensional) baseada no teste 01 .....	36
Figura 11 - Programação visual (organização tridimensional) baseada no teste 01 ..	37
Figura 12 - Programação visual completa baseada no teste 01 .....	38
Figura 13 – Planta baixa simplificada do projeto teste 01 desenvolvido com a programação visual (modelagem, documentação e organização) baseada no teste 01 .....	39
Figura 14 - Isométrico simplificado do projeto teste 01 desenvolvido com a programação visual (modelagem, documentação e organização) baseada no teste 01 .....	40
Figura 15 – Planta baixa simplificada do modelo elétrico do projeto teste 02 .....	41
Figura 16 – Isométrico simplificado do modelo elétrico do projeto teste 02 .....	42
Figura 17 - Programação visual (modelagem 1 de 02) baseada no teste 02 .....	43
Figura 18 - Programação visual (modelagem 02 de 02) baseada no teste 02 .....	43
Figura 19 – Visualização de Dynamo Player da programação visual da Figura 17...44	
Figura 20 – Visualização de Dynamo Player da programação visual da Figura 18...44	
Figura 21 - Programação visual (documentação e organização bidimensional) baseada no teste 02 .....	45
Figura 22 - Programação visual (documentação e organização tridimensional) baseada no teste 02 .....	46
.....	46
Figura 23 – Planta baixa simplificada do projeto teste 02 desenvolvido com a programação visual (modelagem, documentação e organização) baseada do teste 02 .....	47

Figura 24 - Isométrico simplificado do projeto teste 01 desenvolvido com a programação visual (modelagem, documentação e organização) baseada no teste 01 .....	48
Figura 25 – Planta baixa simplificada do modelo arquitetônico do projeto teste 03 ..	49
Figura 26 – Isométrico simplificado do modelo arquitetônico do projeto teste 03 .....	50
Figura 27 – Planta baixa simplificada do modelo elétrico do projeto teste 03 (01/02) .....	50
Figura 28 – Isométrico simplificado do modelo elétrico do projeto teste 03 (01/02) ..	51
Figura 29 – Planta baixa simplificada do modelo elétrico do projeto teste 03 (02/02) .....	52
Figura 30 – Isométrico simplificado do modelo elétrico do projeto teste 03 (02/02) ..	53
Figura 31 - Programação visual (parte da modelagem) baseada no teste 03 .....	54
Figura 32 – Planta baixa simplificada do modelo elétrico do projeto teste 03 baseada na programação visual teste 03 (01/03) .....	55
Figura 33 – Planta baixa simplificada do modelo elétrico do projeto teste 03 baseada na programação visual teste 03 (02/03) .....	56
Figura 34 – Planta baixa simplificada do modelo elétrico do projeto teste 03 baseada na programação visual teste 03 (03/03) .....	57
Figura 35 – Isométrico simplificado do modelo elétrico do projeto teste 03 baseada na programação visual teste 03 (01/03) .....	58
Figura 36 – Isométrico simplificado do modelo elétrico do projeto teste 03 baseada na programação visual teste 03 (02/03) .....	59
Figura 37 – Isométrico simplificado do modelo elétrico do projeto teste 03 baseada na programação visual teste 03 (03/03) .....	60
Figura 38 – Programação visual final baseada no teste 03 apresentada no Dynamo Player .....	63

## **LISTA DE TABELAS**

Tabela 01 – Análise de conceitos com base na programação do teste 01 .....	65
Tabela 02 – Análise de conceitos com base na programação do teste 02 .....	66
Tabela 03 – Análise de conceitos com base na programação do teste 03 .....	67
Tabela 04 – Quantitativo extraído do Revit .....	68
Tabela 05 – Quantitativo extraído do Revit® - modelo com base no projeto teste 03 desenvolvido manualmente.....	69

## **LISTA DE ABREVIATURAS E SIGLAS**

BIM	BUILDING INFORMATION MODELING
ISO	INTERNATIONAL STANDARDIZATION ORGANIZATION
LOD	LEVEL OF DEVELOPMENT
AEC	ARCHITECTURE, ENGINEERING AND CONSTRUCTION
CAD	COMPUTER-AIDED MODELING

## **LISTA DE SÍMBOLOS**

- rvt. Extensão principal de arquivo de projeto Revit
- rfa. Extensão de dados de família de arquivo Revit
- rte. Extensão de template Revit
- dyn. Extensão de arquivo de programação visual Dynamo

## SUMÁRIO

<b>1</b>	<b>INTRODUÇÃO</b> .....	<b>6</b>
1.1	JUSTIFICATIVA .....	9
1.2	OBJETIVOS .....	10
1.2.1	Objetivo geral .....	10
1.2.2	Objetivos Específicos .....	10
<b>2</b>	<b>REVISÃO BIBLIOGRÁFICA</b> .....	<b>11</b>
2.1	BIM ( <i>BUILDING INFORMATION MODELING</i> ).....	11
2.2	<i>LEAN CONSTRUCTION</i> .....	16
2.3	BIM E <i>LEAN CONSTRUCTION</i> (TEORIA DE BIM 9D) .....	18
2.4	REVIT® .....	19
2.5	DYNAMO® .....	20
2.5.1	Programação visual e programação escrita em Dynamo® .....	21
2.6	<i>DESIGN</i> GENERATIVO UTILIZANDO REVIT® E DYNAMO® .....	25
<b>3</b>	<b>METODOLOGIA</b> .....	<b>26</b>
3.1	PARTICIPAÇÃO DO REVIT® .....	28
3.2	PARTICIPAÇÃO DO DYNAMO® .....	28
3.3	CONCEITOS UTILIZADOS .....	29
3.3.1	ANÁLISE DOS CONCEITOS UTILIZADOS .....	29
3.4	EQUIPAMENTO UTILIZADO .....	30
<b>4</b>	<b>DESENVOLVIMENTO</b> .....	<b>31</b>

4.1	PROJETO TESTE 01 .....	32
4.1.1	PROGRAMAÇÃO VISUAL BASEADA NO PROJETO TESTE 01 .....	35
4.2	PROJETOS TESTE 02.....	40
4.2.1	PROGRAMAÇÃO VISUAL BASEADA NO PROJETO TESTE 02 .....	42
4.3	PROJETOS TESTE 03.....	49
4.3.1	PROGRAMAÇÃO COM BASE NO PROJETO TESTE 03 .....	53
<b>5</b>	<b>RESULTADOS E DISCUSSÕES .....</b>	<b>61</b>
5.1	ANÁLISE DA PROGRAMAÇÃO CONFORME CONCEITOS ESTIPULADOS	65
5.2	ANÁLISE DE RESULTADOS CONFORME CONCEITOS ESTIPULADOS ....	70
<b>6</b>	<b>CONCLUSÃO .....</b>	<b>72</b>
<b>7</b>	<b>RECOMENDAÇÕES DE TRABALHOS FUTUROS .....</b>	<b>73</b>
<b>8</b>	<b>REFERÊNCIAS.....</b>	<b>74</b>
<b>9</b>	<b>APÊNDICES.....</b>	<b>78</b>
9.1	APÊNDICE I (APÊNDICES A A W) – QR CODE DE ACESSO A UMA PASTA DO GOOGLE DRIVE PARA FACILITAÇÃO NA VISUALIZAÇÃO DE GRUPOS E NÓS UTILIZADOS .....	78
9.2	APÊNDICE II – QR CODE DE ACESSO A UMA PASTA DO GOOGLE DRIVE PARA VISUALIZAÇÃO DE VÍDEOS DE DESENVOLVIMENTO MANUAL E COM PROGRAMAÇÃO.....	78

## 1 INTRODUÇÃO

Diminuir o tempo em etapas de desenvolvimento e aumentar a eficácia é de interesse em qualquer indústria, algo que tem se tornado comum nas áreas da tecnologia e chegam a impactar na área de desenvolvimento de projetos de engenharia civil, trazendo assim novas maneiras de projetar, demandando mudanças na atitude como profissionais atuam e pensam enquanto projetam devido a essas novas tecnologias.

Conforme a utilização de tecnologias aplicadas no desenvolvimento de projetos apresentam processos mais eficazes, a utilização de computadores e softwares se tornaram ferramentas comuns no campo da engenharia civil, sendo umas dessas tecnologias o *design* generativo. *Design* generativo é uma tecnologia que através de softwares, possibilita explorar todas as diferentes possíveis soluções para se chegar a melhor opção de *design*, aprendendo com cada nova iteração para se chegar ao *design* mais efetivo (NAGY, 2017).

*Design* generativo possui então a capacidade de, a partir de um modelo inicial e informações alimentadas no software, gerar alterações no modelo através de repetições, criando assim um novo escopo com o intuito de ser melhor que o antigo a cada iteração.

A tecnologia trouxe então possibilidades de projetar de diferentes maneiras, no entanto, a simples utilização de tecnologia pode não garantir que o projeto atenda todas as necessidades e objetivos visados pelo projetista, sendo então interessante embasar como essa tecnologia atuará nesse desenvolvimento, surgindo a possibilidade de baseá-la em conceitos e filosofias, como por exemplo o *Lean Construction*.

Para Mossman (2018), uma definição simples seria que *Lean Construction* (construção enxuta) é uma coleção prática de teorias, princípios e axiomas, que abordam questões de reduzir/evitar desperdícios e utilizar as melhores práticas baseadas em teorias advindas de planejamento e estudos, juntando então técnicas e maneiras de pensar que podem auxiliar times e pessoas a se chegar em seu objetivo prevendo a melhor maneira de se fazer isso, trazendo toda uma análise dos impactos que o caminho tomado e todo o processo tem no resultado final.

Como indicado por NAGY (2017), o *design* generativo necessita de softwares e computadores e como indicado por Mossman (2018), *Lean Construction* pode ser descrito em teorias, princípios e axiomas. Entende-se então que existe a possibilidade

de inscrever tais informações, que são baseadas em princípios de evitar/eliminar desperdícios e otimizar o trabalho, dentro do software, gerando assim uma possível abordagem de BIM + *Lean* para se trabalhar com projetos. BIM (*Building Information Modeling* – Modelagem da Informações da Construção) é uma filosofia de trabalho que utiliza tecnologias e conceitos, onde de maneira visual é possível integrar as várias fases da vida de uma construção (desde o seu escopo inicial até caso venha a ocorrer sua demolição) através de informações abertas e disponíveis a todos os interessados (EASTMAN *et al*, 2014).

Esta monografia focou na apresentação e estudo da união de diferentes conceitos relacionados às áreas de engenharia civil através de tecnologia. Apresentou-se uma possibilidade de desenvolvimento de partes específicas de projetos, que não necessitam de desejo humano, através da união de BIM e *Lean Construction*, utilizando design generativo com o auxílio programação visual e escrita, buscando eliminar desperdícios.

Como exemplo ilustrativo, utilizou-se projetos elétricos de baixa tensão com o intuito de apenas auxiliar no desenvolvimento e conclusão, não sendo uma disciplina específica o foco desta monografia, utilizando para análises relacionado a desperdícios trazidos pelo *Lean* e comuns de serem observados.

Segundo Vasconcelos (2016), as principais causas de desperdício no canteiro de obras possuem influência direta desde o planejamento, detalhamento de projeto, especificações insuficientes, alterações de projeto, solicitações feitas por clientes e atrasos devido a revisões de projeto, sendo que estas atividades fazem parte das etapas iniciais da vida de um empreendimento. A autora aponta ainda que a grande maioria desses desperdícios poderiam ser minimizados com a utilização de novas tecnologias e conceitos vindos do BIM já na fase de projeto.

Porém projetos desenvolvidos utilizando apenas a ideia genérica, porém necessária do que é BIM (representação tridimensional do projeto), sem o conhecimento e significado do que é a informatização necessária, pode acarretar em erros devido à modelagem tridimensional realizada de maneira incorreta. Segundo Pereira *et al* (2015), embora modelos desenvolvidos de forma tridimensional possam facilitar a visualização do projeto, a modelagem incorreta de elementos juntamente com a falta de verificação das informações desenvolvidas pode acarretar prejuízos para a execução.

Projetos mal elaborados mesmo que viáveis de serem desenvolvidos corroboram com o aumento de desperdícios, demonstrando assim possibilidade de melhoramento. Segundo Brandstetter e Ribeiro (2019), fiscais e projetistas são responsáveis por 50% do custo total de aditivos (custos não previstos no orçamento inicial) destinados a alterações na construção de obras públicas, sendo as partes de elétrica e estrutura as mais impactadas, tendo a categoria de projeto (entre mão de obra, falta de materiais, entre outros) a que possui maior risco de causar custos extras à obra devido a alterações, adaptações ou reformulações.

Demonstra-se então que mesmo com a utilização de BIM, projetos ainda possuem risco de apresentarem problemas, os quais são comuns a qualquer modelo de desenvolvimento. Apresentou-se então o interesse em otimizar e melhorar processos já nas etapas iniciais de um empreendimento, unindo por exemplo BIM com ferramentas (Dynamo®) e filosofias (*Lean Construction*), visando a diminuição/eliminação de desperdícios.

Espera-se então que, aplicando conceitos advindos do *Lean Construction* e utilizando *design* generativo com o suporte de um software que facilita a utilização de BIM (Revit®), espera-se apresentar uma possível maneira de projetar utilizando programação, demonstrando a possibilidade de eliminar/reduzir desperdícios e otimizar o processo de desenvolvimento, principalmente em partes de detalhamentos e indicações de informações gerais.

Para análise espera-se que os novos modelos gerados a partir de um modelo inicial apresentem reduções e/ou eliminações de desperdícios, como diminuição de materiais que poderiam apresentar um layout diferente para se obter resultados similares, e desperdícios mais complexos de serem analisados, como desperdício intelectual relacionado a informatização e organização de projetos. Essas análises poderiam ser complicadas de serem percebidas e acarretariam em desperdício de tempo para serem resolvidos por projetistas.

Embora os modelos possam apresentar reduções, espera-se que a qualidade das partes dos projetos analisados seja mantida ou melhorada conforme desejo do projetista e cliente.

## 1.1 JUSTIFICATIVA

Segundo Lean Institute Brasil (2007), existem diversos desperdícios que ocorrem em escritórios envolvendo a geração de informações. São citados, por exemplo: os recursos perdidos devido a qualquer tipo de espera, um desperdício de processamento devido a um trabalho não executado da melhor forma, a energia gasta por causa de um trabalho não ter sido feito da melhor forma possível devido à inexistência de padronização, checagens desnecessárias devido a inspeções e retrabalhos, erros gerais e falta de informações.

Desperdícios podem então ocorrer já nas etapas iniciais de desenvolvimento de projetos, as quais podem causar problemas em escritórios de arquitetura e engenharia e eventualmente acarretar problemas durante a execução de obras, gerando prejuízos desnecessários que poderiam ser evitados com processos mais eficientes e diferentes maneiras de pensar o desenvolvimento do projeto.

Segundo Mayr (2000), durante a revisão de projetos e após sua conclusão, o projetista pode identificar maneiras mais eficientes ou econômicas de se realizar a mesma tarefa, tal identificação pode acarretar em mudanças no projeto o que pode levar a um retrabalho (desperdício de tempo), o qual poderia ser evitado. Caso o projetista opte por manter o projeto como está, mesmo sabendo que existe maneiras mais eficientes de ser realizado, o projeto pode então apresentar um desperdício de material desnecessário.

Segundo Harding e Shepherd (2016), softwares que permitem a criação de modelos generativos tais como Dynamo<sup>®</sup> (Autodesk<sup>®</sup>) e Grasshopper<sup>®</sup> (Google<sup>®</sup>), permitem o desenvolvimento de ideias complexas de maneiras mais efetivas que desenhos a mão, maquetes e CAD. Porém, tais softwares apresentam dificuldades para trabalhar em casos de *designs* topologicamente inflexíveis (situações não repetitivas e que apresentam variabilidade), mas possibilitam a inclusão de um maior número de definições paramétricas (inserção de informações) já em fase de projeto.

Muitos desses desperdícios são causados devido a rotina de tarefas iguais/similares, as quais possuem a possibilidade de serem automatizadas devido a sua característica repetitiva.

Pelo exposto neste tópico infere-se a necessidade da apresentação de alternativas para otimizar e melhorar processos e etapas no desenvolvimento de projetos, tendo tecnologias como Dynamo<sup>®</sup> uma possível ferramenta. Justifica-se então o desenvolvimento desta monografia baseando-se no interesse da exploração

de tecnologias (Dynamo®) que permitem a utilização de programação para o desenvolvimento de etapas de projetos (focando nas que possuem processos repetitivos que independem muitas vezes de desejo humano), visando eliminar/otimizar desperdícios no desenvolvimento de projetos/escritórios.

## 1.2 OBJETIVOS

### 1.2.1 Objetivo geral

Analisar como projetos relacionados a engenharia civil, criados com a utilização de *design* generativo através de programação visual em Autodesk® Revit® e Dynamo® a partir de conceitos de BIM e *Lean Construction* podem apontar reduções/eliminação de desperdícios.

### 1.2.2 Objetivos Específicos

- Desenvolver modelagem bidimensional e tridimensional utilizando programação visual;
- Desenvolver informações (documentação) bidimensional e tridimensional utilizando programação visual;
- Organizar informações (documentação) bidimensional e tridimensional utilizando programação visual;
- Analisar a compatibilidade entre BIM e *Lean Construction* através da programação desenvolvida.

## 2 REVISÃO BIBLIOGRÁFICA

A revisão bibliográfica focou em apresentar os temas mais importantes para o desenvolvimento desta monografia, trazendo informações das tecnologias, conceitos e filosofia utilizados.

Neste tópico são descritos os conceitos os conceitos trazidos no título desta monografia e complementou-se com outros que farão parte do desenvolvimento, como: BIM, *Lean Construciton*, programação visual, Revit® e Dynamo®. Realizou-se a relação entre as tecnologias e como uma interdepende da outra.

### 2.1 BIM (*BUILDING INFORMATION MODELING*)

Segundo Eastman *et al* (2014), *Building Information Modeling* (BIM) possui seus primeiros conceitos e terminologias conhecidas datadas do ano de 1975, trazidas pelo autor Charles M. Eastman no trabalho “*Building Description System*”, onde apresentou-se uma introdução do que se tornaria BIM. Nos anos 80 abordagens como “*Building Product Models*” e “*Product Information Models*” surgiram em países diferentes, onde logo se uniriam para formar “*Building Information Model*”, termo conhecido no mercado. A primeira utilização do termo *Building Modeling* apareceu em 1986 em um artigo de Aish, R., onde foi apresentado tecnologias de modelagem 3D, componentes paramétricos, bancos de dados, e outros conceitos que fariam parte do que se entende por BIM. Em 1992 foi cunhado o termo “*Building Information Modeling*”, pelos autores van Nederveen e F. Tolman, no trabalho *Automation in Construction*, tendo o termo BIM obtido diversas definições conforme seu desenvolvimento nos anos consequentes.

De acordo com UK BIM FRAMEWORK (2022), de modo simplificado, o processo para se trabalhar com BIM pode ser dividido em 4: *Design* (Projeto) - criação de estruturas e outros através de boas práticas garantindo a qualidade e gerenciamento de informações; *Build* (Construir) – fazer com que a construção utilize novas tecnologias digitais, tornando as informações compartilháveis e compartilhadas através de todos os interessados no produto, realizando um trabalho com mais qualidade, seguro e produtivo durante a construção; *Operate* (Operar) – sempre utilizar informações atualizadas com o intuito de evitar problemas; *Integrate* (Integrar) – projetar entendendo como aquela construção pode impactar na qualidade de vida das pessoas, fornecendo informações para o *design* com o intuito de se obter um melhor resultado para a infraestrutura e sociedade.

Segundo ISO 19650 (2020), processos BIM possuem documentos de normalização, os quais apresentam noções de trabalhar com BIM e definições de conceitos importantes.

Para os aspectos mais relevantes desta monografia, normas<sup>1</sup> que tratam especificamente sobre BIM não se mostraram uma prioridade, pois não abordam (exceto a própria questão de BIM) como trabalhar com a união dos diferentes conceitos e tecnologias apresentados na introdução, sendo importantes para apresentação de definições oficiais dos conceitos utilizados, assim como trazido pela ISO 19650-1 (2018), que define *Building Information Modeling* como sendo uma maneira de trabalhar através do uso de tecnologia adequada, obtendo todos os benefícios do BIM utilizando a quantidade correta de informações digitais compartilhadas em projetos, para se chegar a melhor decisão relacionado a construções e operações. Entende-se então das palavras *Building* (Construção), *Information* (Informação) e *Modeling* (Modelagem), a realização da modelagem de um projeto, com informações suficientes as quais serão utilizadas em construção para se obter um produto idêntico àquele projetado, sempre objetivando ter um processo completo (do início ao fim do ciclo de vida de um empreendimento) de qualidade.

É comum a utilização do termo “software BIM” relacionando com a noção de tecnologia, ou profissionais que acreditam estarem utilizando BIM por apenas projetarem utilizando certo software, no entanto a simples noção de “software BIM” não possui sentido no momento que se entende o significado de cada elemento no termo BIM. Segundo Autodesk (2022), enquanto BIM é um processo, uma metodologia de trabalho focada em trazer melhores resultados para produtos/projetos da indústria AEC (Arquitetura, Engenharia e Construção), softwares como por exemplo Revit®, possuem a capacidade de facilitar a utilização de BIM através de sua capacidade em carregar, armazenar e compartilhar (tornar visível) informações.

Um elemento, como por exemplo, uma parede, pode ser modelada tridimensionalmente em Revit®, porém se naquela parede não for inserido a quantidade de informações necessária que possibilitará a execução da parede em obra possuir as mesmas informações presentes em projeto, não se está utilizando BIM, mas apenas realizando modelagem tridimensional.

---

<sup>1</sup> Cita-se as normas de maior relevância: ISO 19650 (2018), parte 1 a 5 e ABNT 15965 (2011), parte 1 a 7.

Segundo Eastman *et al* (2014), as seguintes categorias de modelos por si só não podem ser identificadas como BIM:

- Modelagem puramente tridimensional (sem ser informatizada);
- Modelos separados de CAD 2D que se unem para compor um projeto;
- Modelos virtuais que são modificados em uma vista mas que não sofrem alteração em outra vista;
- Modelos sem suporte ou comportamento parecidos com o que se espera da edificação real.

Então, o nível e a maneira em que a modelagem é desenvolvida importa para se trabalhar com BIM, existindo diferentes níveis de desenvolvimento de modelagem, como representados na figura 1.

Figura 01 – LOD (*Level of Development*) de projetos



Fonte: NATSPEC BIM (2013). Adaptado.

LOD é uma definição/classificação criada pelo órgão AIA (*American Institute of Architects*) entre os anos 2011-2013, desenvolvido com o intuito de ser utilizado no documento “AIA G202-2013 Building Information Modeling Protocol Form [1]”, com o interesse em tornar mais claro e confiável o conteúdo BIM aplicado nas diferentes etapas de um projeto para membros da indústria AEC. O LOD do projeto facilita a comunicação e permite com que projetistas definam até que ponto as informações de seu projeto são precisas/realistas com base em seu LOD (diferente de nível de detalhe, que indica a quantidade de detalhes inserido no projeto), possibilitando o melhoramento na representação e inclusão de informações em modelos. As informações e evolução do conceito de LOD são atualizadas anualmente pelo órgão FORUM BIM através do guia LOD (BIM FORUM, 2021).

Segundo BIM FORUM (2021), existem 6 níveis padrões/mais comuns de LOD, os quais a utilização dependerá da importância para o necessário/exigido pelo modelo:

- LOD 100: representação puramente visual a qual não possui informações geométricas realistas ou identificáveis (por exemplo, utilização de cubos para simulação de objetos reais). Não possui precisão em sua localização, forma e tamanho, sendo todo elemento de LOD 100 considerado impreciso.
- LOD 200: representações possíveis de serem identificadas conforme sua utilização, tendo informações geométricas identificáveis, porém possuindo precisão aproximada e não realista.
- LOD 300: representações realistas dos elementos em projeto, os quais possuem medidas, quantidade, localização e informações precisas, podendo o modelo ser utilizado como base para desenvolvimento do projetado (informações podem ser retiradas diretamente do elemento criado através da modelagem, sem necessitar de indicações, como anotações).
- LOD 350: elementos de LOD 350 possuem todas as informações de elementos de LOD 300, com a diferença em que o elemento é modelado prevendo também sua interação com outros elementos/disciplinas de projetos.
- LOD 400: possuindo todas as informações advindas do LOD 350, neste nível o desenvolvimento chegou a tal ponto que torna possível utilizar o modelo para fabricação e execução exata, tornando o projetado uma imagem da realidade.
- LOD 500: relacionado a verificação de informações pós-execução (*as-built*), não sendo uma indicação de progressão de LOD, mas sim uma verificação do produto.

Trabalhar com diferentes LOD causa então uma variação na precisão de informações obtidas em projeto, tornando o projetado mais próximo do executável desejado conforme o LOD aumenta, facilitando/possibilitando o trabalho com diferentes dimensões BIM.

Para Hamil (2021), dimensões BIM são a evolução de modelos de projetos bidimensional e tridimensional (referindo-se a modelagem), em que a cada dimensão é adicionado informações para se obter um resultado específico, existindo 5

dimensões bem definidas (BIM 1D, 2D, 3D, 4D e 5D), as quais possuem definições e informações concretas trazidas como por exemplo pela ISO 1965. Na comunidade científica já existem trabalhos e desenvolvimentos de outras dimensões emergentes, indicadas como nD<sup>2</sup>, sendo estas principalmente BIM 6D, 7D, 8D, 9D e 10D, mas que, no entanto, ainda não possuem documentos que trazem definições e parâmetros como normalizações ISO até a data de desenvolvimento desta monografia.

A aplicação de BIM em projetos apresenta então benefícios em diversos momentos de um empreendimento, já iniciando em projeto, como apresentado por Eastman et al (2014), o uso de BIM no processo de elaboração de um projeto pode ser dividido em quatro pontos de vistas, sendo que cada ponto apresenta benefícios para o empreendimento:

- Anteprojeto: a facilidade e velocidade com que modelos BIM podem trazer feedbacks sobre o plano inicial auxiliam na tomada de decisão de um *design* de modo a fortalecer a qualidade do produto no decorrer de seu desenvolvimento, impactando em seu produto final.
- Projeto e análise: possibilidade de avaliação de vários aspectos funcionais que impactarão na edificação real, como: integridade estrutural, controle de temperatura, iluminação, ventilação, entre outros. Sendo um ponto que é potencializado com a colaboração de diferentes profissionais, todos trabalhando com BIM.
- Informatização da construção: o BIM possibilita não só tornar visual, mas também acelerar o desenvolvimento de informação e documentação, um ponto que é potencializado pela utilização de softwares modernos que permitem e são criados pensando em se trabalhar com BIM.
- Integração de projeto e construção: refere-se à utilização do modelo virtual (projeto) diretamente para a execução da edificação, possibilitando implementar e realizar testes no modelo enquanto ocorre a execução da obra, dando suporte a produção (desenvolvimento).

O trabalho com BIM ainda possui a capacidade de adequar outras filosofias e modos de trabalhar e pensar o projeto, como por exemplo *Lean Construction*, não se

---

<sup>2</sup> Como não é de interesse a esta monografia, não será feito uma discussão e levantado informações sobre as várias dimensões BIM.

restringindo então a conceitos e formas padrões. Como previsto por Eastman *et al* (2014), BIM e *Lean Construction* são conceitos que se complementam de diferentes maneiras e que possuem capacidade de progredirem juntos. O conceito *Lean* aplicado ao desenvolvimento de projetos age de maneira a eliminar etapas desnecessárias no processo de desenvolvimento, eliminar etapas que não produzem valor ao cliente, eliminar erros, retrabalhos e reduzir ciclos, sendo que BIM viabiliza e potencializa todos esses objetivos. Aponta-se ainda a necessidade da redução de tempo, sendo que a utilização de BIM pode exercer um importante papel para diminuir o tempo do ciclo de produção, gerando resultados similares e/ou melhores.

## 2.2 LEAN CONSTRUCTION

Segundo *Lean Enterprise Institute* (2022), interesses no melhoramento da produção na indústria surgiram por volta de 1913 com ideias de Henry Ford sobre produção em massa, no entanto esse sistema não possibilitava inovações e variedade nos produtos. Iniciando por volta de 1930 e tendo continuidade no decorrer dos anos, Kiichiro Toyota e Taiichi Ohno observaram nas ideias de Henry Ford a possibilidade de inovar e melhorar o processo nas fábricas Toyota com pequenas mudanças, criando assim o sistema de produção Toyota (Toyotismo), que sempre busca evitar desperdícios e melhorar a produção. Vindo da ideia do Toyotismo a noção de *lean* desenvolveu-se por volta dos anos 90, tendo se espalhado devido a trabalhos como “The Machine That Changed the World (1990) by James P. Womack, Daniel Roos, e Daniel T”.

Koskela (1992), em seus estudos pioneiros concluiu que a filosofia trazida pelo *Lean Production* aplicada a produção em indústria poderia ser replicada na construção civil, implementando suas maneiras de trabalhar, pensar e princípios, porém com a necessidade de transformação nas práticas e teorias construtivas de engenharia e gestão das áreas de construção. Trouxe-se então uma possível ideia para combater desperdícios no âmbito da construção, espelhando-se nos sucessos obtidos na área de fabricação em indústrias.

Segundo *Lean Enterprise Institute* (2022), *Lean* é uma maneira de pensar a produção/construção (*production/construction*) gerando o maior valor possível, com a menor quantidade de recursos necessária, sempre evitando desperdícios. Em práticas *Lean* nunca se deixa de realizar experimentações, adaptações e melhorias para a

maneira de se criar, desenvolver e trabalhar, sempre almejando desperdício nulo através da união do planejamento e prática.

Segundo *Lean Construction Institute* (2022), os 7 desperdícios mais comuns nos processos de manufatura em fábricas foram identificados devido ao *Lean Production*, no entanto podem ser aplicadas a qualquer processo, sendo esses desperdícios:

- *transportation* (transporte - qualquer tipo de movimentação desnecessária no processo de fabricação, de pessoas, máquinas ou processos administrativos).
- *inventory* (inventário - qualquer produto, desde materiais brutos, a prontos para venda que excedam a demanda)
- *motion* (movimento - movimentações que não trazem valor para o processo).
- *waiting* (espera - tempo desperdiçado por uma parte posterior que necessita do término de uma etapa anterior).
- *over processing* (processamento excessivo – processos, itens, elementos, entre outros a mais no produto os quais o cliente não necessita/não agregam valor).
- *over production* (produção excessiva - produzir mais do que a demanda, gerando estoque)
- *defects* (defeitos - produto que necessita de retrabalho para se chegar onde se desejava inicialmente)

Além desses 7 desperdícios comuns, *Lean Construction Institute* (2022), ainda identifica 4 outros recorrentes:

- *underutilized talent* (talento subutilizado - utilizar de maneiras ineficiente a mão de obra e recursos, entendido também como desperdício intelectual quando se refere a mão de obra especializada).
- *over burdening* (sobrecarga - demanda excessiva de pessoas ou máquinas que leva a problemas de qualidade e defeitos)
- *unevenness* (desigualdade - faz com que o processo de produção seja desigual, eliminando possibilidades de desenvolvimento e melhoramento que outros participantes poderiam apresentar)
- *waste is disrespect* (desperdício é desrespeito - refere-se a qualquer tipo de desperdício, desde os listados até outros possíveis que possam ser identificados no processo de criação).

### 2.3 BIM E *LEAN CONSTRUCTION* (TEORIA DE BIM 9D)

BIM 9D é uma dimensão BIM em surgimento que apresenta poucos estudos na área, no entanto já possui definições não oficiais e abordagens relacionadas ao tema. Segundo Arnal (2018), BIM 9D é uma das dimensões que faz parte da teoria das 10 dimensões BIM (teoria que objetiva chegar à dimensão BIM 10D relacionada a construção industrializada), e pode ser entendido como a introdução das filosofias relacionadas ao *Lean Construction* no setor de construção através da utilização de BIM.

Segundo Montoya *et al* (2019), a utilização de BIM juntamente com *Lean Construction* permite a melhoria durante toda a vida de uma construção, iniciando já nos primeiros designs de projeto, detalhes construtivos, construção, entrega e manutenção da edificação após concluída. Isso impacta em todo o ciclo de vida de um empreendimento através de considerável redução de desperdícios e melhorias, evitando problemas como mudanças de design e falta de informações em projeto. Os autores ainda indicam a falta de informações e exploração nas áreas que relacionam *Lean* e BIM trabalhando juntos.

Segundo estudos de Tavares (2020), notou-se que aplicações de *Lean Construction* com a utilização de ferramentas, práticas e metodologias que se enquadram em projetos, trazem uma maturidade e maior eficiência ao produto, sendo o BIM um desses elementos (se referindo a ferramentas como o próprio Revit®, metodologias e filosofias como *BIM execution plan* entre outros).

Mariz e Picchi (2021), identificaram em suas pesquisas que aplicações de BIM e *Lean Construction* de maneira unida ainda são escassas e pouco utilizadas, e as aplicações encontradas são pouca exploradas, indicando conclusões similares das obtidas nas pesquisas de Montoya *et al* (2019). Os autores, assim como Tavares (2020), ainda identificaram que tecnologias auxiliares como kanban, softwares, metodologias ágeis, entre outros, podem melhorar a aplicação das práticas *Lean*.

Observa-se então que já existem estudos relacionando BIM e *Lean Construction*, e embora BIM 9D não seja um termo oficial, é uma teoria que aborda uma temática similar.

## 2.4 REVIT®

Revit® é um software lançado oficialmente no mercado no ano 2000 pela empresa Revit Design Corporation que foi fundada em 1997, sendo o primeiro modelador paramétrico para construção desenvolvido com o foco na indústria AEC. No ano de 2002 a empresa Autodesk, Inc. incorporou a empresa Revit Design Corporation, expandindo seu acervo de softwares (BUILDINGS, 2002).

Segundo Autodesk (2022), Revit® é um software disponibilizado pela empresa Autodesk® em versões pagas, versão gratuita teste temporária e versão gratuita para estudantes. Utilizado nas áreas de arquitetura, engenharia, urbanismo e design, tem como foco atender as áreas de projetos e integrar a indústria AEC (Arquitetura, Engenharia e Construção) para colaboração na criação e construção.

Segundo Autodesk (2021), Revit® possui quatro principais conceitos característicos do software, sendo eles:

- **Modelagem:** a modelagem/criação no software ocorre de maneira que a modificação em uma vista do projeto (seja 2D ou 3D) altera automaticamente as informações em todas as vistas existentes (plantas, cortes, tabelas, entre outros), sendo então todas as vistas dentro de um mesmo arquivo apenas um modelo de construção virtual, em vez de várias informações do mesmo projeto que independem da modificação de uma à outra.
- **Elementos:** são os objetos tridimensionais, onde representam componentes reais (que serão criados/executados na realidade). Elementos podem ser paredes, janelas, vigas, sendo que qualquer elemento que tenha representação tridimensional pode ser considerado como um elemento que existe/existirá na realidade (diferindo de cotas e anotações por exemplo).
- **Parâmetros:** são informações que definem tamanho, forma, posição, material e informações diversas de um elemento. Muitos parâmetros se relacionam-se entre si de forma automática dentro do software (quando uma porta é colocada em uma parede por exemplo, é criada uma cota automática que fica visível ao se clicar no elemento porta, referindo-se à porta a uma divisória adjacente da parede), sendo essa relação identificada como modelagem paramétrica.
- **Famílias:** são um conjunto de elementos parametrizados gerais que são importados dentro de um projeto, como paredes, janelas, vigas, símbolos, objetos de anotações, entre outros, sendo qualquer elemento adicionado no

projeto (não modelado no próprio modelo hospedeiro). Uma mesma família pode possuir vários elementos, uma família de mesas por exemplo, pode possuir mesas de diferentes tamanhos, cores, materiais, parâmetros gerais diferentes, entre outros.

O Revit® ainda permite a aplicação de conceitos e sistema de desenvolvimento/execução advindas do BIM devido ao armazenamento e compartilhamento de informações que o software possibilita (através de elementos, parâmetros e famílias), permitindo um maior controle do processo de desenvolvimento de projeto e execução com base nas dimensões BIM definidas e aplicadas.

O software em si é focado em atender a indústria AEC, que em sua maioria foca no desenvolvimento de projetos de arquitetura, engenharia, estrutura e complementares, sendo muitas vezes os projetos complementares dependentes da arquitetura. Segundo Autodesk (2021), projetos MEP (*Mechanical, Electrical, Plumbing* ou Mecânico, Elétrico, Tubulação), são normalmente vinculados a modelos de arquitetura (facilitação na visualização e criação do projeto) e permitem o desenvolvimento e gerenciamento completo de um projeto MEP, tudo dentro de um mesmo arquivo Revit®.

## 2.5 DYNAMO®

Segundo Jezyk (2019), Dynamo® é uma ferramenta de programação visual, não necessita obrigatoriamente de conhecimentos de linguagens de programação escrita, onde é necessário apenas a realização da interligação de informações para sua utilização. Possui a característica de ser *open source* (gratuito e acessível ao público para visualização e modificações de programas compartilhados), e está inserido como uma ferramenta dentro do Revit® a partir das versões de 2020, possibilitando boa interoperabilidade<sup>3</sup> entre ambos. O Dynamo® através de linguagem de programação consegue utilizar comandos gerais do Revit® e ações como criar paredes, inserir/deletar elementos, entre outros, tendo acesso a todos os elementos, comandos e informações presentes no Revit®. Dynamo® consegue trabalhar com o Revit API (API – *Application Programming Interface* ou Interface de Programação de Aplicações, tendo acesso a qualquer livreria externa e produtos da Autodesk®), esse

---

<sup>3</sup> Interoperabilidade de maneira simplificada, é a capacidade de sistemas (nesse caso Revit® e Dynamo®) de trocarem informações de forma eficaz sem elas sofrerem modificações no processo de transferência.

API faz uma conexão entre o Dynamo® e o Revit®, onde permite-se criar rotinas e programas que possibilitam desenvolver uma ou várias atividades em sequência (automação) dentro do Revit®.

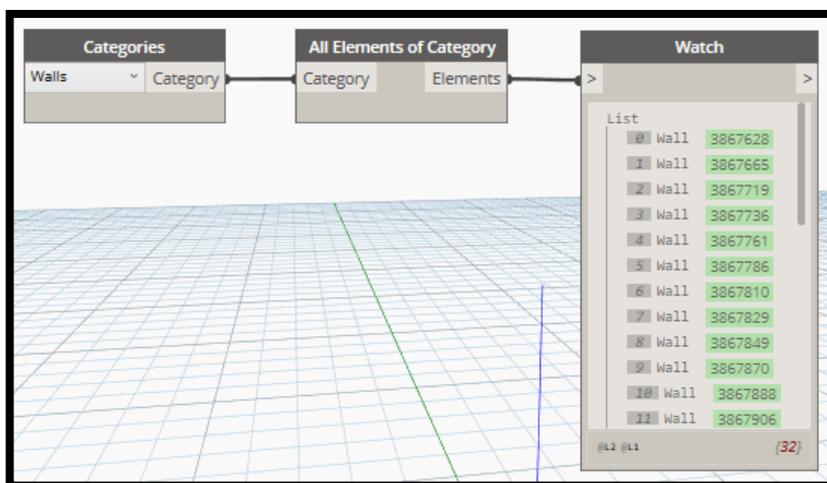
As principais funções do Dynamo® são: gerenciar dados (extrair, processar, exportar e adaptar informações), automatizar atividades repetitivas (através de uma rotina o Dynamo® faz com que o Revit® execute uma série de comandos repetitivos) e criação de geometrias complexas parametrizadas.

### 2.5.1 Programação visual e programação escrita em Dynamo®

Segundo Jezyk (2019), a programação visual em Dynamo® é composta por nós, que são pontos em formato normalmente retangular que possuem entrada(s) e saída(s), que carregam informações específicas dentro de si e que são únicas de cada nó. A conexão entre nós ocorre através de cordas, que possuem a função de transferir informações de um nó a outro através de entradas e saídas, tais conexões permitem a realização de ações específicas com base nas informações que cada nó carrega. Dependendo do nó, ele pode ter várias entradas e saídas, porém em cada nó é possível inserir apenas uma corda por porta de entrada.

A programação visual, representada na Figura 02, dentro do visualizador do Dynamo® foi desenvolvida usando como base um arquivo rvt. que possui diversas paredes (do mesmo e de diferentes tipos, como: alvenaria, *Light Steel Framing*, entre outros).

Figura 02 – exemplo de nós de Dynamo®



Fonte: O autor (2022).

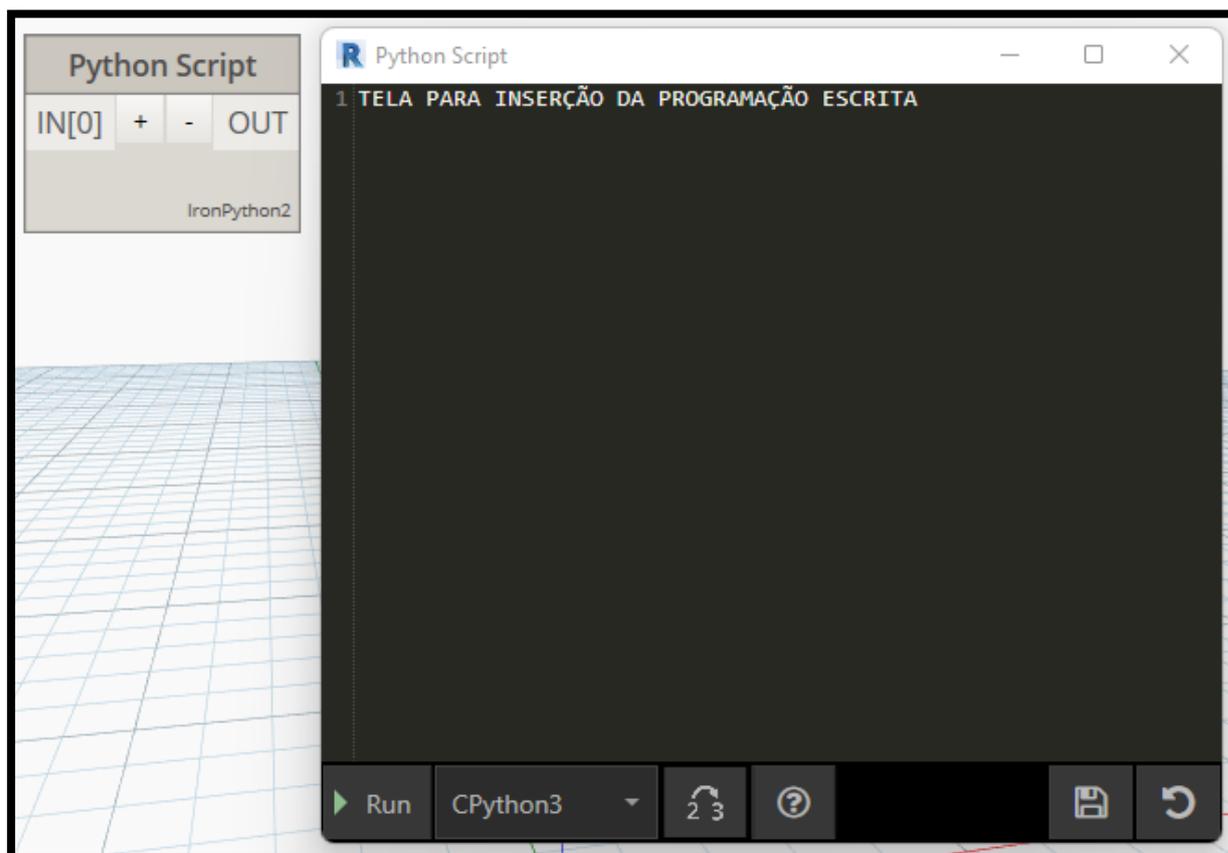
Como apresentado no exemplo da Figura 02 o nó “*Categories*” permite selecionar uma categoria de elementos (no exemplo está sendo selecionado a categoria *Walls* ou Paredes), tendo como saída a categoria selecionada, com sua corda de saída conectada com a entrada do nó “*All Elements of Category*”. O nó “*All Elements of Category*” recebe como entrada (input) uma categoria de elementos e fornece como saída (output) os elementos daquela categoria existente no projeto (no caso paredes) os dispendo em uma lista.

O nó “*Watch*” é apenas um nó para facilitar a visualização das informações de input que recebe. Observa-se que no nó “*Watch*” cada parede possui uma identificação (ID), no caso da parede de index (sua localização na lista) 0, sua ID é 3867628, sendo que cada elemento (independente da categoria) possui uma ID única automaticamente criada pelo Revit®, tornando cada elemento único. Os 3 nós apresentados na Figura 02 são disponibilizados dentro do Dynamo® e não permitem alteração de sua função (modificação em sua programação escrita).

Os nós possuem nomes de *default* (padrões do software ou desenvolvedor), no entanto podem ser modificados para facilitar a visualização e entendimento da programação (a alteração do nome do nó não altera sua função e é puramente textual). No caso da Figura 02 o nome do nó “*Categories*” poderia ser alterado para sua versão em português “*Categorias*”, ou qualquer nome desejado, e isso não alteraria sua função.

Todos os nós, mesmo os já prontos para utilização dentro do programa e em pacotes, são criados utilizando programação escrita, porém o Dynamo® também permite a criação de nós de IronPython, onde pode-se criar um novo nó do zero inserindo as informações que desejar com programação escrita. Nós disponibilizados pelo programa não permitem a alteração de sua programação escrita, diferente de nós disponibilizados em pacotes criados por terceiros (característica *open source*) que são possíveis de serem criados com nós de IronPython, como representado na Figura 03.

Figura 03 – Exemplo de nó de IronPython



Fonte: O autor (2022).

No caso da Figura 03 o nó “Python Script” possui funcionalidade similar a dos nós apresentados na Figura 02 (possui inputs, outputs e suas informações podem ser passadas para outros nós através de cordas), com a diferença de que é possível aumentar seus inputs ou simplesmente eliminar o(s) input utilizando as indicações de “+” e “-” representadas no nó, possibilitando boa adequação dependendo da funcionalidade programada.

Nós de IronPython possuem uma tela (com nome também de Python Script) permitindo desenvolver o nó com a função desejada pelo projetista/programador utilizando programação escrita. No caso do nó representado na Figura 03, devido a ele não possuir uma programação escrita, faz com que o nó não possua nenhuma função. Todos os nós presentes no software são desenvolvidos com IronPython, sendo algo global a toda a programação visual, sendo então, a conexão de nós através da programação visual a conexão de nós de IronPython.

Segundo .NET Foundation (2021), IronPython (utilizado principalmente para aplicações e automações) é uma linguagem de programação *open source* criada pela empresa Microsoft® e uma implementação da linguagem Python desenvolvida com a linguagem de programação C#. A linguagem IronPython possibilita o acesso e utilização das plataformas .NET (plataforma de desenvolvimento e execução de aplicações da Microsoft®), às bibliotecas Python e a realização da conexão com o Revit API através do comando *clr (Common Language Runtime)*, normalmente sendo a primeira linha de código em nós de IronPython quando necessário.

Devido a características da linguagem, não é possível trabalhar com plataformas .NET (como o Revit API), sendo o IronPython uma solução para a utilização de Python juntamente com uma plataforma .NET.

Segundo IBM® (2022), uma API (*Application Interface Programming*) permite a comunicação entre computadores e aplicativos/usuários e elementos digitais e softwares, sendo responsável por enviar pedidos de ações entre provedores e usuários e devolver os resultados, cada ação em uma plataforma/software passa por esses pedidos, sendo que usuários simples não possuem acesso a todas as funcionalidades de uma API (pois poderia modificar a plataforma/software de todas as maneiras). Revit API docs é então a biblioteca (docs de *documentation*) das informações utilizadas para criação do software, trazendo as informações de todos os pedidos entre o software Revit® e o projetista. Essa API permite com que a Autodesk® compartilhe as funcionalidades e informações de seus programas com terceiros (de maneira lógica, não fornecendo informações/acesso para modificações e replicação do software).

Segundo Autodesk (2022), as informações disponibilizadas através do Revit API docs, possibilitam a criação de programas utilizando linguagens compatíveis com .NET dentro do Revit®, como: VB.NET, C# e C++/CLI. Segundo Talarico (2022), é possível fazer com que a maioria dos códigos trazidos no Revit API docs sejam traduzidos de C# (linguagem em que é apresentado o Revit API docs) para Python realizando pequenos ajustes.

A programação escrita permite a realização de ações através de comandos específicos escritos, e como no Dynamo® é possível criar nós de IronPython®, os quais permitem o desenvolvimento de programação escrita, permitindo assim a criação de nós com informações para a realização de uma ação específica conforme o desejado,

tendo estes nós o funcionamento igual a nós advindos do software (ligados por cordas a outros nós com entrada(s) e saída(s)).

## 2.6 DESIGN GENERATIVO UTILIZANDO REVIT® E DYNAMO®

Segundo Autodesk (2021), *design* generativo é um processo específico realizado com a utilização de computadores através de algoritmos, onde o *designer* (projetista) define parâmetros que quer obter com o *design*, e com base neles o computador (utilizando um software) cria o produto esperado juntamente com alternativas, baseando-se no produto anterior e feedbacks/modificações nos parâmetros por parte do projetista. Para Revit® e Dynamo®, o design generativo pode ser criado através da programação visual em Dynamo® com a utilização de nós, incluindo nós de IronPython, onde a execução da programação dentro do Dynamo® gerará modificações dentro do Revit®, criando uma modelagem com base na programação.

Revit® permite então através da utilização de programação visual específica em Dynamo®, baseado nos parâmetros criados pelo projetista, a utilização de um processo de *design* generativo.

Uma ferramenta específica<sup>4</sup> para *design* generativo dentro do Revit® foi implementada pela Autodesk® a partir da versão de Revit® 2021, possibilitando a criação de estudos de *design* generativo sem a necessidade de utilizar Dynamo®, porém mais limitado devido a liberdade que a programação em Dynamo® permite.

---

<sup>4</sup> Não se utilizou tal ferramenta nesta monografia devido a ser recente e não apresentar o necessário para o desenvolvimento desejado. Utilizou-se apenas Dynamo®.

### 3 METODOLOGIA

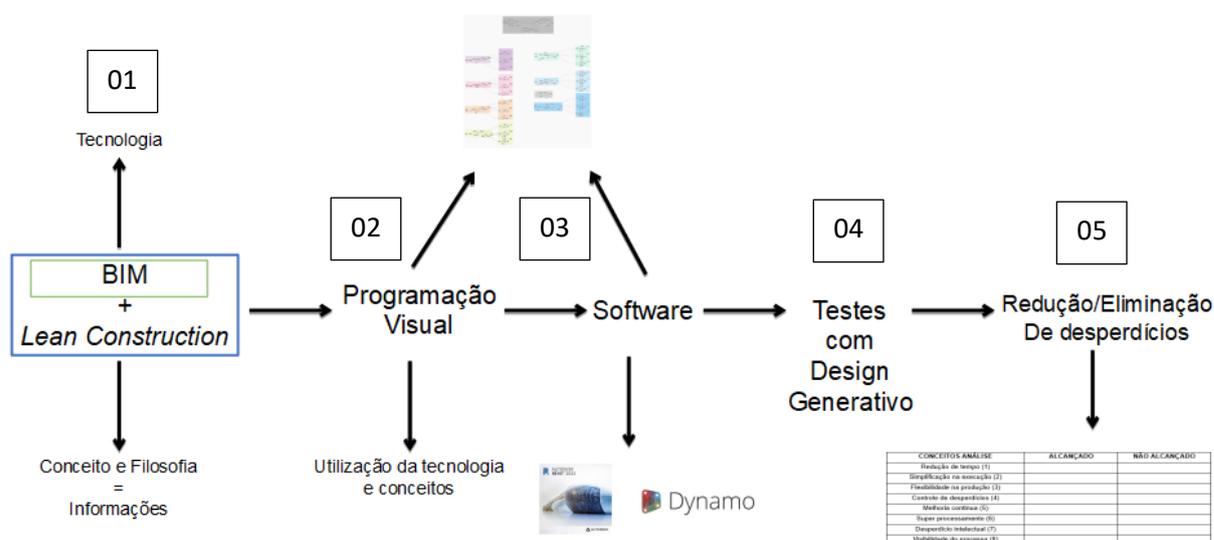
O desenvolvimento iniciou com a criação de modelos elétricos simplificados de baixa tensão (com intuito de testes) em arquivos rvt. Como o foco desta monografia é o estudo e análise das possibilidades da tecnologia, conceitos e filosofias, a escolha da utilização de projetos elétricos se deu devido a possibilidade de trabalhar com localizações geométricas e diferentes elementos tridimensionais de cada ponto em um arquivo dyn.

Dentro de um mesmo arquivo rvt. desenvolveu-se a programação visual em Dynamo® (é necessário abrir arquivos dyn. em arquivos de formato Revit®. como rvt. e rfa.). Para análise dos resultados observou-se se a utilização da programação acarretou na eliminação/diminuição de desperdícios baseadas em conceitos do *Lean Construction* que estão descritos nos tópicos 3.7 e 3.8.

Como o intuito foi que a mesma programação sirva para ser aplicado em diferentes arquivos/projetos rvt., realizou-se iterações em modelos com diferentes layouts e elementos elétricos com melhoramento contínuos da programação conforme necessidade de cada diferente layout com o interesse de realizar a menor quantidade possível de modificações no programa a cada nova iteração quando utilizado em projetos similares (se referindo a área, quantia de elementos e tipo de elementos de projeto).

Para o desenvolvimento de tal programação, seguiu-se o fluxograma de trabalho apresentando na Figura 04.

Figura 04 – Fluxograma da Metodologia



Fonte: O autor (2022).

#### Observação – Indicação Figura 04:

- 01 – Determinação de conceitos e filosofias trazidos pelo *Lean Construction* e pelo BIM aplicados a tecnologia BIM.
- 02 – Utilização dos conceitos e tecnologias para o desenvolvimento de programação visual para se trabalhar com projetos de engenharia através de design generativo.
- 03 – Utilização de softwares (Revit® e Dynamo®) para o desenvolvimento de projetos e programação visual utilizando conceitos estipulados.
- 04 – Aplicação de design generativo em projetos de engenharia civil.
- 05 – Análise de resultados conforme iterações em projetos utilizando os conceitos estipulados.

A metodologia focou então em apresentar os assuntos abordados na revisão bibliográfica, trazendo a relação de como cada tecnologia, conceito ou filosofia depende uma da outra para se chegar aos objetivos estipulados e possibilitar o desenvolvimento.

Apresentou-se a metodologia na ordem em que cada elemento foi utilizado no desenvolvimento, mostrando qual a dependência do tópico seguinte com o tópico anterior.

### 3.1 PARTICIPAÇÃO DO REVIT®

O(s) projetos que complementaram a visualização tridimensional (especificamente a disciplina de arquitetura) e que não forão objetivo de análise utilizando a programação, forão modelados de maneira tridimensional sem foco em informatiza-los (já que suas representações servirão apenas como auxílio), enquanto os projetos de análise (elétricos) forão modelados tridimensionalmente de maneira informatizada com a complementação de famílias (arquivos de Revit® com formato rfa., os quais possuem parâmetros e informações que complementaram o arquivo rvt.) utilizando a ferramenta Revit® versão 2022.01 gratuita para estudantes.

Utilizou-se projetos testes os quais não possuem finalidades executivas para o desenvolvimento e melhoramento da rotina (programação visual) e para comparação de desperdícios utilizou-se um caso exemplo.

A execução da programação em Dynamo® gerou então modificações nos projetos em Revit®, sendo a visualização e análises de qualidade de projetos realizadas observando os resultados em Revit®. As partes de quantidade foram analisadas através de quantitativos automáticos criados em arquivos rvt. e as partes específicas analisando a programação em Dynamo®.

### 3.2 PARTICIPAÇÃO DO DYNAMO®

Para a criação da programação visual utilizou-se o programa Dynamo Core versão 2.10.1.3976 e Dynamo Revit versão 2.10.1.4002. A programação visual foi desenvolvida com a utilização de pacotes (banco de informações que possuem vários nós dentro de si) disponibilizados pelo próprio programa e pacotes disponibilizados pela comunidade Dynamo® (característica de programa *open source*).

Dynamo® possui a característica de permitir o desenvolvimento de nós através da utilização de IronPython. Devido a disponibilidade de pacotes completos dentro do próprio software, não ocorreu a necessidade da utilização da criação de nós próprios (não sendo utilizado programação escrita).

A programação visual para criação da rotina foi então desenvolvida através de nós de Dynamo®, onde cada nó possui um nome específico indicando sua função. Partes de interesse da programação visual foi desenvolvida com a utilização de nós de IronPython, onde utilizou-se programação escrita dentro de nós que foram conectados através de cordas a outros nós.

O(s) projeto(s) em Revit® tiveram sua rotina específica daquela disciplina (elétrico) carregado no software, a rotina foi executada e analisou-se o projeto com base nas informações de antes da utilização do programa em Dynamo® e após sua utilização, observando as modificações/diferenças realizadas em projetos comparação que foram desenvolvidos manualmente.

### 3.3 CONCEITOS UTILIZADOS

Os projetos foram analisados com base em conceitos do *Lean Construction* adaptados a projetos que estão diretamente conectados aos objetivos desta monografia, sendo que alguns conceitos foram divididos em partes para análises específicas, como: (1) redução do tempo voltado ao ciclo de produção, (2) simplificação e redução nos passos dentro da execução de um projeto, (3) melhoria na flexibilidade da produção, (4) previsão do controle de desperdícios e melhor utilização de materiais já em projeto, (5) possibilidade de melhoria contínua e adaptação do processo, (6) mau/super processamento (correções e mudanças após o projeto estar completo), (7) desperdício intelectual, (8) manter todo o processo visível, acessível e compreensível.

Os itens 1, 2, 3 estão relacionados ao desperdício de espera, os itens 4 e 5 estão relacionados ao desperdício de processamento excessivo e produção excessiva, o item 6 está relacionado ao desperdício de defeitos, o item 7 está relacionado ao desperdício de talento subutilizado, todos apresentados no tópico 2.2, e o item 8 é uma característica advinda de gerenciamento de projetos, abrangendo tanto *Lean Construction* e BIM, onde sempre têm-se o interesse de tornar o processo visível.

#### 3.3.1 ANÁLISE DOS CONCEITOS UTILIZADOS

Os conceitos apresentados a seguir foram divididos para facilitar as análises na etapa de resultados.

- (1) Objetivou-se eliminar o tempo gasto em certas etapas de projeto que não demandam raciocínio lógico e necessidade humana quanto ao *design*.
- (2) Pretendeu-se eliminar etapas dentro da execução de projetos utilizando automação de processos.
- (3) Tornar o desenvolvimento do projeto mais simples, possibilitando melhoria nas análises do produto.

(4) Analisou-se a quantidade de materiais utilizados no *design* generativo em comparação a projetos executados “manualmente” por projetistas. Análise da utilização (qualidade) de materiais em projetos executados parcialmente com *design* generativo.

(5) Análise da qualidade/melhoria geral do projeto conforme novo modelo.

(6) Não necessidade de correções/modificações após finalização da etapa específica com *design* generativo (relacionada aos itens 4 e 5).

(7) Eliminação de etapas (relacionada aos itens 1, 2 e 3) que não demandam conhecimento técnico e demandam tempo para sua execução, demandando menos tempo de uma pessoa (profissional/engenheiro) para uma tarefa específica.

(8) Capacidade de toda a programação em Dynamo® de ser acessível, visível e modificável conforme interesse.

Está pesquisa avaliou os resultados relacionados aos desperdícios apresentados nos tópicos 2.2 e 3.5, com base em seus números e qualidade de produto final (partes do projeto que foram criadas utilizando a programação desenvolvida nesta monografia).

### 3.4 EQUIPAMENTO UTILIZADO

Para o desenvolvimento dos projetos em Revit® e desenvolvimento/execução e testes da programação em Dynamo®, utilizou-se um notebook com as seguintes configurações:

- Fabricante do sistema: Acer®
- Modelo do sistema: Aspire A515-51G
- BIOS: V1.15
- Processador: Intel® Core™ i5-8250U CPU @ 1.60GHz (8 CPUs), ~1.8GHz
- Memória: 8192MB RAM
- Placa de vídeo dedicada: NVIDIA® GeForce® MX 130 com 2 GB VRAM
- Versão do DirectX: DirectX 12
- Sistema operacional: Windows 11

Como todo o processo foi desenvolvido no equipamento apresentado, sua capacidade pode se mostrar um fator limitante para o desenvolvimento e apresentação ou tentativas de replicação de produto final.

#### 4 DESENVOLVIMENTO

Os projetos de teste foram desenvolvidos para possibilitar testes iniciais e o desenvolvimento da programação, sendo projetos que apresentam apenas as disciplinas de arquitetura e elétrica (mínimas necessárias para as análises propostas), evitando elementos estruturais e referente à outras instalações devido a questões de simplificação visual e aplicação.

Como o intuito é realizar testes iniciais, optou-se por trabalhar com três modelos simples, sendo o modelo de teste 01 apenas quatro paredes onde serão indicados os pontos elétricos, assim como apresentadas nas Figuras 05 e 06.

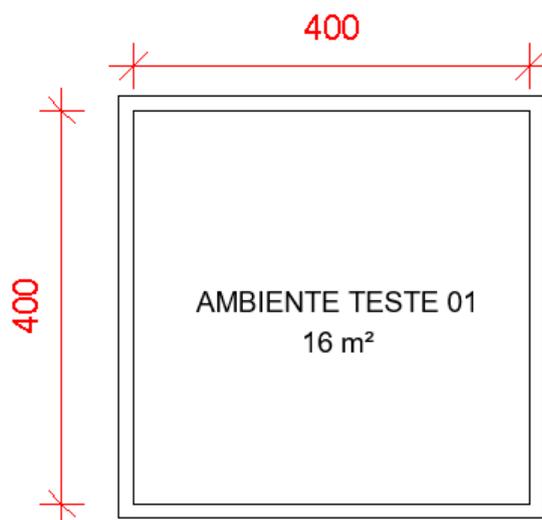
Os projetos apresentados apresentadas não trouxeram informações básicas de projetos (níveis, cotas entre outros) quando não se mostrarem necessárias ao desenvolvimento e foram apresentados em 2 vistas simplificadas, modelo bidimensional (planta baixa) e modelo tridimensional (isométrico).

Devido a característica *open source* do software, vários pacotes são disponibilizados de forma aberta e gratuita para a comunidade, para o desenvolvimento de todos os testes, utilizou-se alguns nós dos pacotes: MEPOver, Clockwork, LunchBox e rhythm.

A modelagem foi desenvolvida utilizando uma template elétrica básica com famílias prontas (tomadas, interruptores, lâmpadas, entre outros), a documentação e organização dessa documentação foi realizada utilizando *tags*, as quais permitem identificar parâmetros inseridos em famílias.

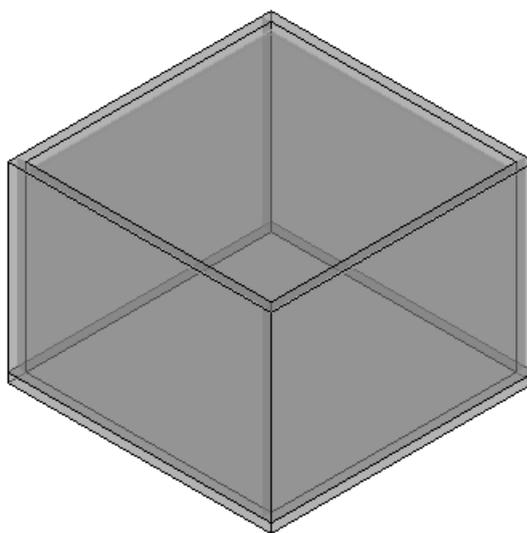
#### 4.1 PROJETO TESTE 01

Figura 05 – Planta baixa simplificada do modelo arquitetônico do projeto teste 01



Fonte: O autor (2022).

Figura 06 – Isométrico simplificado do modelo arquitetônico do projeto teste 01

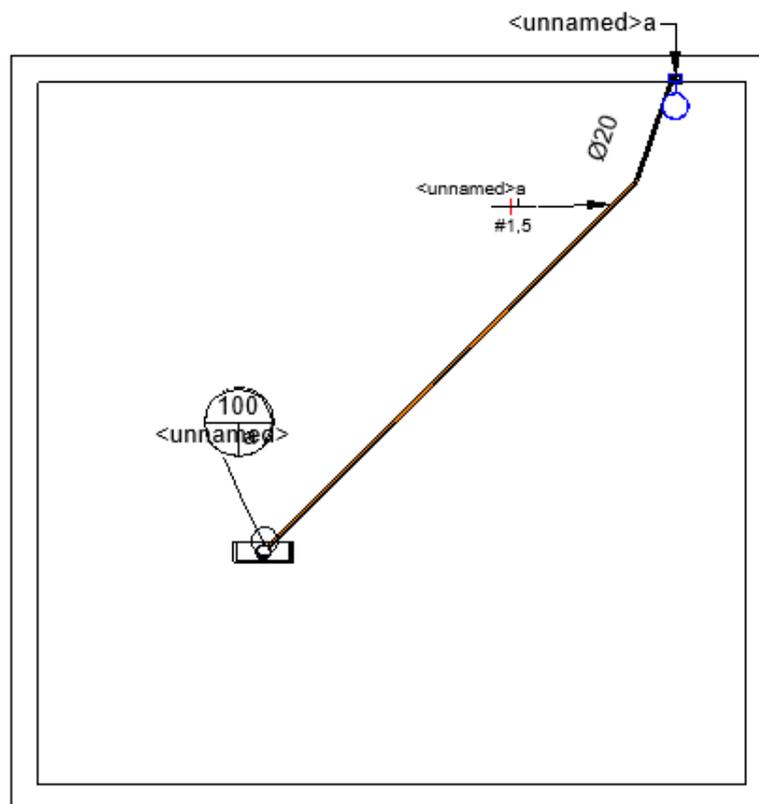


Fonte: O autor (2022).

No teste 01 previu-se um modelo arquitetônico básico, devido ao interesse de se trabalhar com o modelo elétrico especificamente. Como a programação foi desenvolvida com foco na disciplina elétrica, o modelo arquitetônico não apresentou impacto, sendo então optado por um layout simples.

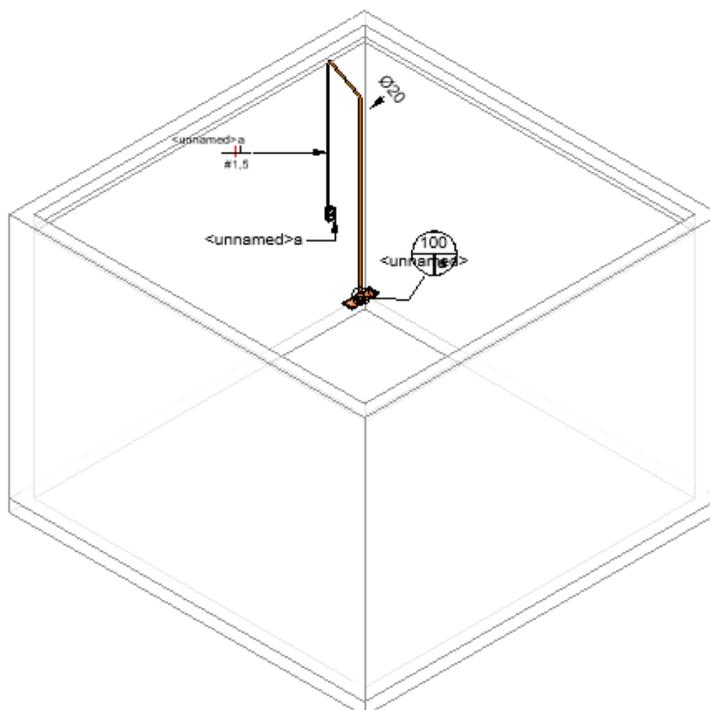
Para todas as apresentações dos testes 01, 02 e 03 das disciplinas arquitetônico e elétrico, utilizou-se a escala 1:50 e unidades em centímetros, assim como apresentado nas Figuras 07 e 08.

Figura 07 – Planta baixa simplificada do modelo elétrico do projeto teste 01



Fonte: O autor (2022).

Figura 08 – Isométrico simplificado do modelo elétrico do projeto teste 01



Fonte: O autor (2022).

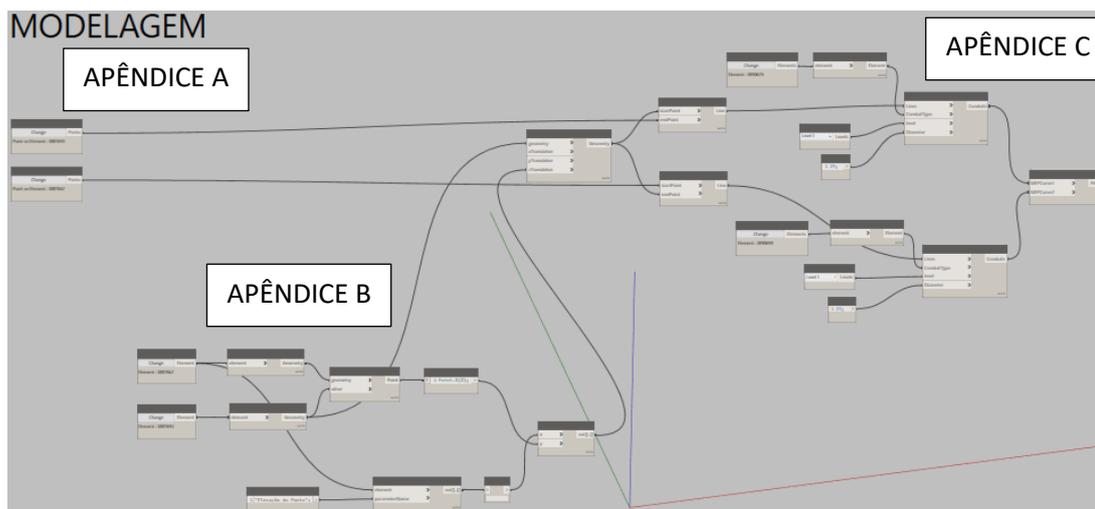
Os modelos apresentados nas Figuras 07 e 08 tiveram o lançamento de pontos elétricos (interruptor e caixa ortogonal para ponto de luz), modelagem de conduítes, documentação e organização desenvolvidos manualmente utilizando ferramentas padrões do Revit®. Observa-se que na documentação básica apresentada, o texto “<unnamed>” se repete em alguns pontos, isso se deve à falta de informatização (parâmetros) no elemento, os quais a *tag* não consegue identificar devido à informação não ter sido fornecida. No caso específico das Figuras 07 e 08, o motivo se deve a não criação de circuitos elétricos, os quais necessitariam estarem associados a um quadro de distribuição central (QDC).

Como o básico de projetos elétricos demanda diferentes elementos, seria impossível resolver o erro de *tag* sem inserir mais informações no modelo, não sendo o intuito do teste 01 trabalhar com vários elementos, pois isto dificultaria o desenvolvimento inicial.

#### 4.1.1 PROGRAMAÇÃO VISUAL BASEADA NO PROJETO TESTE 01

Com o interesse de simplificar o processo de modelagem, documentação e organização, dividiu-se em partes a programação, sendo apresentado na Figura 09 a programação visual utilizada para o desenvolvimento de modelagem.

Figura 09 - Programação visual (modelagem) baseada no teste 01



Fonte: O autor (2022).

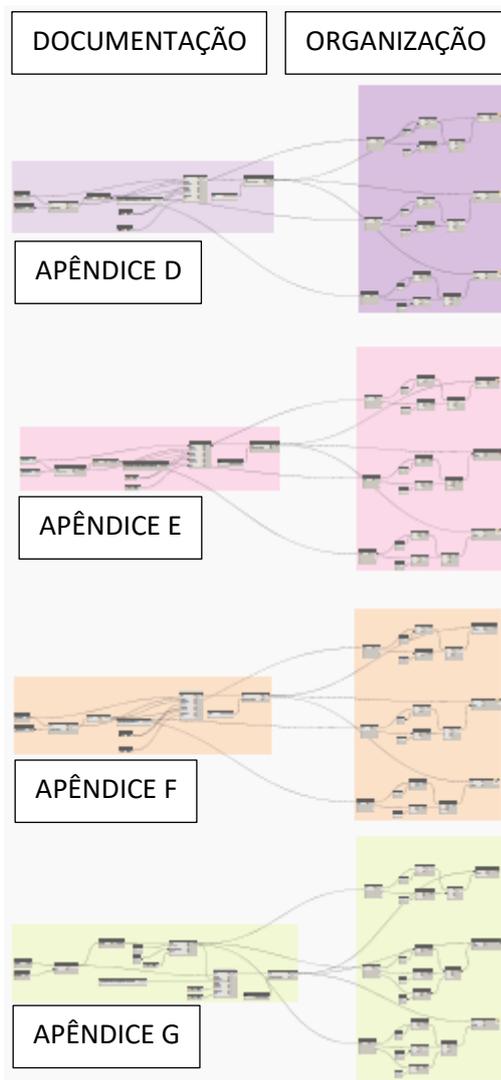
A Figura 09 representa o produto completo responsável para modelagem no teste 01, o qual desenvolve a modelagem bidimensional e tridimensional de conduítes.

A programação visual demonstrada na Figura 09 necessita que os pontos os quais tem-se o interesse de serem conectados sejam indicados através de seleção manual, tendo então o traçado de conduítes gerados a partir desses pontos.

O lançamento de pontos elétricos como tomadas e interruptores utilizando a programação não se mostrou interessante, no momento em que demandam informatização de necessidades de projeto arquitetônico como divisão de ambientes e informações geométricas (área, largura, comprimento e altura) desses ambientes, trazendo o foco para o modelo arquitetônico.

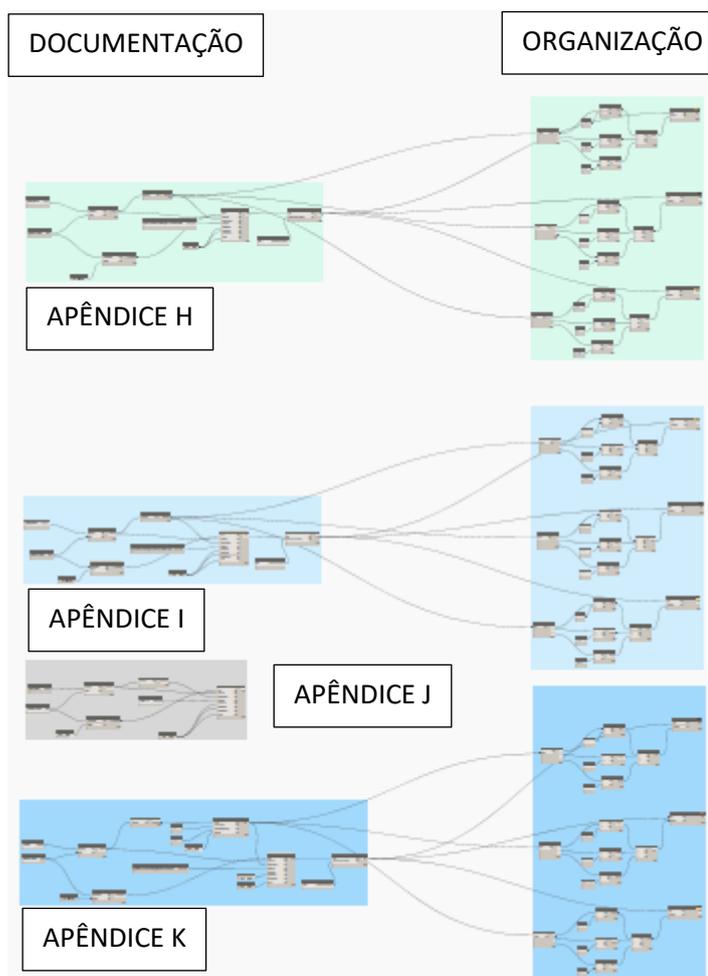
As Figuras 10 e 11 representam então a documentação e organização do projeto elétrico sem dependerem do projeto arquitetônico.

Figura 10 - Programação visual (documentação e organização bidimensional)  
baseada no teste 01



Fonte: O autor (2022).

Figura 11 - Programação visual (organização tridimensional) baseada no teste 01



Fonte: O autor (2022).

As Figuras 10 e 11 funcionam de maneira similar, com as diferenças de na representação tridimensional ser necessário trabalhar com uma terceira dimensão (x, y e z), enquanto a representação bidimensional necessita apenas de informações 2D (x e y).

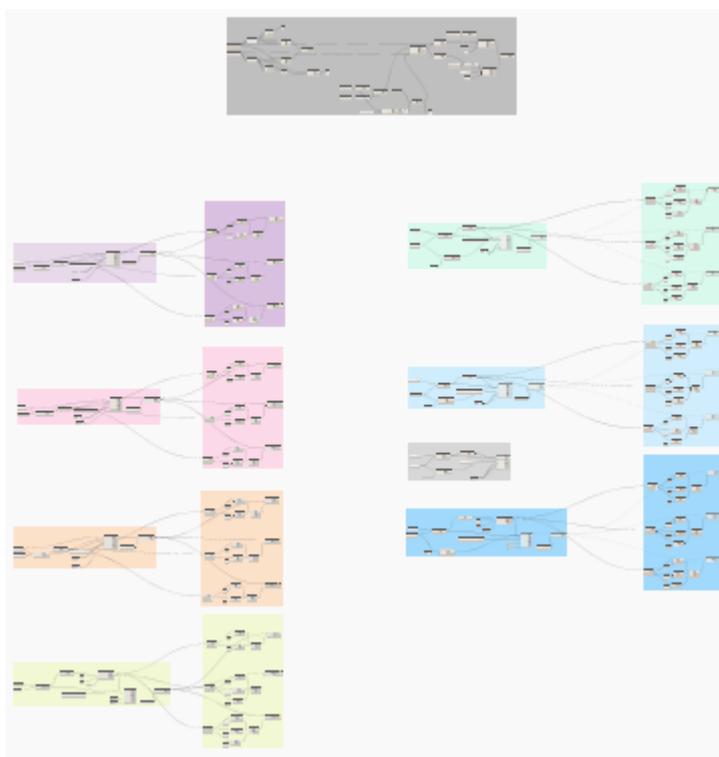
As Figura 10 e 11 representam o produto responsável pela documentação e organização no teste 01, o qual desenvolve a documentação e organização de todos os elementos presentes no projeto. A documentação é responsável por informatizar em projeto os parâmetros apresentados nos elementos, e parte de organização é responsável por dispor tais informações.

Para a parte de documentação cada grupo (nós apresentados em cores diferentes) é responsável por informatizar um elemento específico, sendo a ordem: luminárias, interruptor, fiação e conduítes.

Como os nós possuem escalas de difícil visualização, no tópico de desenvolvimento apresentou-se apenas a rotina de uma forma geral, sem detalhamento devido à dificuldade de representação e detalhamento visual.

As Figuras 13 e 14 representam a modelagem, documentação e organização bidimensional e tridimensional desenvolvidas utilizando a programação apresentada na Figura 12.

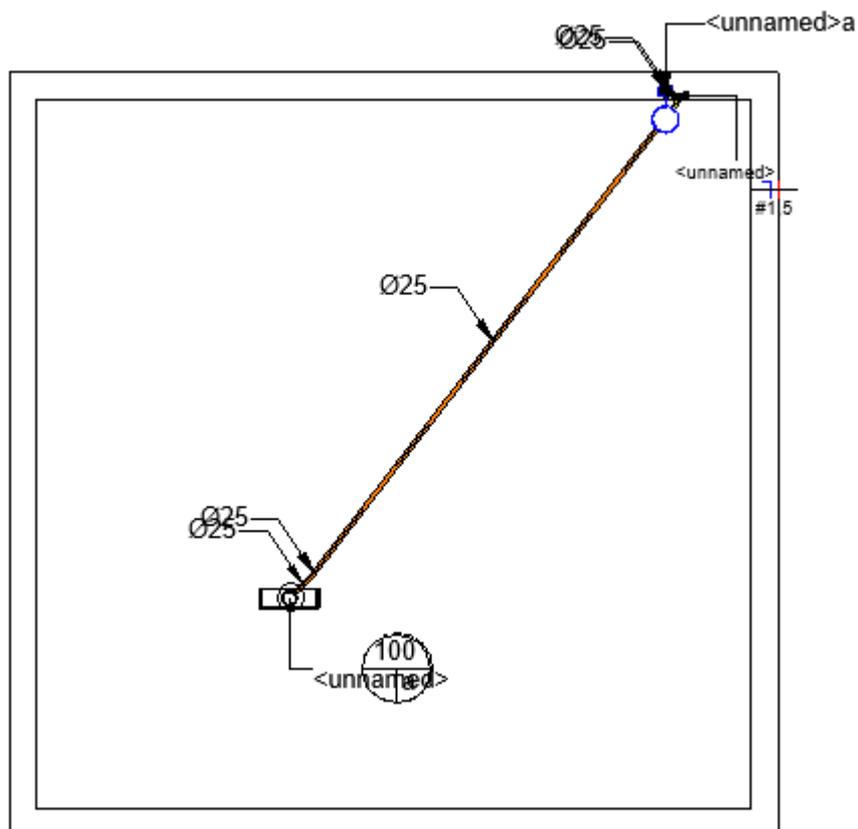
Figura 12 - Programação visual completa baseada no teste 01



Fonte: O autor (2022).

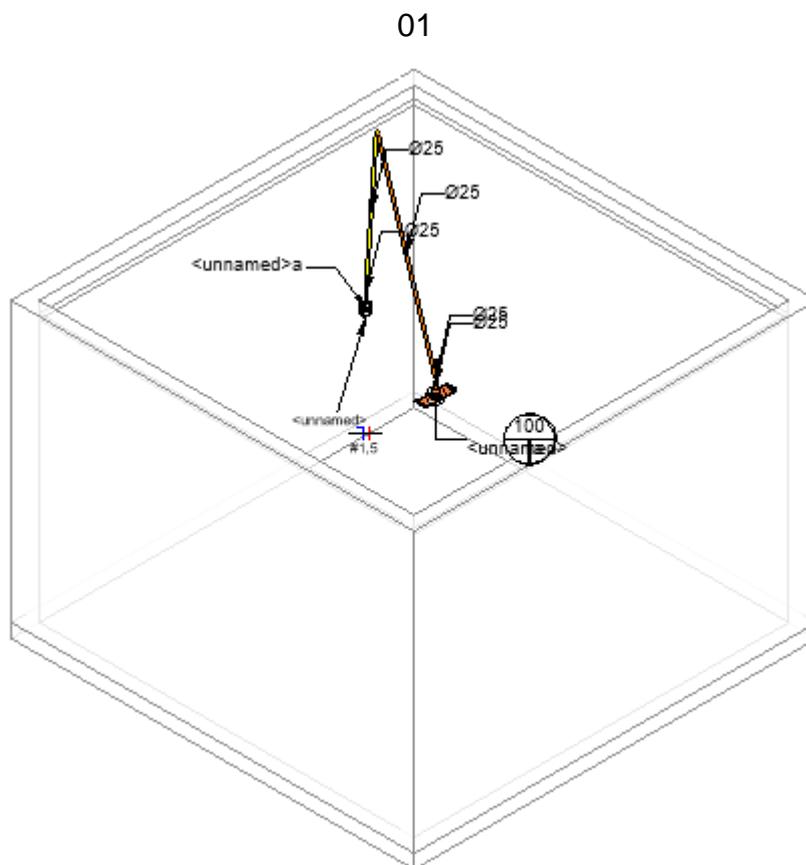
Figura 13 – Planta baixa simplificada do projeto teste 01 desenvolvido com a programação visual (modelagem, documentação e organização) baseada no teste

01



Fonte: O autor (2022).

Figura 14 - Isométrico simplificado do projeto teste 01 desenvolvido com a programação visual (modelagem, documentação e organização) baseada no teste



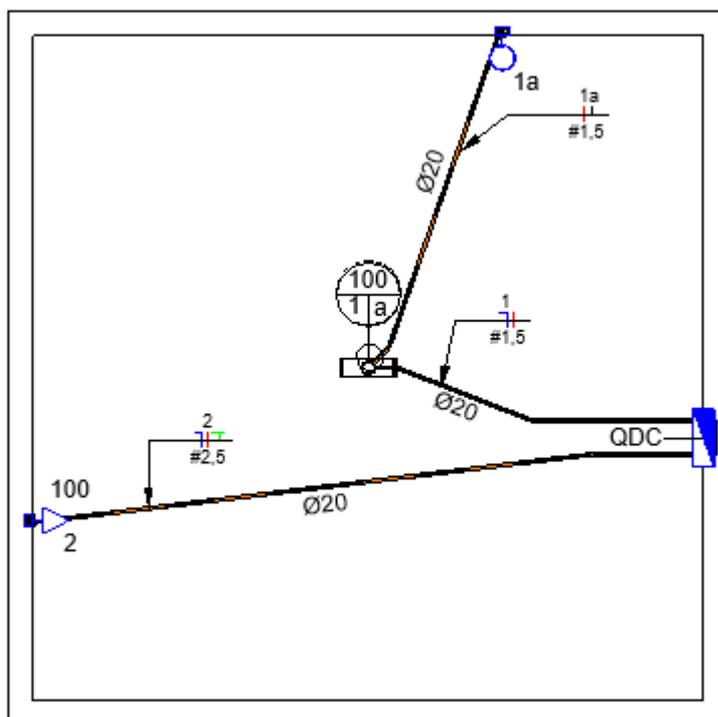
Fonte: O autor (2022).

#### 4.2 PROJETOS TESTE 02

Como o modelo arquitetônico não apresentou impacto no teste 01, utilizou-se o mesmo modelo para teste 02, tendo mudanças apenas no modelo elétrico. Como não existem cômodos definidos no Ambiente 01 ou necessidades específicas, previu-se em localizações quaisquer elementos elétricos (1 tomada, 1 interruptor, 1 ponto de iluminação e 1 QDC) que servirão de exemplo para o desenvolvimento no teste 02.

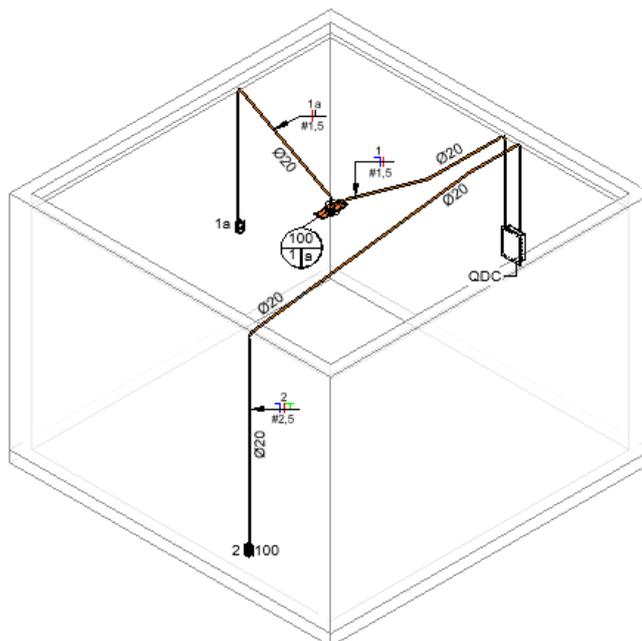
Os pontos elétricos foram modelados (representação 3D dos pontos elétricos e eletrodutos), informatizados (indicação de informação técnicas: divisão de circuitos, VA do ponto, espessura do eletroduto, espessura da fiação, tipo de fiação, nome do elemento e identificação do interruptor) e essa informação foi organizada, evitando poluir o projeto e às tornando visíveis, assim como representado nas Figuras 16 e 17.

Figura 15 – Planta baixa simplificada do modelo elétrico do projeto teste 02



Fonte: O autor (2022).

Figura 16 – Isométrico simplificado do modelo elétrico do projeto teste 02



Fonte: O autor (2022).

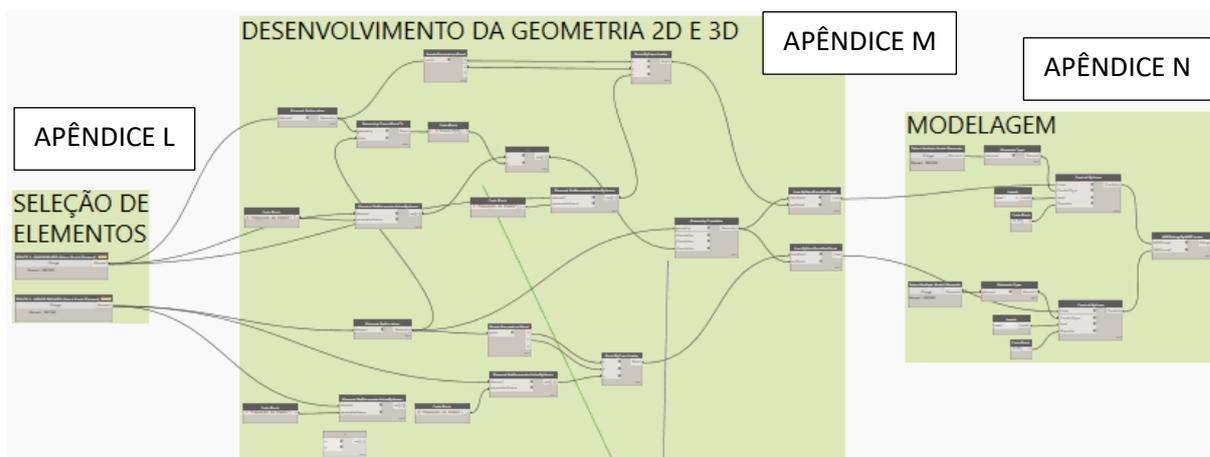
Como apresentado nas figuras 15 e 16, o projeto elétrico do teste 02 possui dois circuitos, tendo o circuito 1 apenas uma luminária e seu interruptor e o circuito 2 apenas uma tomada.

#### 4.2.1 PROGRAMAÇÃO VISUAL BASEADA NO PROJETO TESTE 02

A programação visual baseada no teste 02 reutilizou vários elementos apresentados no teste 01, porém com a diferença da necessidade de ser utilizado na ferramenta “Dynamo Player”, devido a necessidade de manter os elementos criados a cada vez que a rotina era compilada.

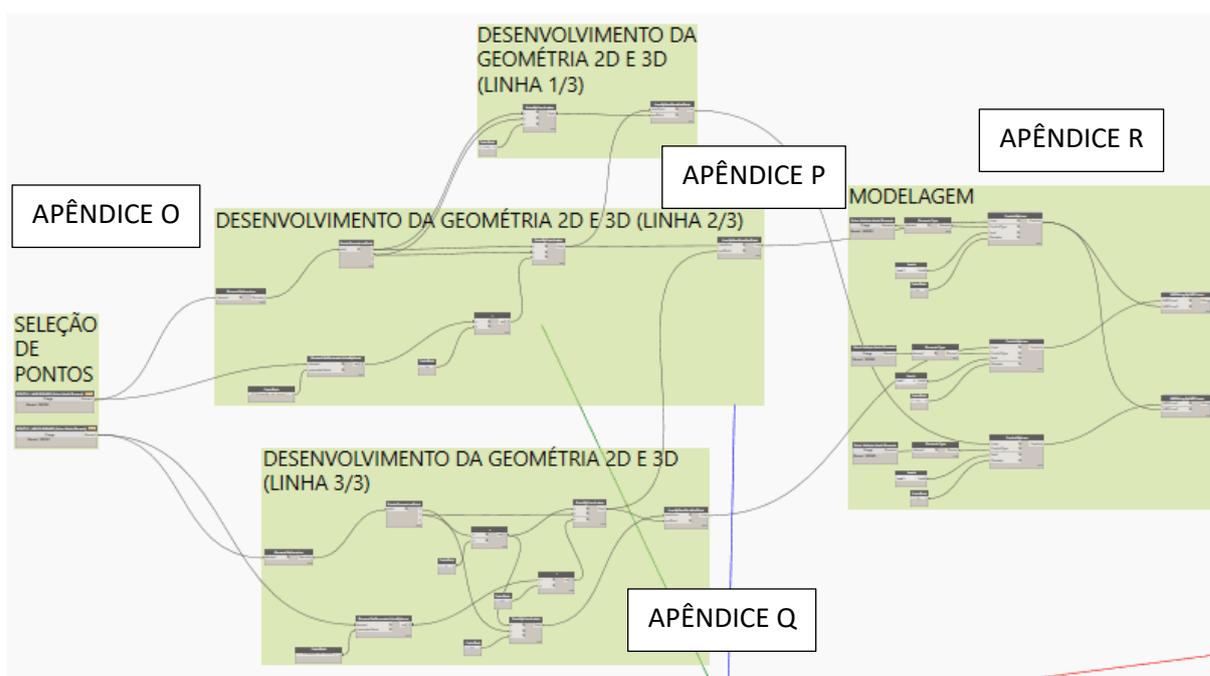
Por padrão, o Dynamo® possui uma função imutável de eliminar todos os elementos desenvolvidos pela rotina caso suas indicações sejam diferentes das originais, tornando assim impossível desenvolver o processo de conexão de vários pontos dentro do Dynamo®. O Dynamo Player possui a função contrária, onde os processos originais são mantidos mesmo caso as informações originais sejam alteradas.

Figura 17 - Programação visual (modelagem 1 de 02) baseada no teste 02



Fonte: O autor (2022).

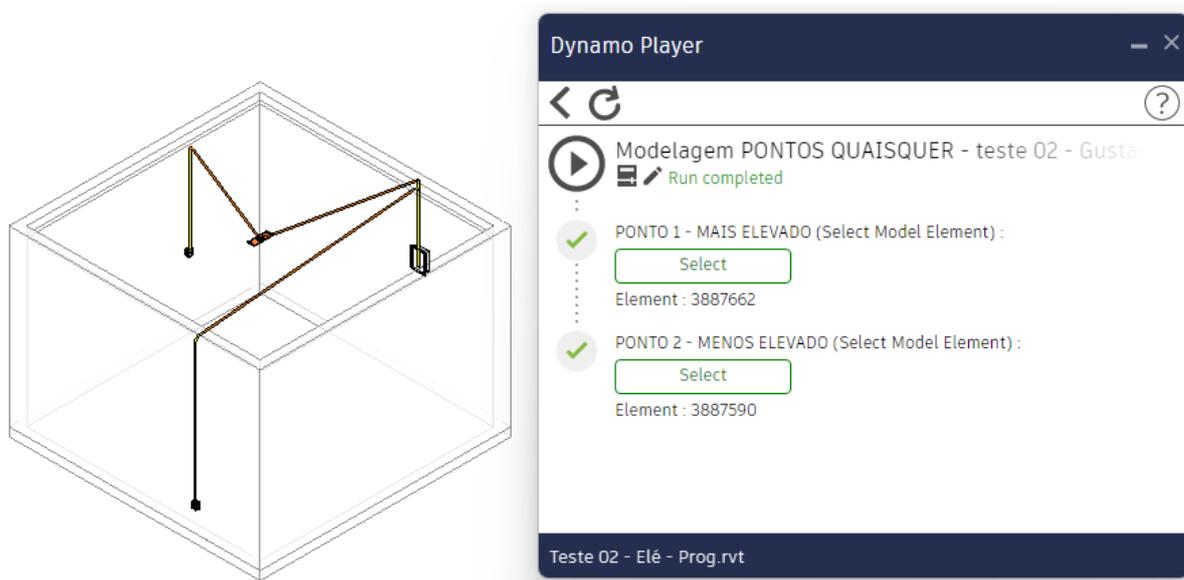
Figura 18 - Programação visual (modelagem 02 de 02) baseada no teste 02



Fonte: O autor (2022).

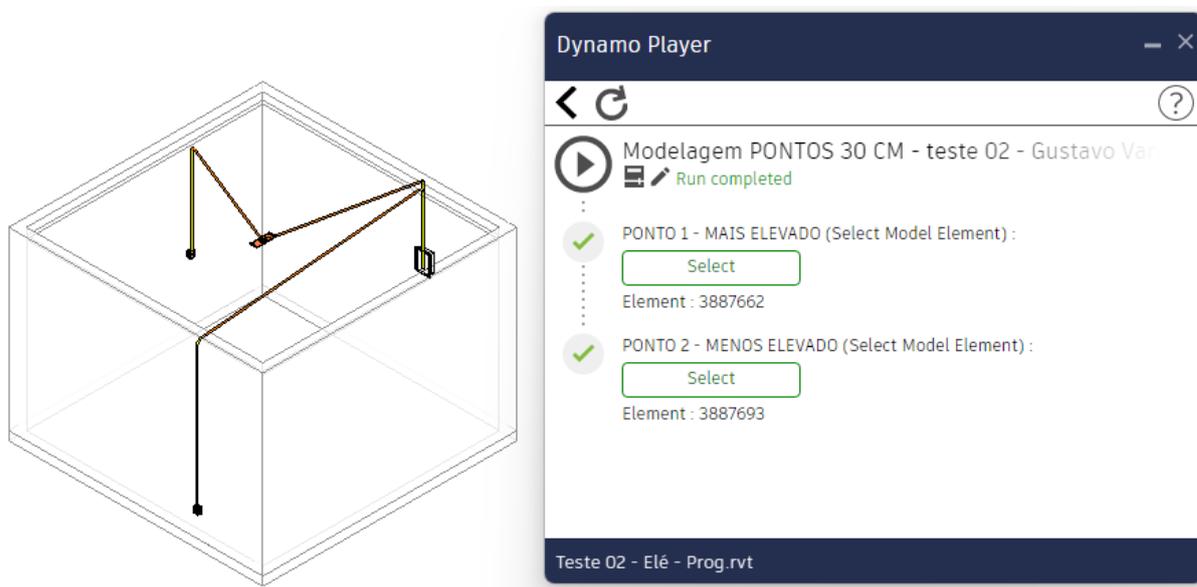
As Figuras 19 e 20 representam a utilização das rotinas trazidas nas Figuras 17 e 18 no Dynamo Player.

Figura 19 – Visualização de Dynamo Player da programação visual da Figura 17



Fonte: O autor (2022).

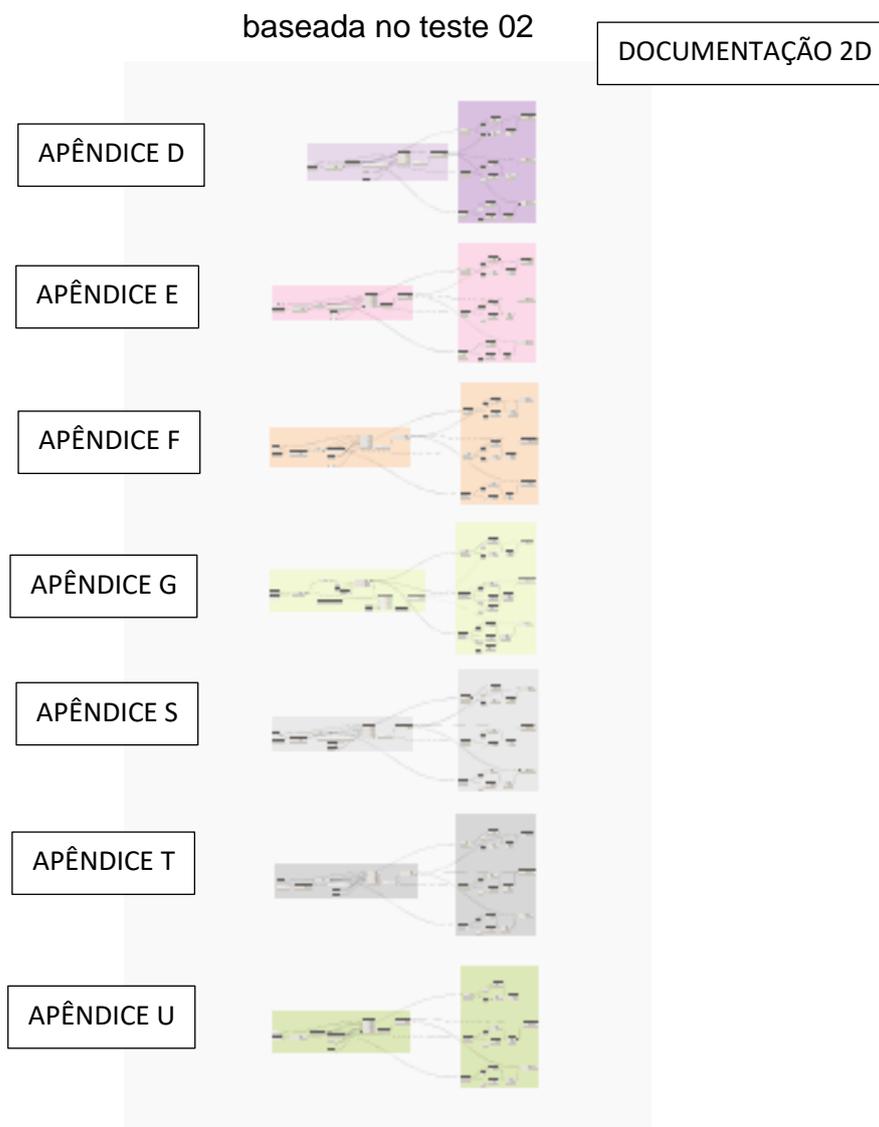
Figura 20 – Visualização de Dynamo Player da programação visual da Figura 18



Fonte: O autor (2022).

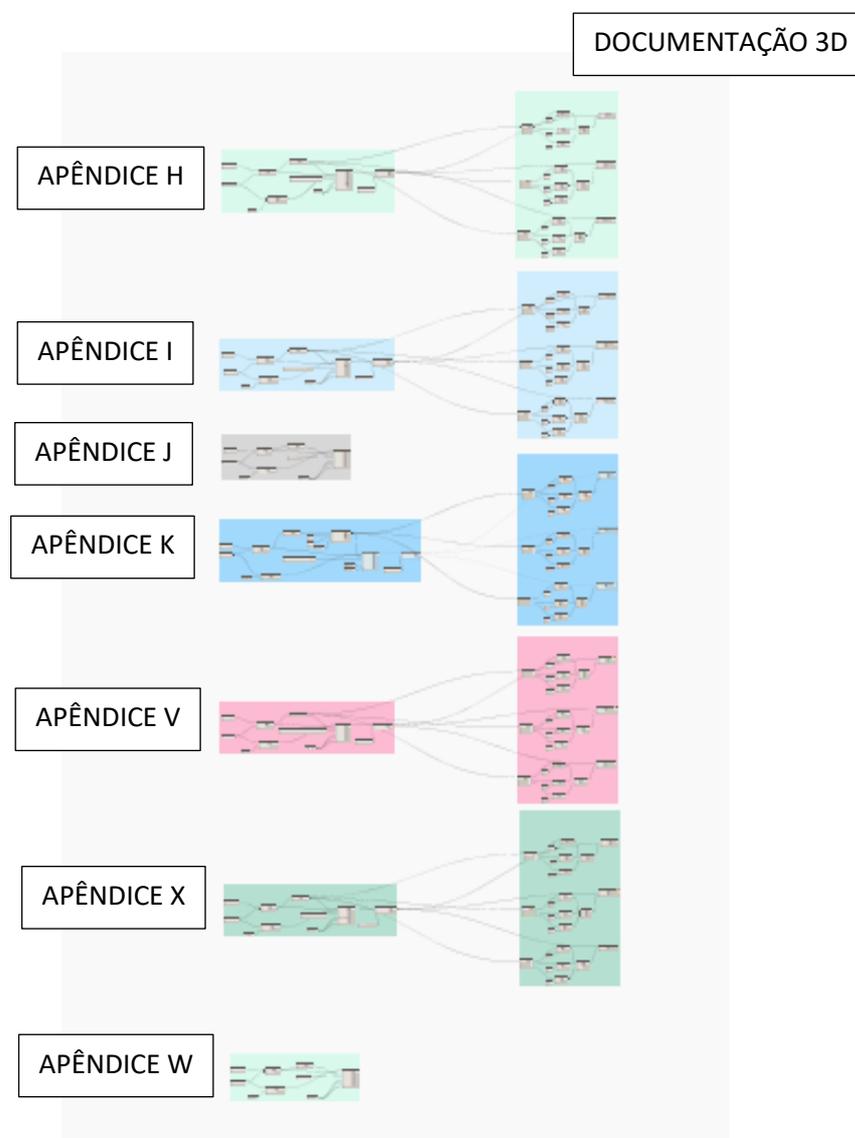
Para o desenvolvimento do apresentado nas Figuras 23 e 24 utilizou-se a programação desenvolvida nas Figuras 21 e 22.

Figura 21 - Programação visual (documentação e organização bidimensional)



Fonte: O autor (2022).

Figura 22 - Programação visual (documentação e organização tridimensional)  
baseada no teste 02

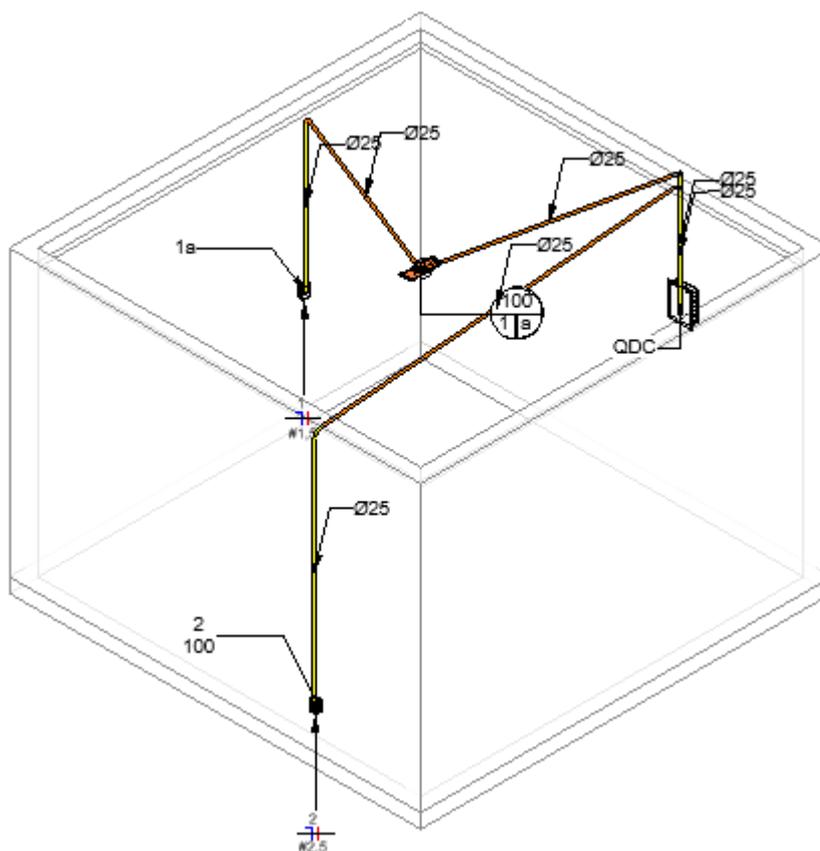


Fonte: O autor (2022).



Figura 24 - Isométrico simplificado do projeto teste 01 desenvolvido com a programação visual (modelagem, documentação e organização) baseada no teste

01



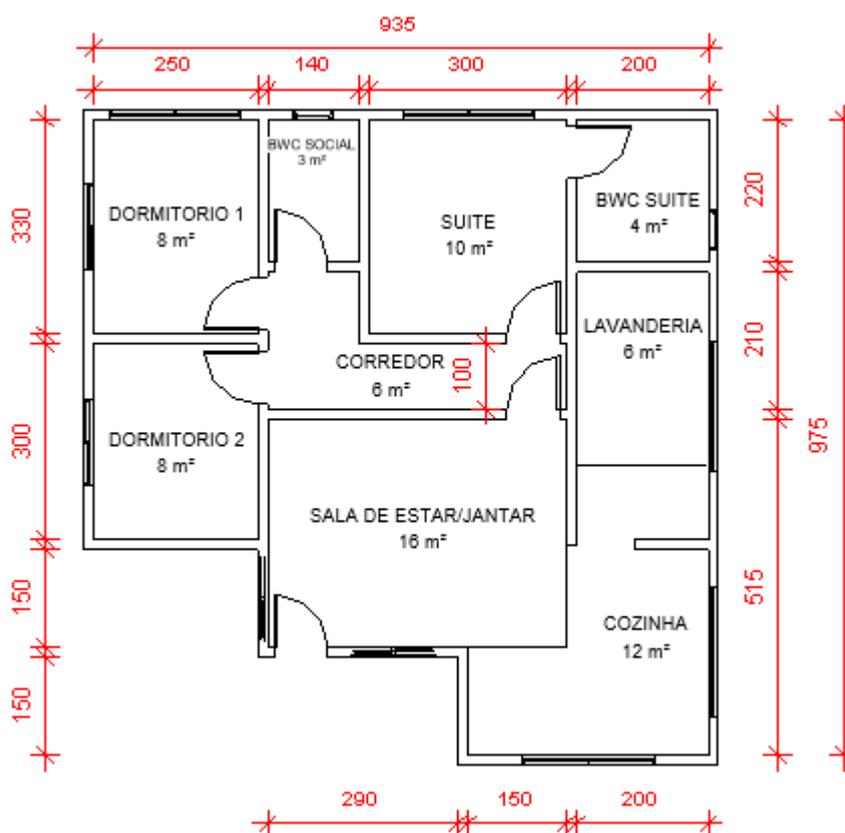
Fonte: O autor (2022).

### 4.3 PROJETOS TESTE 03

Projetos elétricos reais demandam dimensionamentos e planejamentos prévios baseados em normas e metodologias conhecidas, como o intuito desta monografia não é a apresentação ou aprofundamento em projetos elétricos, questões de dimensionamento não serão abordados pela tecnologia apresentada, sendo o foco a modelagem, documentação e organização.

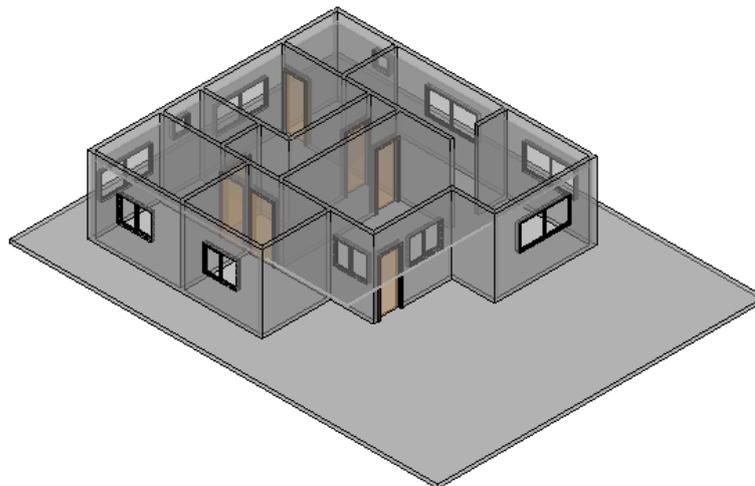
Nas Figuras 25 a 30 foi apresentado o projeto teste 03.

Figura 25 – Planta baixa simplificada do modelo arquitetônico do projeto teste 03



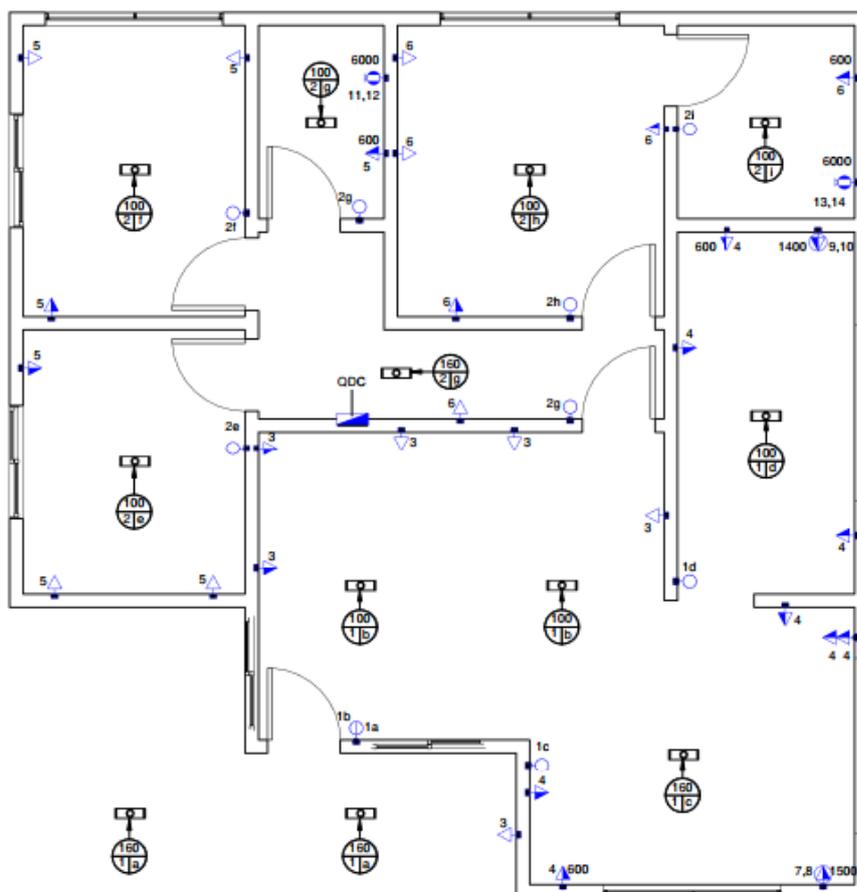
Fonte: O autor (2022).

Figura 26 – Isométrico simplificado do modelo arquitetônico do projeto teste 03



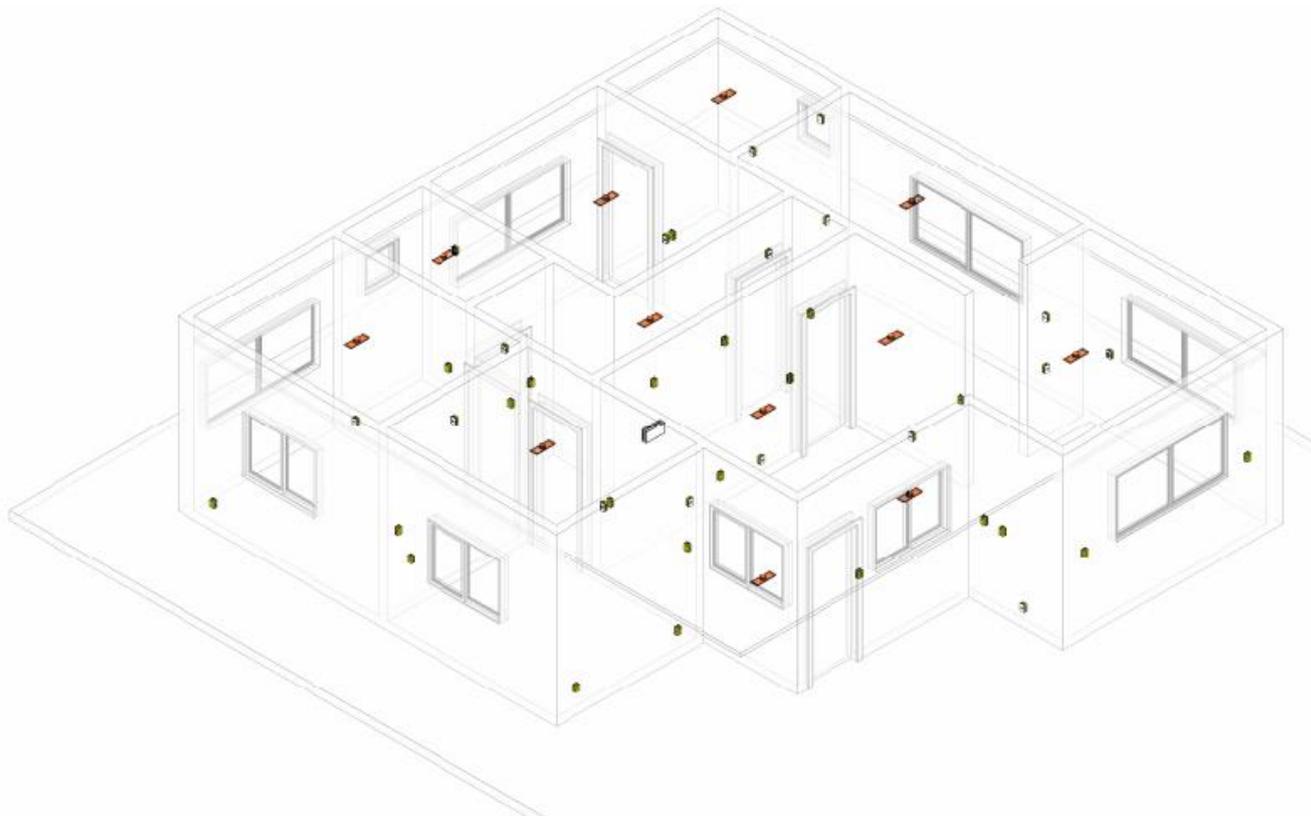
Fonte: O autor (2022).

Figura 27 – Planta baixa simplificada do modelo elétrico do projeto teste 03 (01/02)



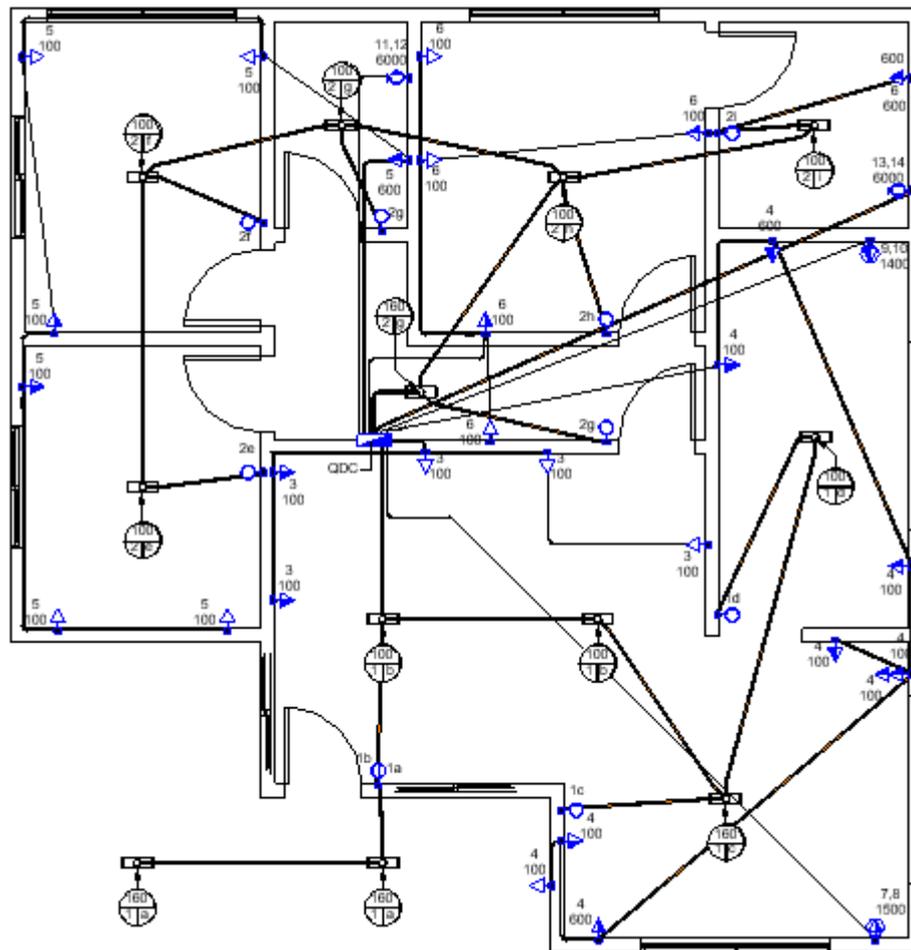
Fonte: O autor (2022).

Figura 28 – Isométrico simplificado do modelo elétrico do projeto teste 03 (01/02)



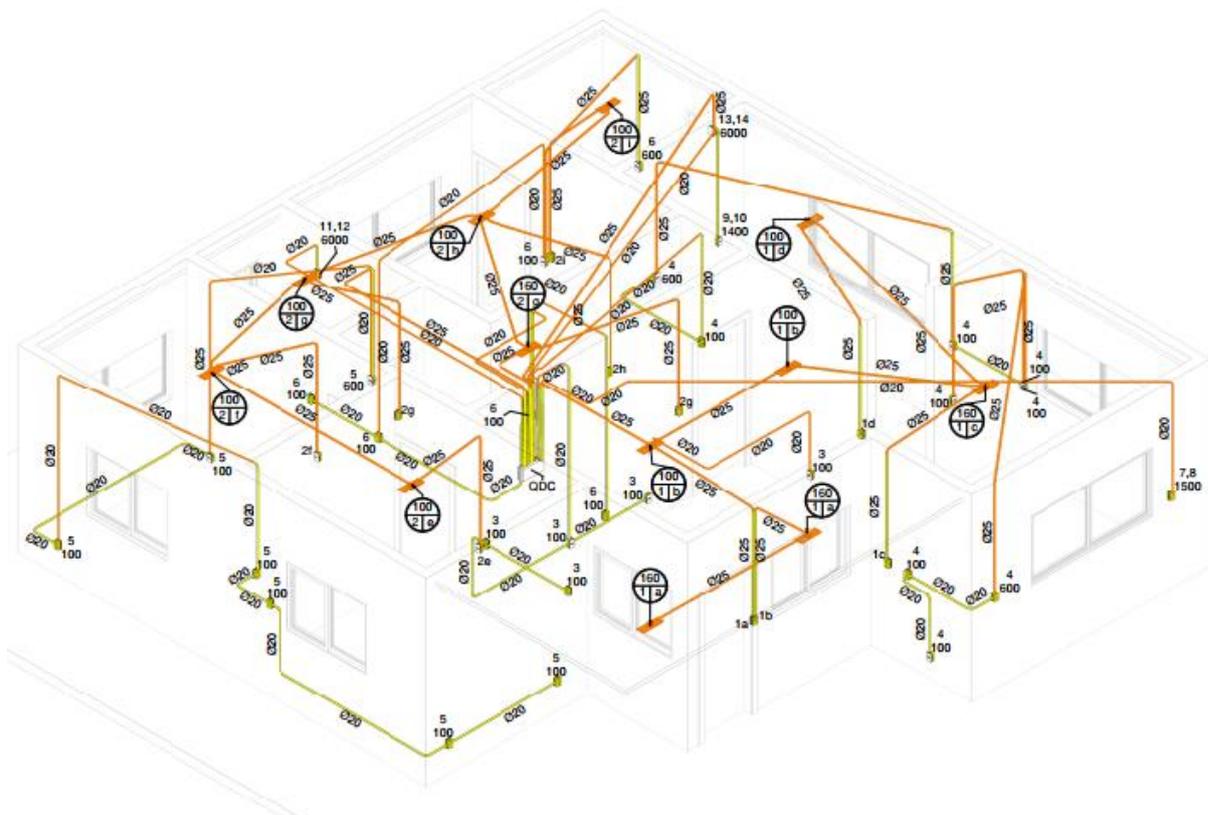
Fonte: O autor (2022).

Figura 29 – Planta baixa simplificada do modelo elétrico do projeto teste 03 (02/02)



Fonte: O autor (2022).

Figura 30 – Isométrico simplificado do modelo elétrico do projeto teste 03 (02/02)



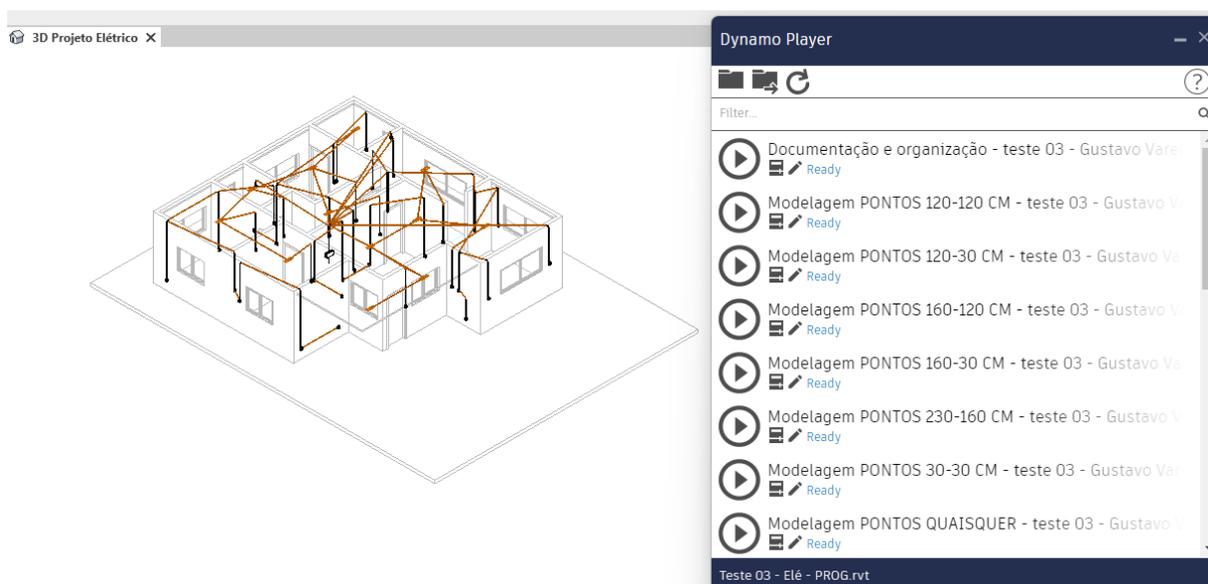
Fonte: O autor (2022).

#### 4.3.1 PROGRAMAÇÃO COM BASE NO PROJETO TESTE 03

O modelo do teste 03 trouxe todos os elementos (famílias) apresentados no teste 02, tendo então a programação apresentada nas Figuras 21 e 22 reutilizadas no teste 03. Embora o projeto não traga informações completas conforme necessidades de execução, foi possível demonstrar que todo parâmetro é capaz de ser informatizado em projeto através da programação visual caso exista previamente em família, embora a organização possa se tornar mais trabalhosa conforme a quantidade e complexidade de tais informações.

A modelagem com relação a programação apresentada no teste 1 possuía a limitação de poder trabalhar apenas com dois pontos por vez, tendo sido eliminado tal limitação como apresentado no teste 02 com a utilização do Dynamo. Como as programações apresentadas nas Figuras 17, 18, 19 e 20 são capazes de se adaptar, serão reutilizadas, tendo a modelagem do teste 03 apenas complementada conforme necessidade, assim como apresentado na Figura 31.

Figura 31 - Programação visual (parte da modelagem) baseada no teste 03

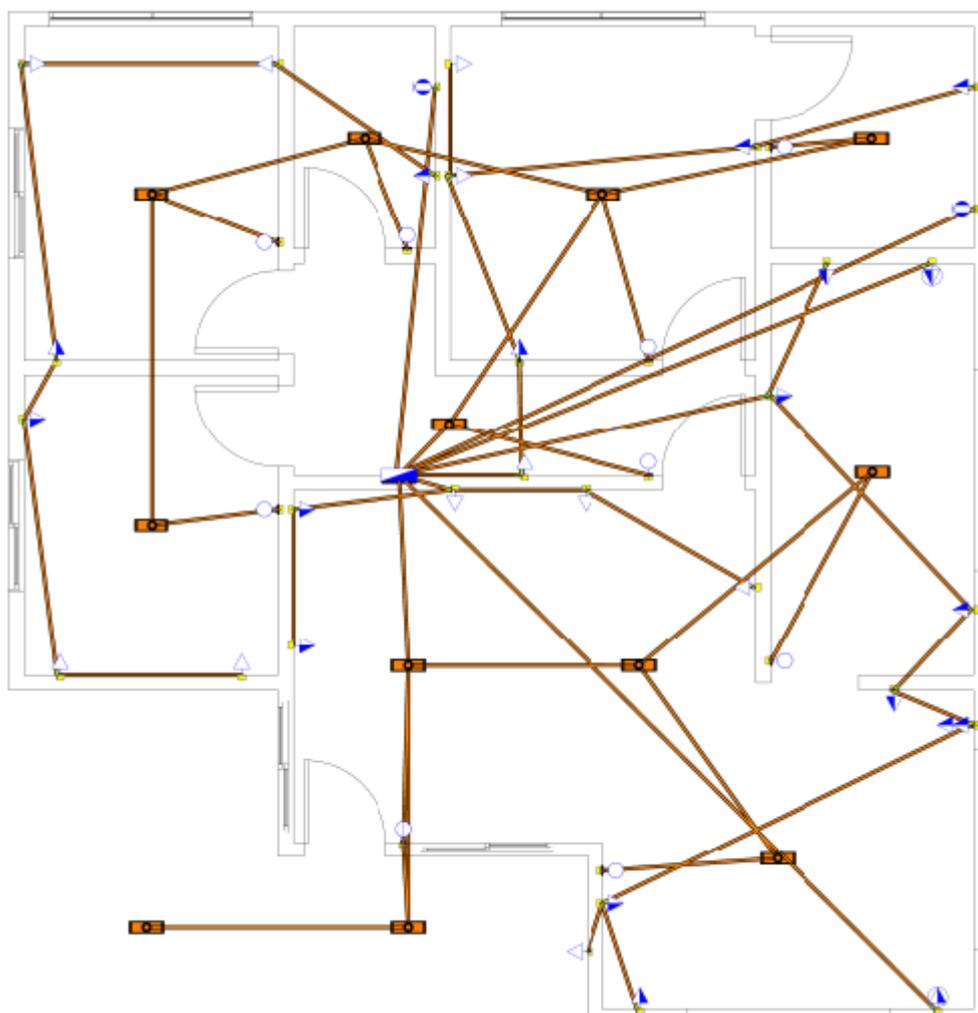


Fonte: O autor (2022).

A Modelagem 3 apresentou limitações devido a maneira como o Revit® observa pontos tridimensionais. A coordenada Z dos pontos não pode ser padronizada, sendo necessário gerar uma repetição de rotina quando surgia a necessidade de conectar dois elementos com coordenada Z diferentes que não estavam presentes no nível do forro (nível das luminárias).

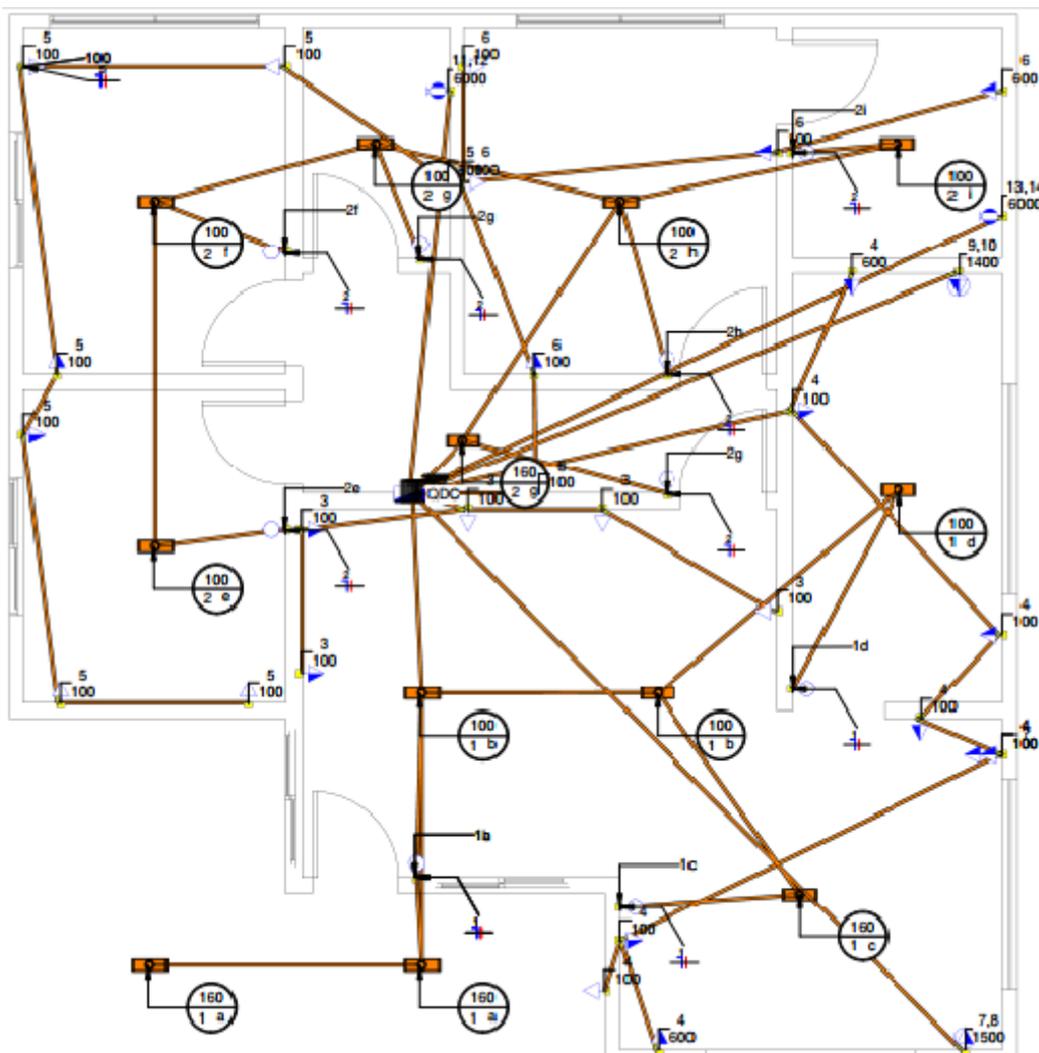
As rotinas apresentadas em Dynamo Player na Figura 31 foram todas as rotinas necessárias (um total de 7) para a conexão dos elementos elétricos do modelo de maneira permitir traçados realistas.

Figura 32 – Planta baixa simplificada do modelo elétrico do projeto teste 03 baseada na programação visual teste 03 (01/03)



Fonte: O autor (2022).

Figura 33 – Planta baixa simplificada do modelo elétrico do projeto teste 03 baseada na programação visual teste 03 (02/03)



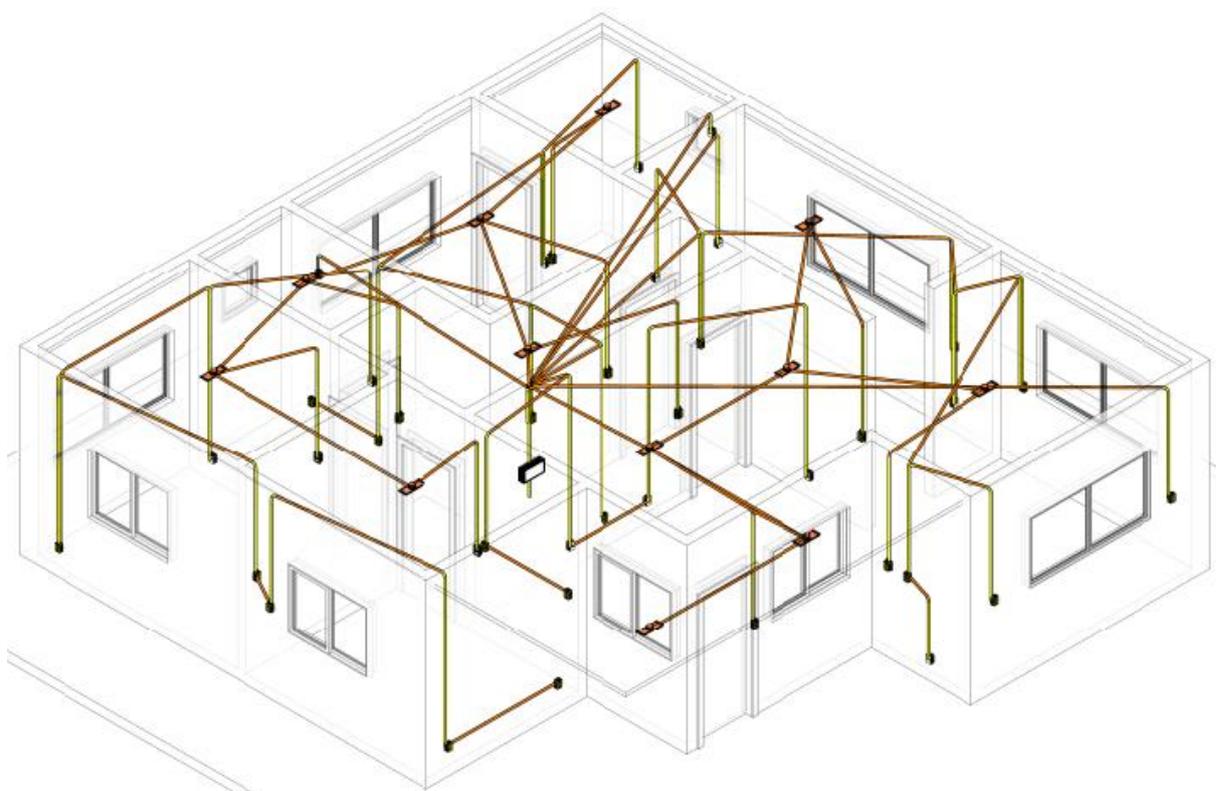
Fonte: O

autor (2022).



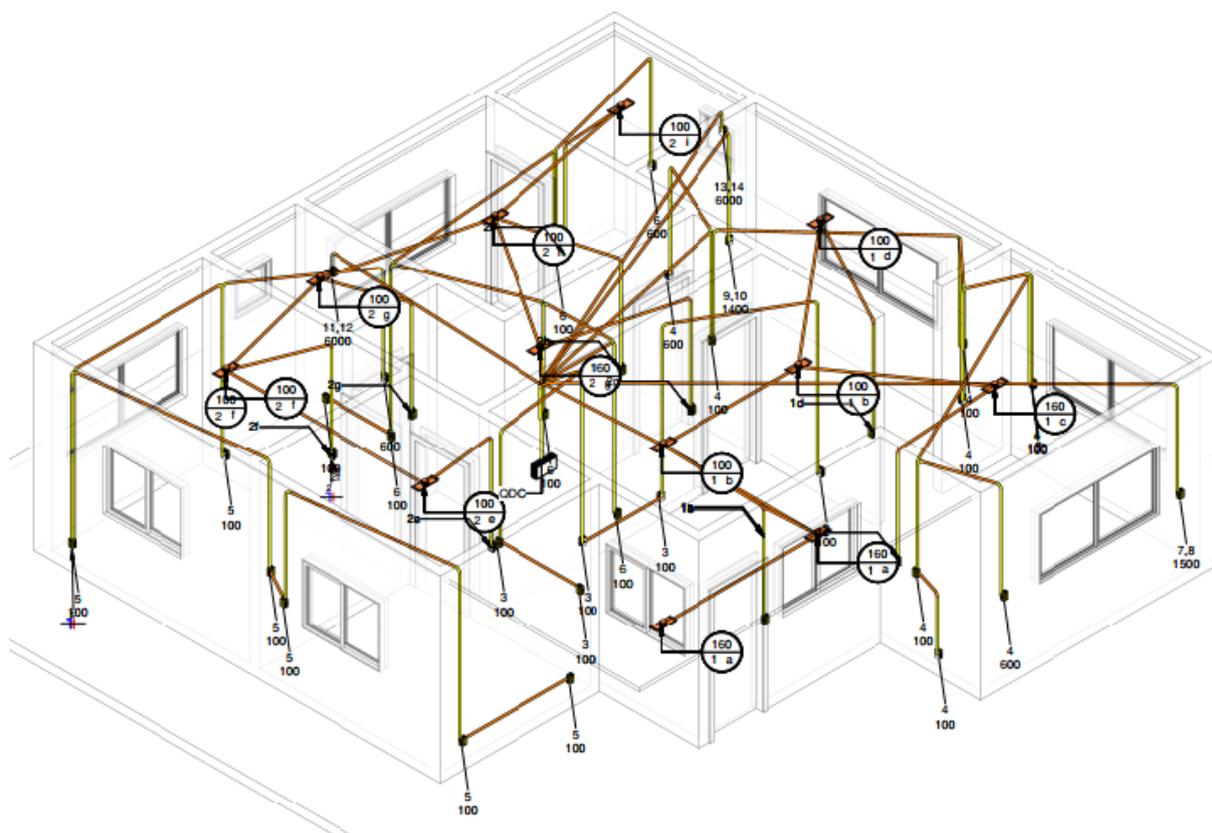
No caso do desenvolvimento apresentado, embora a programação visual de fiação relacionada a tomadas funcione para  $n$  elementos (como apresentado na Figura 23 o qual mapeou o projeto para identificar os elementos sem a necessidade de indicação), a capacidade do computador (no caso a máquina identificada no tópico 3.6) de compreender juntamente com a otimização do Dynamo® e forma de desenvolvimento da programação visual, tornam o processo inviável.

Figura 35 – Isométrico simplificado do modelo elétrico do projeto teste 03 baseada na programação visual teste 03 (01/03)



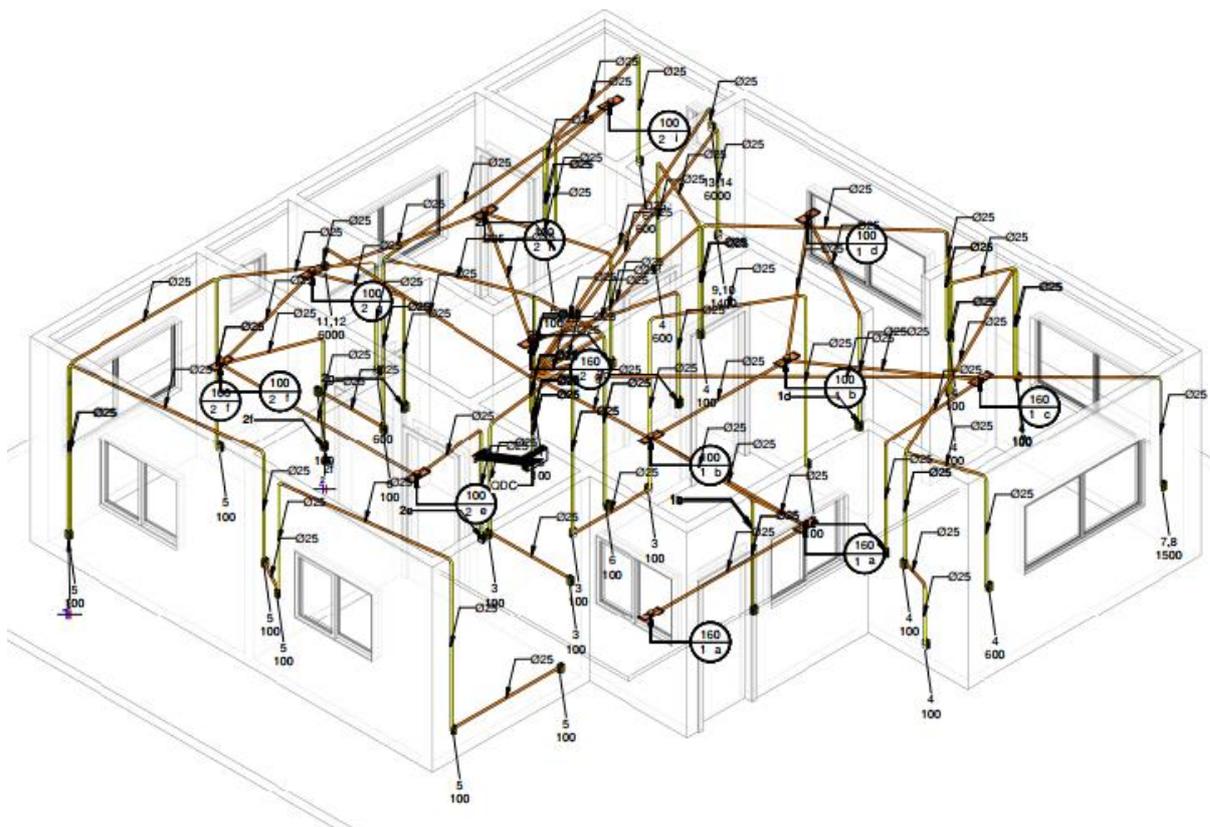
Fonte: O autor (2022).

Figura 36 – Isométrico simplificado do modelo elétrico do projeto teste 03 baseada na programação visual teste 03 (02/03)



Fonte: O autor (2022).

Figura 37 – Isométrico simplificado do modelo elétrico do projeto teste 03 baseada na programação visual teste 03 (03/03)



Fonte: O autor (2022).

Vídeos de apresentação do desenvolvimento do teste 03 de forma manual e utilizando a programação escrita podem ser visualizados acessando o QR CODE presente no tópico 9.2 APÊNDICE II.

## 5 RESULTADOS E DISCUSSÕES

Nos testes 01, 02 e 03 chegou-se a um LOD 300 tanto no desenvolvimento manual como no desenvolvimento utilizando a programação, conforme definição apresentada no tópico 2.1, demonstrando que mesmo de forma simplificada, a programação possui a capacidade em se assemelhar a um processo de desenvolvimento manual.

A programação apresentada no teste 01 se mostrou limitada na parte de modelagem com relação a quantidade de pontos elétricos que consegue conectar, sendo apenas dois pontos elétricos a cada execução da rotina, não salvando a modelagem realizada devido a funcionalidades internas do Dynamo. As programações de todos os testes se mostraram capaz de realizar o desenvolvimento tridimensional, diferenciando os tipos de eletrodutos específicos através de uma simplificação de laje (laranja) por serem em sua maioria horizontais e parede (amarelo) por ser maioria verticais.

Com relação a documentação, a programação dos testes 01, 02 e 03 foram capazes de documentar todos os elementos presentes no projeto com as mesmas informações seguindo o apresentado no desenvolvimento manual, porém de forma mais detalhada devido ao fato da programação não diferenciar elementos da mesma família, trazendo uma vantagem em comparação ao processo manual de selecionar todos os elementos.

Com relação a organização dos testes 01, 02 e 03 a programação apresenta uma melhoria em relação ao desenvolvimento manual que foi a padronização das linhas de chamada das *tags*, no entanto, não sendo capaz de ajustar sozinho o problema de sobreposição e indicação de posição de *tags* em nenhum dos modelos de análise. A vantagem de a documentação apresentar todos os elementos do projeto se tornou uma desvantagem para a organização, sendo difícil evitar sobreposições devido a aleatoriedade em que os layouts elétricos podem ocorrer, apresentado principalmente no teste 03 o problema com a *tag* de eletrodutos.

Com relação a parte de informatização dos elementos, enquanto todos as informações requisitadas foram apresentadas, as mesmas eram apenas informações padrões (como exemplo o diâmetro de conduítes que se manteve em 25 mm para todo o projeto), algo que não ocorreria em projetos reais que passassem por dimensionamentos.

Apresentou-se também uma discrepância e problema na *tag* referente a fiação, a qual a referência é o ponto onde inicia em vez do eletroduto onde estará presente. Isso ocorre, pois, a parametrização das famílias as quais a fiação faz parte está presente nos elementos elétricos e não nos eletrodutos, indicando uma necessidade de adaptação de família e modificações dentro do projeto para obter uma exatidão de projeto. A programação também não apresentou uma adaptação para indicação de fiação (bitola, fase, neutro, terra, entre outros) do(s) circuito(s), sendo necessário indicar nos elementos tais informações manualmente, podendo então a programação apenas realizar a indicação com as *tags*.

Observou-se que a programação dos testes 01, 02 e 03 apresentaram as repetições esperadas, a modelagem, informatização e organização do projeto são tarefas repetitivas, sendo então desenvolvida uma rotina (programação que possibilita a automação de tarefas repetitivas), embora essa rotina não se manteve autônoma, necessitando de atuação manual para sua utilização.

A programação demonstrou ser adaptável na parte de documentação, sendo necessário apenas a inserção de parâmetros e modificação/criação de *tags* em Revit®. A organização também se mostrou de fácil adaptabilidade, funcionando a partir das *tags* da documentação, elas trabalham obtendo a localização bidimensional e tridimensional de elementos, e através de simples equações de soma a localização das *tags* são ajustadas tridimensionalmente para a localização desejada pelo projetista.

A parte da modelagem se mostrou de difícil adaptação, como o Dynamo® trabalha com coordenadas tridimensionais para o desenvolvimento do traçado, os mesmos foram desenvolvidos da maneira mais simples através de linhas guias retas, surgindo assim a dificuldade em realizar adaptações para um traçado mais realista, o qual necessitaria realizar curvas por questões de compatibilização com outras disciplinas, algo que não foi previsto. Para tal previsão seria necessário realizar uma adaptação extrema da programação relacionado a parte da modelagem, onde seria necessário prever a interferência entre elementos de diferentes disciplinas para compatibilização e um referenciamento tridimensional mais complexo dos pontos elétricos para possibilitar a realização de curvas e traçados mais realistas.

Observa-se que embora novas informações fossem inseridas a cada teste, a rotina necessitava apenas de adaptação com relação a novos elementos, possibilitando reaproveitar grupos inteiros, criando assim uma rotina que apresenta a

capacidade de ser utilizada em qualquer projeto elétrico, demandando apenas a adaptação de elementos conforme arquivo rte. ou template (famílias, tags, parametrização, etc) utilizada de projeto, demonstrando por exemplo que os conceitos e as rotinas completas apresentadas nos testes 01, 02 e 03 podem ser utilizadas/adaptadas para qualquer disciplina (não apenas elétrico) de projeto. Caso a programação fosse ser utilizada em outra template ou projeto novo, seria necessário adaptações com relação a nome de parâmetros de famílias, indicação gerais e indicação de famílias para a programação, não sendo necessário alterar a estrutura da programação em si, mas apenas indicar as informações necessárias.

A programação se mostrou interessante para demonstrar as possibilidades da aplicação de programação básica para o desenvolvimento de projetos de engenharia, mostrando compatibilidade entre ambos.

Embora o ideal seria reutilizar a corda de saída de nós iguais para diminuição e otimização da rotina, de forma intencional manteve-se o mais separado possível, de maneira a tornar melhor a visualização da função de cada grupo em anexo e separar o máximo possível a programação, permitindo a utilização apenas do que for interessante ao projetista, assim como apresentado na Figura 38.

Figura 38 – Programação visual final baseada no teste 03 apresentada no Dynamo Player



-  Modelagem PONTOS 230-160 CM - teste 03 - Gustavo Varela Morche - TCC  
 Ready
-  Modelagem PONTOS 30-30 CM - teste 03 - Gustavo Varela Morche - TCC  
 Ready
-  Modelagem PONTOS QUAISQUER - teste 03 - Gustavo Varela Morche - TCC  
 Ready
-  TAGS CONDUÍTES 2D  
 Ready
-  TAGS CONDUÍTES 3D  
 Ready
-  TAGS EQUIPAMENTOS ELÉTRICOS 2D  
 Ready
-  TAGS EQUIPAMENTOS ELÉTRICOS 3D  
 Ready
-  TAGS FIAÇÃO INTERRUPTORES 2D  
 Ready
-  TAGS FIAÇÃO INTERRUPTORES 3D  
 Ready
-  TAGS FIAÇÃO TOMADAS 2D  
 Ready
-  TAGS FIAÇÃO TOMADAS 3D  
 Ready
-  TAGS INTERRUPTORES 2D  
 Ready
-  TAGS INTERRUPTORES 3D  
 Ready
-  TAGS LUMINÁRIAS 2D  
 Ready
-  TAGS LUMINÁRIAS 3D  
 Ready
-  TAGS TOMADAS 2D  
 Ready
-  TAGS TOMADAS 3D  
 Ready

Teste 03 - Elé - PROG.rvt

A programação apresentada no teste 03, sendo a mais completa, poderia ser utilizada nos testes 01 e 02, obtendo os mesmos resultados por ser derivada delas.

## 5.1 ANÁLISE DA PROGRAMAÇÃO CONFORME CONCEITOS ESTIPULADOS

Tabela 01 – Análise de conceitos com base na programação do teste 01

<b>CONCEITOS CONFORME TÓPICO 3.5</b>	<b>ALCANÇADO</b>	<b>NÃO ALCANÇADO</b>
Redução de tempo (1)	X	
Simplificação na execução (2)	X	
Flexibilidade na produção (3)	X	
Controle de desperdícios (4)		NÃO RELEVANTE
Melhoria continua (5)		NÃO RELEVANTE
Super processamento (6)		X
Desperdício intelectual (7)	X	
Visibilidade do processo (8)	X	

Fonte: O autor (2022).

- (1) – Devido a parte de modelagem ser apenas dois pontos elétricos relacionado ao teste 01, o tempo gasto de modelagem foi completamente eliminado; O tempo gasto em documentação foi completamente eliminado, tendo a vantagem de todo elemento presente no modelo ter sido informatizado de forma padronizada; O tempo gasto em organização foi reduzido de forma relativa, não sendo completamente eliminado devido a ajustes de sobreposição e posição.
- (2) – Os processos de modelagem, documentação e organização foram automatizados, sendo necessário pouca interação manual.
- (3) – Após a programação ter sido concluída o processo de desenvolvimento é simples, possibilitando rápidas análises nas necessidades do produto.
- (4) – Como o modelo apresenta apenas um pequeno traçado de eletrodutos, a análise de quantidade não se mostrou interessante.
- (5) – Por ser o primeiro desenvolvimento do primeiro teste, tal análise não se mostrou interessante.
- (6) – A programação não se mostrou autônoma, existindo a necessidade de certa interferência manual.

- (7) – A etapa de modelagem e documentação para o caso específico foi totalmente eliminada, tendo apenas a etapa de organização a necessidade de alterações após utilização da programação e a indicação de pontos.
- (8) – A programação se demonstrou de fácil utilização para a parte de modelagem (necessidade de apenas indicar os pontos através de seleção utilizando o Dynamo Player), não sendo necessária interferências para a documentação e organização para sua utilização. A programação também se mostrou adaptável conforme necessidade, possibilitando a alteração na escolha do que documentar com base nos parâmetros das famílias e como a organização nos modelos bidimensional e tridimensional pode ocorrer.

Tabela 02 – Análise de conceitos com base na programação do teste 02

<b>CONCEITOS CONFORME TÓPICO 3.5</b>	<b>ALCANÇADO</b>	<b>NÃO ALCANÇADO</b>
Redução de tempo (1)	X	
Simplificação na execução (2)	X	
Flexibilidade na produção (3)	X	
Controle de desperdícios (4)		NÃO RELEVANTE
Melhoria continua (5)	X	
Super processamento (6)		X
Desperdício intelectual (7)	X	
Visibilidade do processo (8)	X	

Fonte: O autor (2022).

- (1) – O tempo de modelagem foi praticamente eliminado, obtendo uma modelagem simples independente do layout do projeto elétrico; a documentação do modelo foi completa, acompanhado o desenvolvido manualmente; a organização apresentou o mesmo problema. A programação apenas identificará os elementos, porém não apresentou a capacidade de entender questões de sobreposições e posição.
- (2) – Os processos de modelagem, documentação e organização foram automatizados, sendo necessário pouca interação manual na parte de modelagem (indicação de pontos) e organização (resolução de sobreposição e posição).
- (3) – Após a programação ter sido concluída o processo de desenvolvimento é simples, possibilitando rápidas análises nas necessidades do produto.

- (4) – Como o modelo apresenta apenas três traçados curtos de eletrodutos, a análise de quantidade não se mostrou interessante.
- (5) – No momento que elementos diferentes se fizeram presente no projeto, mostrou-se a necessidade de adaptação com relação a documentação e organização. Quanto mais elementos, mais informações se mostram necessárias, e quanto mais informatizado um mesmo elemento, mais suas informações/maneira de informatizar utilizando a programação se torna difícil. Embora ainda tenha mantido problemas trazidos no teste 01, o teste 02 mostrou-se capaz de se adequar a modelos mais complexos.
- (6) – Não se mostrou capaz desenvolver uma programação visual capaz de ser totalmente autônoma, sendo necessário interferência manual, porém otimizando o trabalho.
- (7) – A etapa de modelagem foi simplificada com relação ao teste 01; a documentação foi totalmente eliminada, se mostrando capaz de informatizar todos os elementos presentes no projeto, tendo apenas a etapa de organização a necessidade de alterações após utilização da programação visual.
- (8) – A programação se demonstrou de mais fácil utilização e compreensão para a parte de modelagem em comparação ao teste 01, não sendo necessária interferências para a documentação e organização para sua utilização. A programação também se mostrou adaptável conforme necessidade, possibilitando a alteração na escolha do que documentar com base nos parâmetros das famílias e como a organização nos modelos bidimensional e tridimensional pode ocorrer, se assemelhando ao teste 01.

Tabela 03 – Análise de conceitos com base na programação do teste 03

<b>CONCEITOS CONFORME TÓPICO 3.5</b>	<b>ALCANÇADO</b>	<b>NÃO ALCANÇADO</b>
Redução de tempo (1)	X	
Simplificação na execução (2)	X	
Flexibilidade na produção (3)	X	
Controle de desperdícios (4)		X
Melhoria continua (5)	X	
Super processamento (6)		X
Desperdício intelectual (7)	P	P
Visibilidade do processo (8)	X	

Fonte: O autor (2022).

**OBSERVAÇÃO Tabela 03:**

P = resultado obtido parcialmente.

- (1) – O tempo de modelagem foi reduzido, porém não eliminado, devido a necessidade de pontos serem selecionados para o desenvolvimento; o tempo de documentação foi totalmente eliminado, sendo o tempo gasto conforme tamanho de projeto e capacidade do computador; devido a alta quantidade de informações, a organização se tornou complexa, criando muitas sobreposições e elementos muitas vezes desnecessários (várias *tags* de diâmetro) que necessitariam de grande ajuste manual. A programação apenas identificou os elementos, porém não apresentou a capacidade de entender questões de sobreposições e posição, principalmente devido a proximidade e quantidade de elementos.
- (2) – Os processos de modelagem, documentação e organização foram totalmente simplificados, sendo necessário interação manual relativamente alta na parte de modelagem (indicação de pontos) e organização (resolução de sobreposição e posição), demonstrando a falta de autonomia do sistema.
- (3) – Após a programação ter sido concluída o processo de desenvolvimento se mostrou simples, conforme apresentado na Figura 38, todo o processo é realizado com simples cliques, sendo dividido conforme interesse/necessidade.
- (4) – Análise de quantitativo conforme Tabela 04:

Tabela 04 – Quantitativo extraído do Revit® - modelo com base na programação do teste 03

<QUANTITATIVO DE ELETRODUTOS - MODELO DESENVOLVIDO COM A PROGRAMAÇÃO VISUAL>		
A	B	C
Descrição do Material	Diâmetro Nominal	Comprimento (m)
Eletroduto flexível corrugado Reforçado, em PVC na cor laranja	Ø25	141,02 m
Eletroduto flexível corrugado, em PVC na cor amarelo	Ø25	125,62 m

Fonte: O autor (2022).

Tabela 05 – Quantitativo extraído do Revit® - modelo com base no projeto teste 03 desenvolvido manualmente

<QUANTITATIVO DE ELETRODUTOS - MODELO DESENVOLVIDO MANUALMENTE>		
A	B	C
Descrição do Material	Diâmetro Nominal	Comprimento (m)
Eletroduto flexível corrugado Reforçado, em PVC na cor laranja	Ø25	102,97 m
Eletroduto flexível corrugado Reforçado, em PVC na cor laranja	Ø20	60,93 m
Eletroduto flexível corrugado, em PVC na cor amarelo	Ø25	19,41 m
Eletroduto flexível corrugado, em PVC na cor amarelo	Ø20	54,71 m

Fonte: O autor (2022).

Embora em ambas as versões apresentadas no teste 03 se observa claras oportunidades de otimizações (como a utilização de um mesmo conduíte para passar mais de um circuito), as modelagens manuais e com programação foram desenvolvidas de maneira a conectar apenas elementos do próprio circuito, para se aproximar de uma comparação de materiais.

Conforme apresentado na Tabela 04, o desenvolvimento com a programação necessitou de 141,02 m de eletroduto laranja e 125,62 m de eletroduto amarelo (totalizando 266,64 m), em comparação a 163,90 m de eletroduto laranja e 74,12 m de eletroduto amarelo (totalizando 238,02 m) utilizado no desenvolvimento manual conforme apresentado na Tabela 05.

O desenvolvimento com a programação apresentou um gasto maior de eletrodutos de aproximadamente 11,0 % ou 28,62 m. O motivo disso ter acontecido pode ser observado na padronização de modelagem apresentada na Figura 38, a qual não apresenta mais possibilidades de desenvolvimento da modelagem, ou uma maneira única, a qual consiga se adaptar a todos os casos.

- (5) – O teste 03 apresentou melhorias significativas com relação a facilidade de utilização por parte do usuário, mostrou-se capaz de se adequar a modelos complexos até certo ponto, demonstrando que quanto maior o modelo e quanto mais ponto é necessário o desenvolvimento de novos elementos. No entanto, a modelagem se mostrou inferior e menos realista em comparação a modelagem desenvolvida manualmente.
- (6) – Devido a complexibilidade do teste 03 em comparação aos testes 01 e 02, a programação se tornou dependente de trabalho manual na parte de modelagem. Com relação a parte de organização, seria necessária toda uma

reestruturação da organização com relação ao apresentado nas Figuras 34 e 37 para possibilitar fácil visualização e compreensão das informações em modelo.

- (7) – Apenas partes do processo foi otimizado, como a possibilidade da informatização de todos os elementos em projeto e a otimização da modelagem em alguns pontos, porém, ao mesmo tempo, a programação visual criou problemas na parte de organização os quais criariam um desperdício para serem ajustados manualmente,
- (8) – A programação se mostrou de fácil utilização para o usuário, mesmo sendo leigo na parte de programação visual, a utilização através do Dynamo Player possibilita uma programação visual de fácil utilização.

## 5.2 ANÁLISE DE RESULTADOS CONFORME CONCEITOS ESTIPULADOS

A programação visual desenvolvida visou basicamente eliminar diversos desperdícios trazidos pelo *Lean* através de uma abordagem BIM.

Considerando o produto final apresentado na Figura 38 e os 8 conceitos estipulados no tópico 4.4.1, observa-se que:

(1) a programação auxiliou na redução de tempo em etapas específicas de desenvolvimento de projeto, os quais não necessitam de raciocínio lógico; com relação ao conceito;

(2) eliminou-se etapas no desenvolvimento, porém apenas de forma parcial, ainda dependendo de auxílio manual, já com relação ao conceito

(3) o processo de desenvolvimento através da seleção de pontos ou utilização da rotina para documentação e organização se mostrou um processo simples, porém incompleto com relação a melhoria do produto;

(4) a variação da quantidade de materiais com relação ao desenvolvimento manual embora pequena se mostrou menos otimizada, não atendendo ao interesse em obter o melhor traçado através da eliminação de erro humano.

(5) o novo modelo, exceto com relação a padronização de organização, não mostrou melhoria na qualidade, não atendendo ao interesse de apresentar um produto de igual ou melhor qualidade.

(6) devido a questão de organização, mostrou-se a necessidade de grande trabalho manual para organizar as informações sobrepostas ou fora de posição, mostrando assim a dependência da programação visual do trabalho manual.

(7) a etapa de informatizar o projeto foi totalmente eliminada conforme necessidades de informações estipuladas, no entanto enquanto etapas de indicação de informações foram eliminadas, novas de organização foram criadas, atendendo parcialmente o objetivado neste tópico.

(8) Devido a característica *open source* do Dynamo® e a facilidade de utilização no Dynamo Player, terceiros poderiam utilizar a programação visual apresentada conforme necessidade, tendo suas informações dentro do Dynamo totalmente abertas e visuais.

Observa-se que dos 8 conceitos estipulados, dois conceitos foram atendidos (1 e 8), três foram parcialmente atendidos (2, 3 e 7) e três foram pouco ou não atendidos (4, 5 e 6).

## 6 CONCLUSÃO

A programação visual desenvolvida apresentou no final uma capacidade de LOD 300, gerando benefícios parciais devido a criação de outros problemas após sua utilização. Dos objetivos específicos estipulados, a programação visual final apresentada na Figura 38 foi capaz de desenvolver a modelagem, documentação e organização de projetos, no entanto, através da análise dos 8 conceitos observados no tópico 4.4.2, chega-se à conclusão que os objetivos relacionados a modelagem e organização foram atendidos de forma parcial, enquanto apenas o objetivo relacionado a documentação/informatização do projeto foi atendido completamente considerando a adaptabilidade de *tags* e parâmetros de família que o Revit® apresenta.

Com relação a compatibilidade entre BIM e *Lean Construction*, a utilização de BIM apresentou a capacidade de visualizar com facilidade os desperdícios em projetos capazes de serem previstos pelo *Lean Construction*, mesmo se os desperdícios forem criados pelo próprio processo BIM, como apontado no tópico 4.4.2, apresentando boa compatibilidade entre ambos os conceitos.

Conclui-se então que a união de BIM e *Lean Construction* para o desenvolvimento de projetos é benéfica devido ao *Lean* tornar claro os desperdícios que podem ocorrer em processos de desenvolvimento, enquanto o BIM possui elementos e ferramentas (modelagem tridimensional, informatização e organização de elementos de projeto e programação visual) para observar reduzir e/ou eliminar tais desperdícios.

## 7 RECOMENDAÇÕES DE TRABALHOS FUTUROS

Observou-se em vários pontos do desenvolvimento a necessidade de melhoria com relação a autonomia da programação visual, eliminando também o malefício da criação de desperdícios pela mesma, recomenda-se o estudo de trabalhos sobre melhorias na parte de automação de projetos BIM.

Embora tenha sido apresentado a possibilidade, esta monografia não utilizou programação escrita através de nós de IronPython. Devido a maleabilidade da programação escrita, o desenvolvimento da programação visual com o auxílio da utilização de IronPython se torna interessante para atacar os pontos de difícil desenvolvimento com programação visual, objetivando atender os oito conceitos *Lean* analisados. Recomenda-se então o estudo do desenvolvimento de projetos de engenharia civil através da utilização de programação escrita.

Esta monografia utilizou como base projetos elétricos, recomenda-se a realização de um estudo com as mesmas características apresentadas nesta monografia porém através da utilização de projetos hidrossanitários, estruturais ou de outra disciplina de interesse que possibilite a análise da relação entre BIM e *Lean Construction*.

## 8 REFERÊNCIAS

ARNAL, I. P. **Why don't we start at the beginning? The Basic of a Project: Lean Planning and Pre-Construction.** Disponível em: <<https://www.bimcommunity.com/news/load/490/why-don-t-we-start-at-the-beginning>>. Acesso em: 27 mar. 2022.

AUTODESK. **Design and build with BIM.** Disponível em: <<https://www.autodesk.com/industry/aec/bim>>. Acesso em: 31 mar. 2022.

AUTODESK. **Generative Design Primer.** Disponível em: <<https://www.generativedesign.org/>>. Acesso em: 28 mar. 2022.

AUTODESK. **Principais conceitos do Revit.** Disponível em: <<https://knowledge.autodesk.com/pt-br/support/revit/learn/caas/qsarticles/key-revit-concepts.html>>. Acesso em: 18 jul. 2022.

AUTODESK. **Revit®.** Disponível em: <[https://www.autodesk.com.br/products/Revit\(R\)/overview](https://www.autodesk.com.br/products/Revit(R)/overview)>. Acesso em: 16 mar. 2022.

AUTODESK. **Revit Developer Center.** Disponível em: <<https://www.autodesk.com/developer-network/platform-technologies/revit>>. Acesso em: 30 jun. 2022.

AUTODESK. **Tarefas básicas: MEP.** Disponível em: <<https://knowledge.autodesk.com/pt-br/support/revit/learn/caas/qsarticles/basic-tasks-mep.html>>. Acesso em: 18 jul. 2022.

BRANDSTETTER, M. C. G. O. ; RIBEIRO, H. R. O. **Causas de custos adicionais e impacto financeiro em obras públicas sob a perspectiva da gestão de risco.** Disponível em: <<https://www.scielo.br/j/ac/a/Pvj7Mp9nTWSj9XFWMGDJtzS/?lang=pt>>. Acesso em: 06 dez. 2021.

BIM FORUM. **Level of Development Specification.** Disponível em: <<https://bimforum.org/lof/>>. Acesso em: 31 jul. 2022.

BIM FORUM. **LEVEL OF DEVELOPMENT SPECIFICATION For Building Information Modeling PART I, GUIDE, & COMMENTARY December 2021.** Disponível em: <<https://bimforum.org/wp-content/uploads/2022/02/LOD-Spec-2021-Part-I-FINAL-2021-12-28.pdf>>. Acesso em: 31 jul. 2022.

BUILDINGS. **Autodesk Completes Acquisition of Revit Technology Corporation.** Disponível em: <<https://www.buildings.com/news/article/10195475/autodesk-completes-acquisition-of-revit-technology-corporation>>. Acesso em: 18 jul. 2022.

EASTMAN, C. ; TEICHOZ, P. ; SACKS, R. ; LISTON, K. **Manual de BIM: um guia de modelagem da informações da construção para arquitetos, engenheiros, gerentes, construtores e incorporadores.** Porto Alegre: Bookman, 2014, 155 – 157 p. 309 p. 483 p.

HAMIL, S. **What are BIM dimensions?**. Disponível em: <https://www.thenbs.com/knowledge/bim-dimensions-3d-4d-5d-6d-bim-explained>. Acesso em: 31 mar. 2022.

INTERNATIONAL BUSINESS MACHINE (IBM). **Application Programming Interface (API)**. Disponível em: <https://www.ibm.com/cloud/learn/api#:~:text=An%20API%20is%20a%20set,processes%20data%20transfer%20between%20systems>. Acesso em: 31 mar. 2022.

INTERNATIONAL ORGANIZATION FOR STANDARDIZATION. **ISO 1950-1: Organization and digitization of information about buildings and civil engineering works, including building information modelling (BIM) — Information management using building information modelling — Part 1: Concepts and principles.** Geneva, 2018. 34 p.

ISO. **ISO 19650-5: 2020**. Disponível em: <https://www.iso.org/standard/74206.html>. Acesso em: 31 mar. 2022.

ISO. **ISO 19650-1: 2018**. Disponível em: <https://www.iso.org/standard/68078.html>. Acesso em: 29 jul. 2022.

JEZYK, MATT. **The Dynamo Primer**. Disponível em: <https://primer.dynamobim.org/>. Acesso em: 15 mar. 2022.

KOSKELA, L. **Application of the new production philosophy to construction**. Disponível em: <https://cife.stanford.edu/application-new-production-philosophy-construction>. Acesso em: 28 mar. 2022.

LEAN CONSTRUCTION INSTITUTE. **Getting started with lean**. Disponível em: <https://leanconstruction.org/pages/learning/getting-started-with-lean/>. Acesso em: 28 mar. 2022.

LEAN ENTERPRISE INSTITUTE. **A Brief History of Lean**. Disponível em: <https://www.lean.org/explore-lean/a-brief-history-of-lean/>. Acesso em: 28 mar. 2022.

LEAN ENTERPRISE INSTITUTE. **What is Lean?**. Disponível em: <https://www.lean.org/explore-lean/what-is-lean/>. Acesso em: 28 mar. 2022.

LEAN INSTITUTE BRASIL. **Escritório Enxuto (Lean Office)**. Disponível em: [https://www.lean.org.br/artigos/57/escritorio-enxuto-\(lean-office\).aspx](https://www.lean.org.br/artigos/57/escritorio-enxuto-(lean-office).aspx). Acesso em: 05 jul. 2022.

MARIA, R. N. ; PICCHI, F. A. **Implementation of lean practices facilitated by BIM functionalities in the construction phase: advances and opportunities.** Disponível em: <<https://www.scielo.br/j/ac/a/n9dwqZqXSZTYCjVzs5JXTWB/>>. Acesso em: 27 mar. 2022.

MAYR, L. R. **Falhas de projetos e erros de execução: uma questão de comunicação.** Disponível em: <[MONTOYA, M. D. A. ; GARRIGÓS, A. G. ; MAESTRE, C. R. ; IRIBARREN, V. E. \*\*Bim and lean construction interactions: a state-of-the-art review.\*\* Disponível em: <<https://www.witpress.com/elibrary/wit-transactions-on-the-built-environment/192/37527>>. Acesso em: 27 mar. 2022.](https://repositorio.ufsc.br/bitstream/handle/123456789/79218/177370.pdf?sequence=)

MOSSMAN, A. **What is lean construction: another look – 2018.** Disponível em: <[https://www.researchgate.net/profile/Alan-Mossman/publication/326466614\\_What\\_Is\\_Lean\\_Construction\\_Another\\_Look\\_-\\_2018/links/5b5994dfa6fdccf0b2f8502d/What-Is-Lean-Construction-Another-Look-2018.pdf](https://www.researchgate.net/profile/Alan-Mossman/publication/326466614_What_Is_Lean_Construction_Another_Look_-_2018/links/5b5994dfa6fdccf0b2f8502d/What-Is-Lean-Construction-Another-Look-2018.pdf)>. Acesso em: 07 dez. 2021.

NAGY, D. **The Problem of Learning.** Disponível em: <<https://medium.com/generative-design/generative-design-introduction-64fb2db38e1>>. Acesso em: 07 dez. 2021.

NATSPEC BIM. **BIM and LOD.** Disponível em: <[https://bim.natspec.org/images/NATSPEC\\_Documents/NATSPEC\\_BIM\\_LOD\\_Paper\\_131115.pdf](https://bim.natspec.org/images/NATSPEC_Documents/NATSPEC_BIM_LOD_Paper_131115.pdf)>. Acesso em: 30 jul. 2021.

.NET FOUNDATION. **IronPython.** Disponível em: <<https://ironpython.net/documentation/dotnet/>>. Acesso em: 16 mar. 2022.

PEREIRA, L, M. et al. **ESTUDO EXPLORATÓRIO COMPARATIVO DA EFICÁCIA ENTRE PROTÓTIPOS FÍSICO, ANALÍTICO 2D E 3D NA IDENTIFICAÇÃO DE INCONSISTÊNCIAS DE PROJETOS.** Disponível em: <[https://redib.org/Record/oai\\_articulo2390748-estudo-explorat%C3%B3rio-comparativo-da-efic%C3%A1cia-entre-prot%C3%B3tipos-f%C3%ADsico-anal%C3%ADtico-2d-e-3d-na-identifica%C3%A7%C3%A3o-de-inconsist%C3%A2ncias-de-projetos](https://redib.org/Record/oai_articulo2390748-estudo-explorat%C3%B3rio-comparativo-da-efic%C3%A1cia-entre-prot%C3%B3tipos-f%C3%ADsico-anal%C3%ADtico-2d-e-3d-na-identifica%C3%A7%C3%A3o-de-inconsist%C3%A2ncias-de-projetos)>. Acesso em: 05 jul. 2022. Gestão e Tecnologia de Projetos, volume 10, p. 29 – 47, 2015.

Harding, J. ; Shepherd, P. **Meta-Parametric Design.** Disponível em: <[https://www.researchgate.net/publication/309339993\\_Meta-Parametric\\_Design](https://www.researchgate.net/publication/309339993_Meta-Parametric_Design)>. Acesso em: 06 jul. 2022.

TALARICO, G. **Revit API Code Samples BETA.** Disponível em: <<https://www.revitapidocs.com/code/>>. Acesso em: 30 jun. 2022.

TAVARES, P. R. F. **Ferramentas de apoio à implementação lean construction em projetos.** Disponível em: <[https://bibliotecadigital.ipb.pt/bitstream/10198/22991/1/Tavares\\_Pedro.pdf](https://bibliotecadigital.ipb.pt/bitstream/10198/22991/1/Tavares_Pedro.pdf)>. Acesso em: 28 mar. 2022.

UK BIM FRAMEWORK. **The BIM process.** Disponível em: <<https://www.ukbimframework.org/>>. Acesso em: 31 mar. 2022.

VASCONCELOS, A. N. S. **A influência da tecnologia bim na geração minimizada e gestão de resíduos sólidos da indústria de construção civil.** Disponível em: <<https://repositorio.ufc.br/handle/riufc/47022>>. Acesso em: 22 mar. 2022.

## 9 APÊNDICES

9.1 APÊNDICE I (APÊNDICES A A W) – QR CODE DE ACESSO A UMA PASTA DO GOOGLE DRIVE PARA FACILITAÇÃO NA VISUALIZAÇÃO DE GRUPOS E NÓS UTILIZADOS



9.2 APÊNDICE II – QR CODE DE ACESSO A UMA PASTA DO GOOGLE DRIVE PARA VISUALIZAÇÃO DE VÍDEOS DE DESENVOLVIMENTO MANUAL E COM PROGRAMAÇÃO

