



**UNIVERSIDADE FEDERAL DO CEARÁ**  
**CENTRO DE TECNOLOGIA**  
**DEPARTAMENTO DE ENGENHARIA ESTRUTURAL E CONSTRUÇÃO CIVIL**  
**PROGRAMA DE PÓS-GRADUAÇÃO EM ENGENHARIA CIVIL: ESTRUTURAS E**  
**CONSTRUÇÃO CIVIL**

**YAN MOTA VERAS DE CARVALHO**

**FERRAMENTA BIM DE ANÁLISE AUTOMATIZADA DE PROJETOS**

**FORTALEZA**

**2021**

YAN MOTA VERAS DE CARVALHO

FERRAMENTA BIM DE ANÁLISE AUTOMATIZADA DE PROJETOS

Dissertação apresentada ao Programa de Pós-Graduação em Engenharia Civil da Universidade Federal do Ceará, como requisito parcial à obtenção do título de Mestre em Estruturas e Construção Civil. Área de concentração: Construção Civil.

Orientador: Prof. Dr. José de Paula Barros Neto.  
Coorientadora: Profa. Dra. Mariana M. Xavier de Lima.

FORTALEZA

2021

Dados Internacionais de Catalogação na Publicação  
Universidade Federal do Ceará  
Biblioteca Universitária

Gerada automaticamente pelo módulo Catalog, mediante os dados fornecidos pelo(a) autor(a)

---

- V584f Veras de Carvalho, Yan Mota.  
FERRAMENTA BIM DE ANÁLISE AUTOMATIZADA DE PROJETOS / Yan Mota Veras de  
Carvalho. – 2021.  
126 f. : il. color.
- Dissertação (mestrado) – Universidade Federal do Ceará, Centro de Tecnologia, Programa de Pós-  
Graduação em Engenharia Civil: Estruturas e Construção Civil, Fortaleza, 2021.  
Orientação: Prof. Dr. José de Paula Barros Neto.  
Coorientação: Profa. Dra. Mariana Monteiro Xavier de Lima.
1. BIM Model Checking. 2. Avaliação de Projetos. 3. Programação Visual. I. Título.

CDD 624.1

---

YAN MOTA VERAS DE CARVALHO

FERRAMENTA DE ANÁLISE AUTOMATIZADA DE PROJETOS

Dissertação apresentada ao Programa de Pós-Graduação em Engenharia Civil da Universidade Federal do Ceará, como requisito parcial à obtenção do título de Mestre em Estruturas e Construção Civil. Área de concentração: Construção Civil.

---

BANCA EXAMINADORA

---

---

Prof. Dr. José de Paula Barros Neto (Orientador)  
Universidade Federal do Ceará (UFC)

---

Prof. Dra. Mariana Monteiro Xavier de Lima (Coorientadora)  
Universidade Federal do Ceará (UFC)

---

Prof. Dr. Alexandre Araújo Bertini  
Universidade Federal do Ceará (UFC)

---

Prof. Dr. Sergio Scheer  
Universidade Federal do Paraná (UFPR)

---

Prof. Dr. Daniel Ribeiro Cardoso  
Universidade Federal do Ceará (UFC)

**Dedicado a  
minha mãe, base e fortaleza  
Iêda  
e a memória e os ensinamentos de meu pai  
Eudes**

## RESUMO

O processo de projeto BIM abre possibilidades de utilização das informações contidas nos modelos para aprimoramento dos projetos. Uma das vertentes de aplicação do BIM é a possibilidade de avaliar e checar projetos. Este trabalho propõe a utilização de ferramentas de programação visual para automatizar a checagem de desempenhos do projeto, com o intuito de melhorar o processo de projeto. A pesquisa se desenvolve seguindo o delineamento metodológico de Design Science Research composto das seguintes etapas: identificação do estado da arte da avaliação automatizada de modelos, identificação e seleção dos requisitos de desempenho a serem avaliados, transformação e codificação das regras, testes e projeto piloto, e apresentação de um Modelo para desenvolvimento de critérios de avaliação. O artefato desenvolvido em programação visual por meio da integração do Dynamo com o Revit avalia modelos por meio de 43 critérios automatizados, em 4 dimensões: Construtibilidade, Sustentabilidade, Flexibilidade e Conformidade. Baseado neste artefato apresenta-se um modelo com atividades, ferramentas de classificação e definições processuais que generalizam o método aplicado.

**Palavras-chave:** Avaliação de Projetos. BIM Model Checking. Programação Visual.

## ABSTRACT

The BIM design process reveal possibilities to use information contained in the models to improve designs. One of the main aspects of applying BIM is the ability to evaluate and check designs. This work proposes the use of visual programming languages to automate the design performance checking, to improve the design process. The research is developed following the Design Science Research composed of the following steps: identification of the state of the art for automated evaluation of BIM models; selection of performance criteria to be evaluated; transformation and coding of rules; tests and pilot project; and framework proposal for evaluation criteria development. The artifact developed in visual programming through the integration of Dynamo with Revit evaluates models through 43 automated criteria, in 4 dimensions: Constructability, Sustainability, Flexibility and Conformity. Based on this artifact, a framework is presented with activities, classification tools and procedural definitions that generalize the applied method.

**Keywords:** Design Evaluation. BIM Model Checking. Visual Programming

## SUMÁRIO

<b>1</b>	<b>INTRODUÇÃO .....</b>	<b>1</b>
<b>1.1</b>	<b>JUSTIFICATIVA .....</b>	<b>1</b>
<b>1.2</b>	<b>Problema de Pesquisa.....</b>	<b>3</b>
<b>1.3</b>	<b>Objetivos .....</b>	<b>4</b>
<i>1.3.1</i>	<i>Objetivo Geral .....</i>	<i>4</i>
<i>1.3.2</i>	<i>Objetivos Específicos.....</i>	<i>4</i>
<b>2</b>	<b>REFERENCIAL TEÓRICO .....</b>	<b>6</b>
<b>2.1</b>	<b>Building Information Modeling .....</b>	<b>6</b>
<i>2.1.1</i>	<i>Modelagem Paramétrica .....</i>	<i>7</i>
<i>2.1.2</i>	<i>BIM ao longo do ciclo de vida do projeto .....</i>	<i>8</i>
<b>2.2</b>	<b>Avaliação Automatizada de Modelos.....</b>	<b>10</b>
<i>2.2.1</i>	<i>Processo de avaliação .....</i>	<i>12</i>
<i>2.2.2</i>	<i>Classificação das regras .....</i>	<i>13</i>
<i>2.2.3</i>	<i>Transformação de regras .....</i>	<i>16</i>
<i>2.2.4</i>	<i>Implementação.....</i>	<i>22</i>
<b>2.3</b>	<b>Aplicação dos sistemas de checagem.....</b>	<b>27</b>
<b>3</b>	<b>METODOLOGIA.....</b>	<b>30</b>
<b>3.1</b>	<b>Estratégia de Pesquisa.....</b>	<b>30</b>
<b>3.2</b>	<b>Delineamento da Pesquisa.....</b>	<b>32</b>
<b>3.3</b>	<b>Etapas Metodológicas.....</b>	<b>33</b>
<i>3.3.1</i>	<i>Conscientização.....</i>	<i>33</i>
<i>3.3.2</i>	<i>Sugestão.....</i>	<i>35</i>
<i>3.3.3</i>	<i>Desenvolvimento.....</i>	<i>35</i>
<i>3.3.4</i>	<i>Avaliação .....</i>	<i>37</i>
<i>3.3.5</i>	<i>Conclusão .....</i>	<i>37</i>
<b>3.4</b>	<b>Delineamento geral .....</b>	<b>38</b>
<b>4</b>	<b>RESULTADOS .....</b>	<b>41</b>
<b>4.1</b>	<b>Conscientização: Levantamento dos Critérios.....</b>	<b>41</b>
<i>4.1.1</i>	<i>Revisão Sistemática: Métricas de Avaliação de Projeto.....</i>	<i>41</i>
<i>4.1.2</i>	<i>Consulta aos especialistas.....</i>	<i>44</i>
<b>4.2</b>	<b>Protótipo: Construtibilidade .....</b>	<b>45</b>



4.2.1	<i>Conceituação</i> .....	45
4.2.2	<i>Sugestão de Indicadores</i> .....	47
4.2.3	<i>Implementação</i> .....	48
4.2.4	<i>Avaliação</i> .....	49
4.2.5	<i>Avaliação como protótipo</i> .....	51
<b>4.3</b>	<b>Desenvolvimento</b> .....	<b>52</b>
4.3.1	<i>Medir</i> .....	52
4.3.2	<i>Implementar</i> .....	64
<b>4.4</b>	<b>Avaliação</b> .....	<b>72</b>
<b>4.5</b>	<b>Proposição de Modelo de Desenvolvimento de Critério de Avaliação de Projeto Automatizados</b> .....	<b>76</b>
4.5.1	<i>Definição</i> .....	77
4.5.2	<i>Desenvolvimento</i> .....	78
4.5.3	<i>Avaliação</i> .....	82
<b>5</b>	<b>CONCLUSÕES</b> .....	<b>84</b>
	<b>REFERÊNCIAS</b> .....	<b>87</b>
	<b>APÊNDICE</b> .....	<b>96</b>
	<b>Construtibilidade</b> .....	<b>96</b>
	<b>Flexibilidade</b> .....	<b>103</b>
	<b>Sustentabilidade</b> .....	<b>106</b>
	<b>Conformidade</b> .....	<b>110</b>

## Sumário de Figuras

Figura 1 Hierarquia de elementos no Revit.....	8
Figura 2 Etapas da verificação de regras .....	13
Figura 3 Método Tx3 de transformação de regras.....	17
Figura 4 Operadores RASE .....	17
Figura 5 Classificação das Regras no Tx3 .....	20
Figura 6 Funcionamento do TIO .....	21
Figura 7 Estrutura de um nó no Autodesk® Dynamo .....	26
Figura 8 Diagrama da metodologia geral em Design Science Research.....	32
Figura 9 Método adotado para a RSL.....	34
Figura 10 Resumo do Método .....	38
Figura 11 Programa de avaliação da construtibilidade.....	49
Figura 12 Projeto 1 .....	50
Figura 13 Projeto 2 .....	50
Figura 14 Classificação dos critérios informações X complexidade.....	54
Figura 15 Classificação Geral dos Critérios .....	61
Figura 16 Classificação dos Critérios Por Categoria.....	63
Figura 17 Classificação Discriminada por Dimensão .....	63
Figura 18 Critério Razão e Orientação de Janelas.....	66
Figura 19 Critério Razão e Orientação de Janelas - 1 .....	66
Figura 20 Critério Razão e Orientação de Janelas - 2 .....	67
Figura 21 Critério Razão e Orientação de Janelas - 3 .....	68
Figura 22 Critério Razão e Orientação de Janelas - 4 .....	68
Figura 23 Critério Razão e Orientação de Janelas - Visualização.....	69
Figura 24 Lista de Programas no Dynamo Player.....	71
Figura 25 Exibição de Programa no Dynamo Player .....	71
Figura 26 Exemplo de visualização de geometria Dynamo no Revit - Pé Direito.....	72
Figura 27 Modelos Utilizados nos Testes .....	73
Figura 28 Verificação dos Recuos.....	75
Figura 29 Variação do Modelo.....	75
Figura 30 Interação e Etapas do Modelo de Desenvolvimento.....	76
Figura 31 Modelo de Desenvolvimento de Critério de Avaliação de Projeto Automatizados.....	76
Figura 32 Detalhamento do modelo – Definição.....	77

Figura 33 Detalhamento do modelo - Desenvolvimento.....	79
Figura 34 Geometria do Critério de Rota de Fuga (a) rota de fuga (b).....	81
Figura 35 Detalhamento do modelo - Avaliação.....	82

## Sumário de Tabelas

Tabela 1 Usos do BIM nas etapas do projeto .....	9
Tabela 2 Conceitos dos modelos de checagem .....	15
Tabela 3 Exemplo de aplicação RASE.....	18
Tabela 4 Exemplo de resultado da aplicação do RASE .....	18
Tabela 5 Dicionário TIO para metas qualitativas transformadas em quantitativas.....	22
Tabela 6 Funções dos Softwares comerciais usados na checagem .....	23
Tabela 7 Síntese do Método .....	39
Tabela 8 Artigos analisados.....	42
Tabela 9 Perfil dos Especialistas .....	44
Tabela 10 Indicadores para construtibilidade.....	47
Tabela 11 Resultados dos indicadores de construtibilidade .....	51
Tabela 12 Critérios de Construtibilidade.....	55
Tabela 13 Critérios de Flexibilidade .....	57
Tabela 14 Critérios de Sustentabilidade .....	58
Tabela 15 Critérios de Conformidade .....	59
Tabela 16 Modelo de Tabela Resumo .....	69
Tabela 17 Resumo Implementação - Orientação de Aberturas .....	70
Tabela 18 Síntese dos resultados de avaliação .....	73

## 1 INTRODUÇÃO

A evolução da Ciência da Computação tem mudado o processo de produção de diversas indústrias, a ponto de, hoje, a comunicação sem fio, o monitoramento móvel e a documentação eletrônica fazerem parte de seu modelo de negócios destas empresas (CHAN, 2014). Building Information Modelling (BIM) é a expressão atual de inovação técnica e processual na indústria da construção, pode ser definido como um conjunto interativo de políticas, processos e tecnologias (SUCCAR, 2013). A adoção do BIM está se tornando um assunto cada vez mais importante para a indústria da construção que vem enfrentando barreiras e desafios para aumentar a produtividade, a eficiência, a qualidade e a sustentabilidade (ARAYICI *et al.*, 2011).

Entre os benefícios esperados do BIM estão: aumentar o desempenho da edificação projetada e assegurar a sua conformidade, através de ferramentas de projeto e análise contínua do modelo baseados em BIM para melhorar o desempenho geral da edificação (EASTMAN, Chuck *et al.*, 2011). Atualmente, o avanço computacional, tanto em nível de *software* quanto de *hardware*, tem possibilitado o acesso a ferramentas que automatizam a coleta e o processamento de dados para a avaliação (LIMA, 2016).

Para medir o desempenho são formuladas métricas para a edificação, estas devem ser medidas de qualidade para edifícios, podendo ser usadas para estudos comparativos, para orientar o projeto ou para comparar um edifício com um conjunto de outros edifícios (REINHART; MARDALJEVIC; ROGERS, 2006). Pois, segundo Lima (2016) “ao visar à melhoria da qualidade habitacional, há de se considerar a melhoria do desempenho da edificação como um todo”.

Atualmente grandes investimentos de capital em soluções digitais pioneiras, estão estabelecendo tecnologias futuras em toda a cadeia de valor da construção, indicando uma forte mudança no sentido de sua adoção (HEIGERMOSER *et al.*, 2019). O modelo BIM inclui geometria, relações espaciais, informações geográficas, quantidades e propriedades de componentes de construção (EL-OMARI; MOSELHI, 2011), funcionando como uma base de dados que pode ser analisada. Desenvolver ferramentas que utilizem esses dados para aperfeiçoar o desenvolvimento e execução de projetos de arquitetura e engenharia são desafios atuais para a área.

### 1.1 JUSTIFICATIVA

À medida que aumenta a complexidade do ambiente construído, aumenta a necessidade de profissionais especializados em trabalhar com cada uma das muitas questões

envolvidas no projeto do edifício (LIMA, 2016). O atual processo de projeto se torna mais complexo a cada dia, devido à introdução de novas tecnologias de construção, resultados de pesquisa e códigos de construção cada vez mais rigorosos. Esses desafios exigem uma intensa comunicação entre os participantes do projeto e uma avaliação profunda do projeto do edifício desde os primeiros estágios do processo de projeto (PAUWELS *et al.*, 2011).

O BIM pode ser utilizado para integrar o conhecimento especializado em etapas iniciais do projeto. Pois, se mostra uma ferramenta para visualizar e coordenar o trabalho de AEC (arquitetura, engenharia e construção), evitando erros e omissões, melhorando a produtividade e facilitando planejamento, segurança, orçamento e gerenciamento de qualidade em projetos de construção (CHEN; LUO, 2014).

O advento das aplicações computacionais que suportam a modelagem de objetos 3D de edifícios, admite a criação paramétrica de projetos que respondem a vários critérios permitindo ao projetista a possibilidade de explorar diferentes alternativas de soluções de projeto de forma rápida e segura (ANDRADE; RUSCHEL, 2009). Estes modelos interpretáveis por computador podem ser verificados automaticamente após serem gerados (EASTMAN, C. *et al.*, 2009). Pesquisadores em todo o mundo visam dar um passo importante no desenvolvimento de métodos e tecnologias auxiliadas por computador para automatizar aspectos abrangentes dos processos de verificação, Com o auxílio de tais ferramentas, o esforço manual necessário para a verificação da conformidade a normas e códigos pode ser drasticamente reduzido (PREIDEL; DAUM; BORRMANN, 2017). Estes sistemas aplicam regras, restrições ou condições a um projeto proposto, como resultados pode-se "aprovar", "reprovar" ou , ainda, pontuar características de um projeto (EASTMAN, C. *et al.*, 2009).

Até recentemente, os únicos meios para lidar com esse corpo complexo e crescente de conhecimento eram os processos de cognição humana e revisão organizacional, embora alguns critérios envolvessem análises computadorizadas (por exemplo, estruturas), ainda assim, os detalhes dos fatores de segurança e a interpretação dos resultados têm sido uma tarefa manual e centrada nas pessoas (PREIDEL; BORRMANN, 2016). A abordagem de verificação manual predominante hoje é baseada principalmente em projetos de construção (desenhos técnicos bidimensionais) e documentos textuais adicionais. Convencionalmente, a avaliação do design é executada manualmente por vários especialistas específicos do domínio, um processo demorado, dispendioso e sujeito a erros (ZHANG, S. *et al.*, 2013). Em muitos

casos, ciclos de iteração indesejados tornam-se necessários devido a modificações exigidas pelas respectivas autoridades (PREIDEL; BORRMANN, 2015).

A verificação de modelos explora o conteúdo das informações nos arquivos BIM de maneira transparente, validando esforços de enriquecimento do modelo BIM, de tal forma que informações com este objetivo específico são adicionadas (HJELSETH, 2015b). Ao usar o BIM paramétrico, as interfaces de programação visual podem substituir a codificação convencional para explorar o modelo. A programação visual permite que os usuários criem programas de computacionais manipulando os elementos do programa graficamente, em vez de textualmente (ASL *et al.*, 2014). Exemplos de ferramentas de programação visual utilizados em softwares BIM para são o Grasshopper para McNeel Rhinoceros® e o Dynamo para o Autodesk Revit®.

Uma das tendências internacionais de pesquisa na construção se foca na evolução do BIM, uma das áreas que vem ganhando destaque é a de checagem automática de critérios de normas por meio dos modelos (YALCINKAYA; SINGH, 2015), porém a checagem para outros fins, como segurança, encontra-se ainda em estágios iniciais de desenvolvimento (SCHWABE; TEIZER; KÖNIG, 2019). A checagem de modelos BIM pode ser utilizada para medir a qualidade do modelo e do projeto (DIMYADI; SOLIHIN; HJELSETH, 2016). Esta segunda aplicação será explorada mais profundamente nesta pesquisa, utilizando-se da elaboração de regras que possam ser verificadas automaticamente no modelo BIM para avaliar o desempenho do projeto, estas regras não só checam o conteúdo do modelo, mas utilizam os dados para calcular indicadores (HJELSETH; NISBET, 2010b).

Este trabalho se justifica em utilizar o potencial dos projetos desenvolvidos em BIM como banco de dados de informação da construção, que podem ser analisados através de checagem automática de regras. A programação visual permite explorar os dados do modelo e executar as checagens. Desta forma, busca-se propor uma ferramenta de avaliação automática de desempenho de projetos baseada em modelos BIM.

## **1.2 Problema de Pesquisa**

Os projetos de edificações estão aumentando em complexidade (XIA; CHAN, 2012), estes vem se tornando mais abrangentes ao envolver diversos setores profissionais causando dificuldade para os projetistas que precisam conduzi-los (KO, 2017). Assim, os projeto concluídos podem esconder problemas que podem surgir na fase de construção causando alterações de projeto, um dos principais causadores de atrasos em empreendimentos

(ASSAF; AL-HEJJI, 2006), ou ainda na fase de uso da edificação, causando desconforto e insatisfação dos usuários.

O projeto é fator importante que define todo o ciclo de vida da edificação, pois ele através de todas as suas disciplinas define a forma e a função da edificação (VRIJHOEF; KOSKELA, 2005). Porém, a qualidade do projeto é um conceito, por vezes, subjetivo e dependente da performance do projeto em diversos aspectos diferentes (ALARCÓN; MARDONES, 1998; FABRICIO; MELHADO, 2001). De modo que, foge do domínio de apenas um projetista e exige conhecimentos específicos de diversos especialistas.

Há propostas e recomendações para quantificar a performance dos projetos com métricas de desempenho, uma tentativa de padronizar a avaliação dos projetos. (MASCARÓ, 2010; NARLOCH, 2015). No entanto, avaliar projetos de forma quantitativa é um esforço dispendioso e requer muito tempo. Tais dificuldades podem levar os projetistas a não avaliarem o projeto para não atrasar o desenvolvimento dos mesmos. Perdendo oportunidades de melhorias e, até conhecimento sobre o projeto em aspectos fora da sua área direta de atuação.

Assim, a pergunta de pesquisa que motiva este trabalho é: É possível desenvolver ferramentas automatizadas para analisar características de projetos com base em modelos BIM?

## **1.3 Objetivos**

### ***1.3.1 Objetivo Geral***

Desenvolver uma ferramenta para avaliação automatizada de projeto de edificações em BIM com programação visual.

### ***1.3.2 Objetivos Específicos***

Têm-se como objetivos específicos:

1. Identificar dimensões de desempenho de uma edificação para serem avaliadas por meio do modelo de informação;
2. Selecionar critérios relativos às dimensões especificadas a serem avaliadas;
3. Desenvolver algoritmo para a avaliação dos critérios através programação visual;
4. Propor modelo para desenvolvimento de algoritmos de avaliação de projeto.





## 2 REFERENCIAL TEÓRICO

Este referencial tem como objetivo servir de base teórica para o trabalho. Os conceitos abordados seguem uma ordem lógica. Inicia-se com as definições apropriadas de BIM e suas implicações, parte-se então para o estabelecimento do estado atual da checagem automática de modelos e na última seção avaliam-se formas de aplicação das checagens, com o foco em programação visual.

### 2.1 Building Information Modeling

Succar (2013) define BIM como “um conjunto de tecnologias, processos e políticas que permitem aos múltiplos interessados de um empreendimento a colaborativamente projetar, construir e operar uma construção no espaço virtual”. Já para a Buildingsmart Alliance (2015) BIM é uma representação digital das características físicas e funcionais de uma instalação, fornecendo uma base sólida para as decisões desde o início do seu ciclo de vida, e servindo como fonte de conhecimento compartilhado de recursos para a obtenção de informações sobre a instalação. Para Eastman et al. (2011) possibilita criar digitalmente um ou mais modelos virtuais precisos de uma construção que oferecem suporte ao projeto ao longo de suas fases, permitindo melhor análise e controle do que os processos manuais. O BIM ajuda arquitetos, engenheiros e construtores a visualizar o que deve ser construído em um ambiente simulado para identificar potenciais problemas (AZHAR, 2011).

Este trabalho busca contribuir nos três conjuntos de Succar (2013) ao propor uma ferramenta tecnológica que permite a avaliação automática de desempenho capaz de alterar o processo de projeto, agilizando verificações e análises que podem afetar regulações políticas. Para isso é parte-se do conceito de representação digital, ambiente simulado e fonte de conhecimento (AZHAR, 2011; BUILDINGSMART ALLIANCE, 2015) para realizar análises e controle de processos como proposto por Eastman et al. (2011).

Os *trade-offs*, parte do processo de design, podem ser melhor avaliados usando o BIM, tais como: custo, energia, funcionalidade, estética e construtibilidade (EASTMAN, Chuck *et al.*, 2011). As tecnologias que usam o BIM adotam modelos paramétricos dos elementos construtivos de uma edificação e permitem o desenvolvimento de alterações dinâmicas em um modelo gráfico (COELHO; NOVAES, 2008).

BIM é um conceito que fundamentalmente envolve a modelagem das informações do edifício, criando um modelo digital integrado de todas as especialidades, e que abrange todo o ciclo de vida da edificação. A modelagem 3D paramétrica e a interoperabilidade são características essenciais que dão suporte a esse conceito (AZEVEDO, 2009), o que fornece a

construção uma nova abordagem para a coordenação de todas as partes do empreendimento. Segundo Azhar (2011), há os seguintes usos do BIM:

- a) **Visualização:** Renderizações 3D podem ser facilmente geradas;
- b) **Fabricação:** Capaz de gerar projetos detalhados para vários sistemas construtivos;
- c) **Revisões de norma:** Bombeiros e outras agências governamentais podem utilizar o modelo para avaliar o atendimento das normas;
- d) **Estimativas de custo:** Softwares BIM já incorporam ferramentas para estimar custos. Quantitativos são extraídos automaticamente do modelo e atualizados sempre que o modelo é alterado;
- e) **Sequenciamento da construção:** O modelo pode ser efetivamente utilizado para pedidos de material, fabricação e planejamento de entregas para todos os componentes do empreendimento;
- f) **Conflitos, interferências e detecção de colisões:** Já que os modelos BIM são criados em um espaço 3D geométrico, todos os sistemas principais podem ser automaticamente e instantaneamente checados para interferências. Por exemplo, se uma tubulação atravessa uma viga ou duto de ventilação.

Alguns dos conceitos necessários para utilização do BIM no que tange este trabalho serão discutidos aqui. O que garante sua utilização efetiva é a automatização dos modelos, que se adaptam às mudanças do projeto e à integração dos participantes por meio da colaboração.

### ***2.1.1 Modelagem Paramétrica***

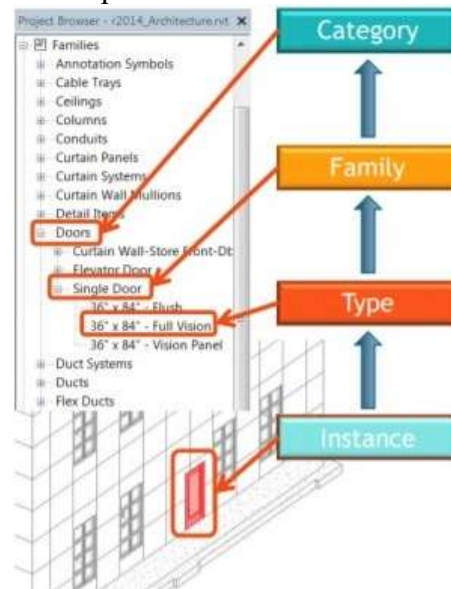
O conceito diferencial do BIM é a parametrização dos modelos, pois permite que os projetos sejam inteligentes. Andrade e Ruschel (2009) afirmam que a modelagem paramétrica consiste na representação computacional de um objeto construído com entidades, cujos atributos podem ser fixos ou variáveis. Os atributos fixos são denominados como controlados e os atributos variáveis podem ser representados por parâmetros e regras, de forma a permitir que objetos sejam automaticamente ajustados de acordo com o controle do usuário e com a mudança de contexto.

As tecnologias que usam o BIM adotam modelos paramétricos dos elementos construtivos de uma edificação e permitem o desenvolvimento de alterações dinâmicas em um modelo gráfico da mesma, de modo que são imediatamente refletidas em todas as pranchas de desenhos associadas, bem como nas tabelas de orçamento e especificações (COELHO;

NOVAES, 2008). Isto se reflete em melhorias no processo de projeto e garante uma resposta rápida as alterações, além de poupar retrabalhos na elaboração de cortes e detalhamentos.

Os modelos BIM apresentam uma hierarquia nos seus elementos na Figura 1 mostra-se a mesma para a ferramenta de elaboração de modelos Revit. A Categoria controla a organização, visibilidade, representações gráficas e opções de agendamento das famílias. Uma Família é um agrupamento de informações 2D e/ou 3D que serve para representar um elemento distinto de construção ou documentação no projeto. Um Tipo é uma representação específica de uma Família definida por características paramétricas, gráficas e de documentação distinta, o que o torna único de outros tipos da família. Uma Instância é uma representação individual de um Tipo no Projeto definido por características paramétricas, gráficas e de documentação exclusivas, o que o torna exclusivo de outras Instâncias no Projeto.

Figura 1 Hierarquia de elementos no Revit



Fonte: Autor

### 2.1.2 BIM ao longo do ciclo de vida do projeto

O BIM pode ser aplicado em todas as fases do projeto de construção, que são as fases de pré-construção, de construção e pós-construção (LATIFFI *et al.*, 2013). A Tabela 1 mostra a aplicação do BIM para projetos de construção em cada uma de suas fases. Pode ser visto que a aplicação BIM tem benefícios específicos e muda ao longo do ciclo de vida do projeto. A capacidade do BIM de promover a colaboração entre os agentes de construção facilita a decisão no processo de projeto (AZHAR, 2011). Além disso, a detecção de

interferências durante o estágio de projeto pode reduzir o tempo e o custo de construção (EASTMAN, Chuck *et al.*, 2011).

A principal aplicação do BIM se concentra na pré-construção, servindo de plataforma para desenvolvimento do produto a ser construído. Porém, sua utilização afeta os estágios subsequentes e os diversos stakeholders do projeto (GHAFARIANHOSEINI *et al.*, 2017), a Tabela 1 mostra como o BIM é usado ao longo do empreendimento . Isso mostra que melhorias no processo de aplicação BIM aprimoram fortemente a qualidade e desempenho do empreendimento. A ferramenta proposta nessa pesquisa deverá ser aplicada em estágios iniciais do projeto, especificamente nas etapas de definição das soluções adotadas, para prever o desempenho do projeto nas etapas futuras.

Realizar análises automatizadas na etapa de Viabilidade executiva permite a tomada de decisão com informações mais assertivas. Durante o Design colabora com o entendimento do projeto especificando sua qualificação em relação aos critérios de análise, permitindo alterações para atende-los ou assumir posições conscientes de desempenho.

Tabela 1 Usos do BIM nas etapas do projeto

<b>Fase</b>	<b>Estágio</b>	<b>Usos do BIM</b>	
<b>Pré-construção</b>	Modelagem das condições existentes	Aumenta a precisão da documentação	
	Viabilidade executiva	Identifica falhas no sequenciamento ou no faseamento	
	Design		Facilita a comunicação e a tomada de decisões mais rápidas
			Realiza verificação e análise de interferências
			Aumenta a efetividade do projeto
	Planejamento	Permite ao gerente de projetos e a construtora a visualizar o sequenciamento das atividades, equipamentos e materiais em busca de melhorias na logística	
	Orçamento	Permite a extração de quantitativos e medições diretamente do modelo 3D	
Análise de canteiro	Diminui o gasto com utilidades e demolição		

		Permite demonstrar o processo de construção, incluindo acessos e saídas, rotas de tráfego, matérias e maquinário
<b>Construção</b>	Construção	Proporciona um melhor acompanhamento de custo e de fluxo de caixa
		Permite o acompanhamento de do trabalho em tempo real, fluxo de recursos veloz e melhor gerenciamento do canteiro
		Acompanhamento de ativos construídos
<b>Pós construção</b>	Operação/Gerenciamento de instalações	Gerenciamento proativo de instalações
		Permite o planejamento de manutenções e proporciona uma histórico de manutenções

Fonte: Adaptado de Latiffi et al. (2013)

## 2.2 Avaliação Automatizada de Modelos

A avaliação automatizada de modelos não é um termo claramente definido. É entendido principalmente como detecção de conflitos ou verificação de conformidade, mas esse entendimento limita o escopo da verificação do modelo. Neste trabalho tomou-se como definição a análise do modelo BIM com o intuito de responder a um **critério**. O critério pode ser uma regra, escrita em linguagem natural ou não, ou um indicador numérico. Para isso utiliza-se de algum sistema computacional que tem a capacidade de interagir com o modelo, extrair, criar e reportar as informações necessárias para responder completamente o critério.

O conceito de *BIM Model Checking* ou Checagem de Modelos BIM é amplamente utilizado para a avaliação automatizada dos critérios do tipo regra, a maior parte das pesquisas se focam nesta área. Os critérios do tipo indicador são normalmente apresentados como um caso específico do BMC. Logo, o que será apresentado nesta seção se aplica aos critérios de ambos os tipos que serão avaliados pela ferramenta proposta neste trabalho.

Múltiplas agências em todo o mundo iniciaram esforços para desenvolver a capacidade de checagem automatizada de códigos de construção e regras de acessibilidade (EASTMAN, Chuck *et al.*, 2011), e este esforço que vem impulsionando o desenvolvimento da checagem de modelos BIM. Pois, a conformidade dos projetos de edifícios com a grande quantidade de regulamentações é verificada em grande parte manualmente, tanto pelo

projetista como o responsáveis pela a regulamentação governamental de construção (PREIDEL; BORRMANN, 2016).

Eastman et al. (2009) define a checagem automatizada de regras como um software que não modifica um projeto de construção apenas avalia-o com base na configuração de objetos, suas relações ou atributos; estes sistemas aplicam regras, restrições ou condições a um projeto. Tais regras retornam resultados como “aprovado”, “reprovado” ou valores de referência, de acordo com a adequação a regra aplicada. A avaliação de modelos automatizada possui diversas aplicações e se mostra uma tendência conforme a adoção do BIM cresce. Solihin e Eastman (2015) mostram que o escopo das regras de verificação se encaixa nas seguintes categorias:

1. Verifica a formação do modelo de construção: este grupo de regras refere-se principalmente a aspectos sintáticos de acordo com o conjunto de padrões ou conjunto previamente acordado de condições requeridas para o modelo.
2. Verificação de códigos reguladores: se concentra na conformidade com códigos ou regulamentos de construção bem definidos ou geralmente prescritivos.
3. Requisitos específicos do cliente: exemplos incluem requisitos para projeto hospitalar ou projeto de tribunais.
4. Construtibilidade e outros requisitos do contratado: estes geralmente envolvem objetos temporários ou condições presentes apenas durante o processo de pré-construção e durante a construção, como fôrmas e escoramento.
5. Segurança e outras regras com possíveis ações corretivas programadas: Elas apoiam as decisões e ajudam a automatizar a busca de possíveis perigos para os trabalhadores durante a construção e para a equipe de manutenção durante a operação.
6. Aprovações de garantia: O modelo de pós-construção é verificado quanto a problemas que podem afetar a garantia ou o custo de manutenção. Esses requisitos podem precisar ser combinados com a inspeção real do local.
7. Integralidade dos dados BIM para a transferência para o gerenciamento de instalações: O papel das mudanças no modelo BIM durante o ciclo de vida do processo de construção, os requisitos posteriores muitas vezes não são considerados, como os de gerenciamento de instalações. As regras para verificar essa completude no final das diferentes fases do contrato serão cada vez mais exigidas.

### 2.2.1 *Processo de avaliação*

Os modelos de construção envolvem grandes conjuntos de dados, mesmo para edifícios de médio porte. Os modelos construídos tipicamente não incluem o nível de detalhe necessário e há, ainda, problemas com a qualidade desses dados (SOLIHIN, Wawan, 2004). Uma recente visão geral de ambientes para verificação de código de construção foi recentemente feita por Solihin e Eastman (2015).

Para os autores o processo de verificação de regras é separado em quatro fases, considerando a primeira fase como de interpretação da regra, a segunda fase de preparação do modelo de construção, a terceira fase de execução da regra e a última fase sendo de relatório de verificação da regra. Cada uma dessas fases é discutida, combinada com seus respectivos efeitos no processo geral de verificação de regras. Conforme descrito em Eastman et al. (2009) e ilustrado na Figura 2, a verificação de regras envolve as seguintes etapas:

**Interpretação de regras:** interpretação e estruturação lógica de regras para cada aplicação. A maioria das regras foi escrita em linguagens humanas. Um intermediário comum para o mapeamento de regras da linguagem natural para a forma processável pelo computador é uma tabela que consiste em regras parametrizadas.

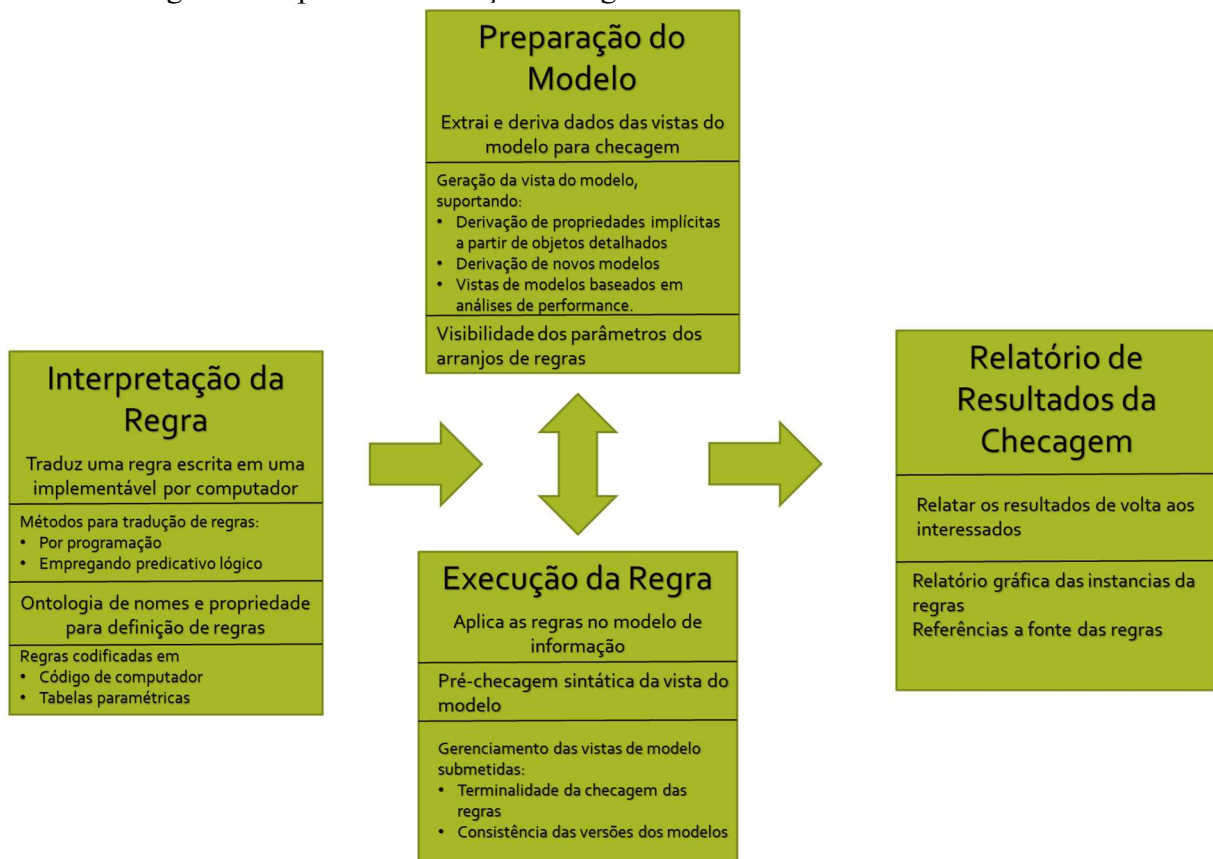
**Preparação do modelo:** as informações necessárias para a verificação. Diversas definições de visualização de modelo podem ser usadas para especificar as informações necessárias para executar determinadas verificações de regras. Deriva-se os dados necessários para um tipo específico de verificação de regras e se extrai subconjuntos de um determinado modelo de construção para permitir uma execução mais eficiente.

**Execução da regra:** executa a verificação de regras em determinados modelos de construção. Lida com o gerenciamento do processo de execução. Tecnicamente, essa etapa envolve tanto o pré-processamento quanto o pós-processamento da verificação de regras.

**Relatórios:** reporta os resultados da verificação de regras. Nesse estágio, o sistema geralmente produz relatórios gráficos, incluindo detalhes textuais, bem como referências a regras de origem codificadas. Em último caso, configurar procedimentos para corrigir automaticamente falhas de regras



Figura 2 Etapas da verificação de regras



Fonte: (EASTMAN, C. *et al.*, 2009)

### 2.2.2 Classificação das regras

Solihin e Eastman (2015) recorrem à uma extensa exposição internacional com vários códigos de construção e verificação de regras na AEC para introduzir uma classificação geral para regras de verificação. Os autores propõem uma classificação baseado em critérios que se aplicam a todos os aspectos conhecidos da checagem de modelos, chegando a quatro classes distintas:

**Classe 1 - Regras que requerem um único ou pequeno número de dados explícitos:** Esta classe de regras verifica atributos explícitos e referências de entidades que existem dentro do modelo BIM. Este tipo de regra é usado para verificar as configurações de

atributos obrigatórias para a correta verificação ou, ainda, para algumas verificações simples de código de construção como: tipo de portas na saída da edificação ou necessidade de elevador devido ao número de pavimentos.

**Classe 2 - Regras que exigem valores de atributos derivados simples:** As verificações são baseadas em um único valor ou em um pequeno conjunto de valores derivados. Essa classe deriva atributos ou valores, mas não gera novas estruturas de dados. Esta classe de regras envolve um trade-off entre os dados exigidos do usuário versus os dados derivados da verificação de regras. Um exemplo é a verificação de espaçamento entre dois objetos, onde a informação não está contida no componente, utilizando-se da geometria como referência.

**Classe 3 - Regras que exigem estrutura de dados estendida:** Esta classe de regras requer entender a estrutura de dados para que encapsule condições semânticas de nível mais alto de dados de construção. A ideia principal é ser capaz de “computar uma vez, usar muitas”, já que tais informações geralmente requerem computação extensa, muitas vezes envolvendo operações de geometria. Várias estratégias de construção da estrutura de dados podem ser aplicadas. O uso típico dessa classe de regras é a criação de verificação de código que envolve requisitos complexos, como cálculo do menor caminho.

**Classe 4 - Regras que exigem uma “prova de solução”:** Esta é um conjunto de regras que não exigem estritamente conformidade ou não conformidade, mas requerem uma “prova de solução”. As regras geralmente se concentram mais em como o modelo de construção prova a conformidade, em vez de simplesmente cumprir os critérios prescritos. Ele representa códigos baseados em desempenho ou outras regras semelhantes. No entanto, ele também pode ser aplicado a regras que podem ser modificadas para satisfazer a conformidade, examinando uma solução adicional inserido virtualmente no projeto. Estas estão fortemente baseados em bancos de regras que são atualizados conforme a necessidade. Por exemplo, crie regras como “Projetar colunas com furos de 0,55 a 1,05 m acima do nível do piso para fornecer locais de suporte para linhas de vida e guarda-corpos”. Esta regra deve ser adicionada à base de conhecimento como resultado de estatísticas de que houve um número relativamente alto de mortes causadas por esse motivo.

A Tabela 2 apresenta uma estrutura que amplia o uso da avaliação de modelos em quatro categorias principais com base na intenção e no tipo de checagem realizada. Hjelseth; Nisbet (2010b) os define como:

Tabela 2 Conceitos dos modelos de checagem

Intenção da checagem	Checagem	Resultados
Validação de conformidade	Conformidade	Aprovado/Reprovado
Orientação para soluções técnicas	Orientação	Proposta
Adaptação de objetos	Objetos Inteligentes	Objeto adaptável
Verificar o conteúdo da informação	Conteúdo	Presente/ausente

Fonte: Adaptado de (HJELSETH; NISBET, 2010b)

**Conformidade:** Este é o entendimento “padrão” do BMC (e, portanto, importante estar ciente de que este não é o único). Esse tipo de verificação é baseado na comparação do modelo com critérios predefinidos. Um exemplo desse tipo de regra diz que uma largura de porta mínima aceitável é de 800 mm ou mais. A conformidade é atingida se a largura mínima for maior que a largura necessária.

**Orientação:** A intenção desse conceito é orientar o projetista a considerar uma gama maior de soluções mais usadas de acordo com as regras de melhores práticas. Isto é particularmente relevante em esferas onde o designer não é um especialista. A verificação é baseada em dois elementos: regras que identificam as situações em que os problemas ocorrem e uma apresentação de uma lista de possíveis soluções. Este tipo de verificação está intimamente relacionado às melhores práticas dos sistemas de apoio à decisão, mas até agora este conceito não foi implementado na indústria da construção civil.

**Objetos inteligentes:** Objetos ativos ou objetos inteligentes são uma descrição alternativa para esse tipo de verificação. A intenção é permitir que o próprio objeto registre seu ambiente e depois se adapte automaticamente seguindo as regras de comportamento incorporadas. A verificação de modelos adaptativos pode ser dividida em adaptação de objetos e adaptação do sistema. Um exemplo no nível de objeto implementado na maioria das ferramentas de criação BIM pode ser uma regra que diz que você deve sempre colocar portas ou janelas em uma parede. Um exemplo no nível do sistema poderia ser se o edifício deve cumprir os regulamentos de acessibilidade, todas as larguras das portas devem ser ajustadas para um mínimo de 800 mm.

**Conteúdo:** Pode ser considerada uma declaração de informações entregues. A intenção aqui é examinar o modelo BIM para um propósito específico. As regras de filtragem podem ser usadas para relatar informações relevantes. As informações podem ser analisadas em software, planilhas, processadores de texto e bancos de dados da BMC de maneiras flexíveis.

Estas formas de classificação permitem uma melhor adequação de toda avaliação para o objetivo. Já que ao compará-las com um tipo semelhante pode-se encontrar uma abordagem de sucesso ou um problema conhecido para determinado tipo de critério. Neste trabalho os sistemas de classificação são usados principalmente para orientar o desenvolvimento.

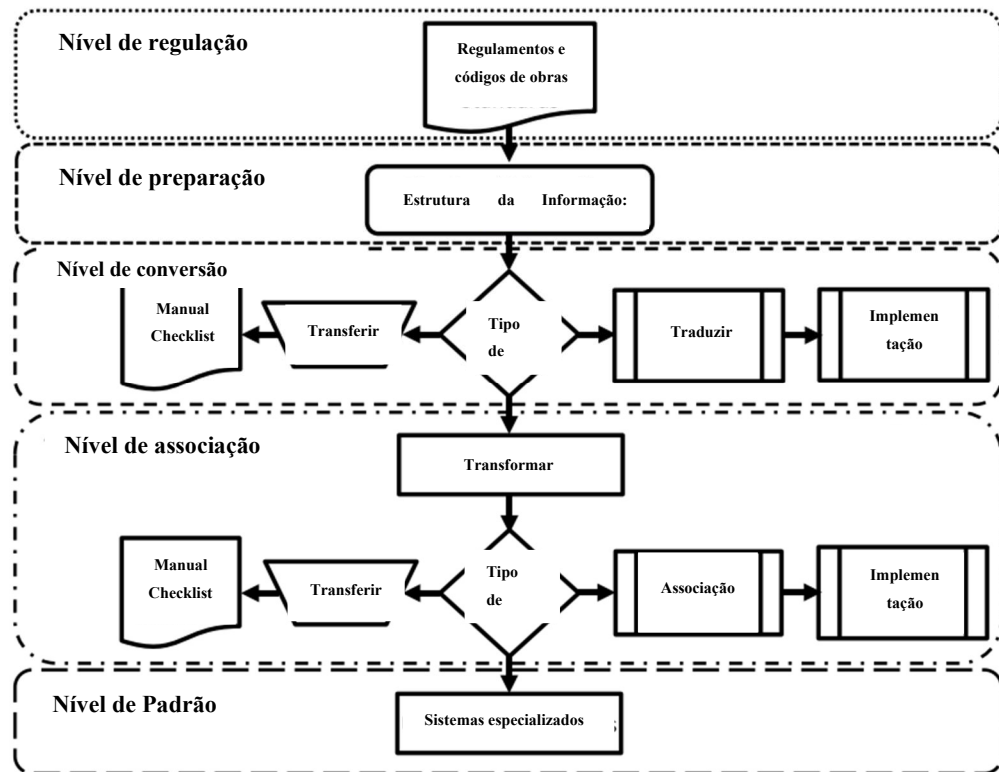
### ***2.2.3 Transformação de regras***

Há grande quantidade de pesquisas realizadas internacionalmente nas últimas quatro décadas na área de representação de códigos de construção em um formato computável para verificação automatizada de conformidade (MACIT İLAL; GÜNAYDIN, 2017). Os mesmos autores afirmam que os esforços de pesquisa se concentraram principalmente em abordagens codificadas diretamente no sistema (exigem conhecimento em programação para definir, escrever e manter códigos), recentemente, a atenção tem sido direcionada para o estudo da abordagem de modelagem semântica (criadas para serem utilizadas sem conhecimento de desenvolvimento de software).

Nesse momento inicial será mostrado o método que será utilizado para converter regras em linguagem real natural para um formato que pode ser implementado em computador. Baseou-se na proposta de análise semântica baseada em ontologia feita por Hjelseth (2015), mostrada na Figura 3, que como resultado da sua pesquisa combinou três métodos: RASE, TIO e Tx3. O processo propõe ferramentas para lidar com diversos tipos de regulações e convertê-las em regras que poderão ser executadas em um sistema de verificação automático.

A etapa inicial é no nível de regulação, onde as regras em linguagem natural humana correspondem ao insumo inicial, segue-se por quatro níveis delimitados a seguir. Esta metodologia foi inicialmente proposta para regulações e códigos de obra, porém será adaptada para abranger outros tipos de regras.

Figura 3 Método Tx3 de transformação de regras



Fonte: Adaptado de Hjelseth (2015)

### 2.2.3.1 Nível de preparação

Nesse nível as regras precisam ser compreendidas de uma maneira lógica para que seja possível avaliar como a regra será aplicada. As regulamentações são escritas em uma linguagem jurídica ou técnica e frequentemente devem ser reestruturadas antes de poderem ser usadas como especificações para implementação em software (HJELSETH, 2012b).

O método utilizado será a metodologia RASE discutida em Hjelseth e Nisbet (2010a), uma abreviação para os marcadores utilizados: Requirimento, Aplicação, Seleção e Exceção. Os autores definem como uma metodologia de marcação baseada em semântica para identificar elementos em texto não estruturado e transformá-los em uma forma estruturada preparada para implementação em software de verificação de modelo BIM. Os marcadores possuem uma codificação de cores que faz com que se identifique facilmente as funções, a Figura 4 mostra as cores sugeridas.

Figura 4 Operadores RASE

Requisitos	Aplicação	Seleção	Exceção

Fonte: (HJELSETH; NISBET, 2010a)

Os marcadores são utilizados para identificar funções nas sentenças, como mostrado a seguir:

- O escopo superior da regra é a **Seleção**, corresponde ao tipo de objeto na qual a regra será executada;
- O escopo restrito da regra é a **Aplicação**, identifica qual parâmetro ou informação será avaliada;
- "Deve conter", "precisa conter" ou "Não deve superar" são sentenças de **Requisito**, mostram as condições que precisam ser satisfeitas;
- "A menos que" ou "exceto" são sentenças de **Exceção**, servem para mostrar situações em que a regra não se aplica

A metodologia RASE foi aplicada como exemplo na ISO 21542, Larguras das entradas. Uma amostra de texto desta norma é mostrada na **Erro! Fonte de referência não encontrada.** e as partes selecionadas são marcadas de acordo com a metodologia.

Tabela 3 Exemplo de aplicação RASE

Largura da entrada
A largura mínima desobstruída de uma porta de entrada não deve ser inferior a 800 mm; recomenda-se 850 mm ou mais, pois pode ser necessário mais espaço para uma pessoa que usa uma cadeira de rodas elétrica.
NOTA: Muitos regulamentos nacionais de construção exigem uma largura mínima de 900 mm para uma porta de entrada

Fonte: (HJELSETH; NISBET, 2010a)

O Tabela 4 é a versão tabular do texto marcado no Tabela 3. Cada elemento de texto é suportado por metadados que descrevem seus atributos e valores. Esta tabela apresenta o regulamento como uma regra aplicável. Essa forma de representação de conhecimento é uma preparação para implementação em software. A separação de "Propriedade" e "Comparação" permite análises parametrizadas.

Tabela 4 Exemplo de resultado da aplicação do RASE

Frase marcada	Operador	Tipo	Propriedade	Comparação	Métrica	Unidade
Porta de entrada	A	Porta	é externa?	Igual	True	booleano
Largura desobstruída	S	Porta e moldura	Acessível para cadeirantes?	Inclui	Largura	
Não deve ser	R			≥	800	mm

inferior a 800 mm						
Regulamentos nacionais de construção	E		Regulamento nacional?	Prioridade ao nacional	Nome da regulação	
Largura desobstruída	S	Porta e moldura	Acessível para cadeirantes?	Inclui	Largura	
Não deve ser inferior a 900 mm	R			≥	900	mm

Fonte: Adaptado de (HJELSETH; NISBET, 2010a)

O uso da metodologia RASE pode servir a dois propósitos: especificação de regras para programação como conjuntos de regras digitais para o software de avaliação de modelos BIM e uso do RASE como banco de dados de requisitos (HJELSETH, 2015b). A metodologia RASE já foi utilizada para extrair requisitos de regulamentos de sustentabilidade, permitindo que as regras fosse geradas rapidamente e excluindo a necessidade de manipular regras em linguagem natural (KASIM *et al.*, 2013).

#### 2.2.3.2 *Nível de conversão*

Neste nível as regras já compreendidas e tabuladas passarão pela Tx3. Uma metodologia para classificar regras de acordo com a capacidade de implementação da regulação (HJELSETH, 2012b). Baseia-se na classificação em três categorias principais: Transcrever, Transformar e Transferir. Os critérios para cada categoria baseiam-se na capacidade de estabelecer, respectivamente, uma ligação direta, indireta ou inexistente entre a intenção qualitativa ou quantitativa do indicador na regra e uma métrica quantitativa discreta aplicável ao software (HJELSETH, 2015b). A Figura 5 mostra com detalhes os tipos de regra.

Figura 5 Classificação das Regras no Tx3



Fonte: Adaptado de (HJELSETH, 2012b)

Declarações classificadas como “Transcrever” podem ser expressas como regra computável por procedimentos pré-definidos, como o RASE. A parte desafiadora são as declarações classificadas como “Transformar”, a decisão se estas serão definitivamente implementadas ou precisaram ser transferidas, será feita no Nível de Associação. A “Transferência” para profissionais experientes da AEC para interpretação será, muitas vezes, a melhor solução para declarações que são muito dependentes de contexto ou tem grande número de restrições e informações no modelo.

### 2.2.3.3 Nível de Associação

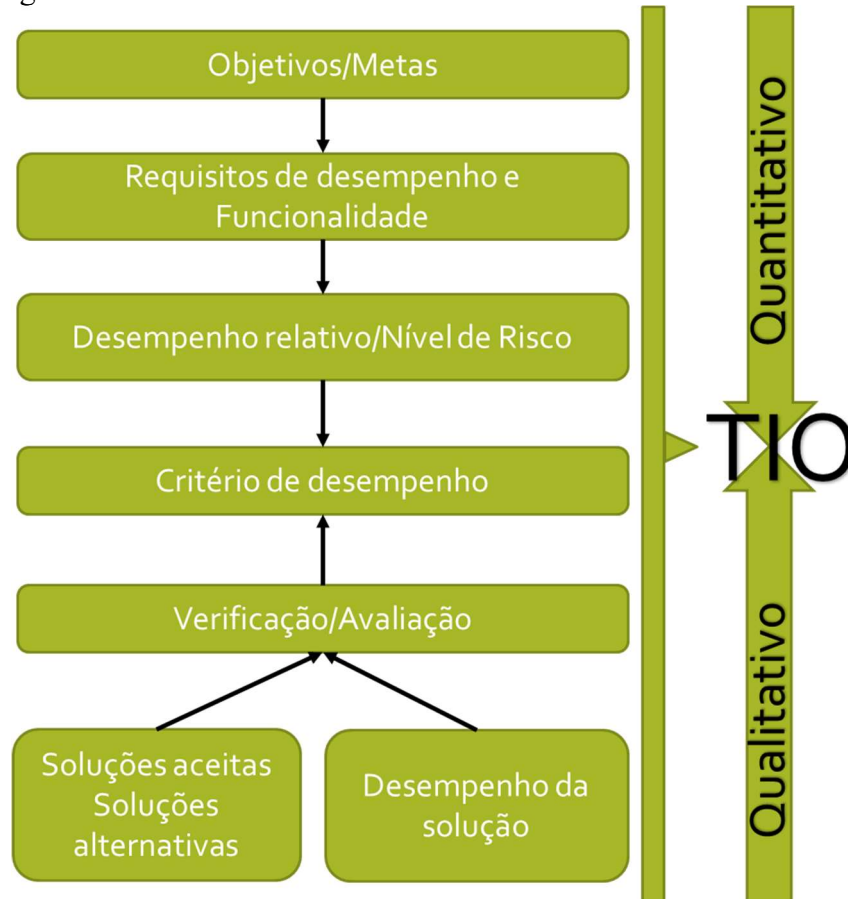
A função deste nível é especificar métricas para sentenças não prescritivas ou que não estabelecem um critério específico. São sentenças como “Janelas devem ser de fácil abertura” e “as vagas devem estar o mais próximo possível da entrada” onde não é possível especificar diretamente o critério para avaliação da regra.

O nível de associação apoiado pelo método TIO, “Test Indicator Objectives”. A TIO pode ser considerado como uma “regra de associação” que, de maneira transparente, válida e confiável, estabelece um mapeamento entre o objetivo qualitativo/intenção na declaração e a métrica quantitativa discreta requerida nas regras computáveis (HJELSETH, 2012b). Para permitir a verificação automatizada de modelos, é necessário que os requisitos



baseados em função sejam transferidos de métricas qualitativas para métricas quantitativas com valores discretos. Então, a partir da análise de soluções que foram aceitas anteriormente, formula-se um valor quantitativo para a aquele requisito.

Figura 6 Funcionamento do TIO



Fonte: Adaptado de (HJELSETH, 2012a)

É importante estar ciente de que, quando a regra lógica é estabelecida, os valores da métrica podem ser considerados como instâncias paramétricas. Essa abordagem pode habilitar a verificação de desempenho em diferentes níveis; um conjunto de regras com os requisitos mínimos e outro com requisitos mais altos (HJELSETH, 2012a). O Tabela 5 apresenta um dicionário de TIOs onde as metas qualitativas são transformadas em métricas quantitativas, o exemplo usado foi a ISO 21542:2011 que estabelece normas de acessibilidade. É possível observar que através das afirmações da norma apenas um profissional com experiência na área conseguiria definir se as condições do projeto podem ser aprovadas, constatando a importância desta etapa para a implementação do sistema de checagem.

Tabela 5 Dicionário TIO para metas qualitativas transformadas em quantitativas

Artigo	Expressão qualitativa	TIO	Métrica
<b>Dimensão mínima</b>			
26.3	Os sinais devem estar bem iluminados, sem brilho	Iluminação mínima lux	X lux
<b>Dimensão máxima</b>			
6.8.3	Localização de vagas de estacionamento acessíveis, devem ser o mais próximo possível das entradas/elevadores.	Distância máxima	X mm
<b>Solução pré-aprovada</b>			
18.2.3	As janelas devem ser fácil de abrir e fechar. Deve ser possível abrir e fechar as janelas com apenas uma mão	Tipo de janela pré-aceito (aprovado)	Aprovado pela organização x
<b>Propriedade do produto - Superfície</b>			
26.3	A superfície do piso deve ser antiderrapante, anti-reflexo e firme.	Especifique o coeficiente de atrito para pisos	0.x

Fonte: Adaptado de (HJELSETH, 2012a)

#### 2.2.3.4 Nível de padrão

O nível de padrão é uma opção para resolver declarações regulatórias mais complicadas por meio de suporte de métodos e tecnologia baseados em KBE (Knowledge Based Engineering), AI (inteligência artificial) e sistemas especializados são necessários. Desenvolver soluções sofisticadas de mineração textual para geração automatizada de regras baseado em regulações é um campo de pesquisa atual (GHANNAD *et al.*, 2019; KOO *et al.*, 2019; SINGER; BORRMANN, 2015). No entanto, a solução mais comum é permitir que um profissional capacitado em AEC interprete essas declarações reguladoras manualmente, “Transferência”. Nesta pesquisa essa análise específica não será realizada

#### 2.2.4 Implementação

De acordo com Zhang et al. (2013) um sistema de verificação baseado em regras pode ser implementado de duas maneiras diferentes: Uma maneira é um aplicativo ou plug-in baseado em um software de design que permite aos projetistas verificarem um modelo de construção durante o processo de projeto, a outra abordagem é usar um visualizador de modelo baseado na Industry Foundation Class (IFC). Este é um padrão público reconhecido

internacionalmente para a troca de dados e a integração nas indústrias de construção civil (EASTMAN, C. M. *et al.*, 2010).

As ferramentas de design BIM disponíveis atualmente não fornecem funções de verificação de modelo, para contornar tal problema um aplicativo pode ser desenvolvido na plataforma BIM, fornecendo facilidade para um designer validar seu modelo de acordo com as regras de destino sem a necessidade de alterar o projeto posteriormente (ZHANG, S. *et al.*, 2013). No entanto, fazer interfaces eficientes e robustas envolve uma variedade de problemas técnicos. Isso inclui os requisitos de diferentes geometrias a serem usadas em diferentes análises, diferentes propriedades de materiais e diferentes convenções de nomenclatura (LEE *et al.*, 2012).

Os verificadores de modelo baseados em regras automatizadas amplamente conhecidos incluem Revit, Navisworks, Solibri Model Checker (SMC), *Express Data Manager* (EDM) e FORNAX (UHM *et al.*, 2015). Hjelseth (2015) avaliou um conjunto de softwares comerciais com relação a suas funções de checagem de modelos, Tabela 6. Os resultados indicam que a funcionalidade em softwares comerciais abrange os requisitos para verificação de modelos com base em regras simples e conteúdo não específico de informações no arquivo BIM.

Tabela 6 Funções dos Softwares comerciais usados na checagem

Funções/Softwares		Solibri Model Checker	Navisworks Manage	Projectwise Navigator	BIMsight	dRofus Nosyko	Open source BIM server	Free IFC Viewers
Funções	Clash detection	X	X	X	X		X	
	Checagem de conformidade	X				X	X	
	Checagem de conteúdo	X				X	X	
	Função de busca	X	X	X	X	X	X	X
	Relatórios	X	X	X	X	X	X	
	Quantitativos	X	X	X		X	X	
	Linha do Tempo		X					

Funções/Softwares	Solibri Model Checker	Navisworks Manage	Projectwise Navigator	BIMsight	dRofus Nosyko	Open source BIM server	Free IFC Viewers
	Animações		X				
Formatos	Importação de arquivo BIM	ifc, dwg	quase todos	dgn, dwg, pdf	ifc	ifc	ifc, dwg, outro
	Link direto com o Revit		X			X	
	Link direto com Archicad	X					
	Relatórios de exportação	X					
Conjunto de regras	Conjunto de regras separados	X					
	Modificação de regras existentes	X	X		X	X	X
	Combinar regras em um novo conjunto	X				X	X
	Desenvolver novos conjuntos de regras					X	X
Custo	Custo da licença	Pro	Pro	Pro	Livre	Pro	Livre

Fonte: Adaptado de (HJELSETH, 2015a)

Na maioria das abordagens conhecidas, as regras foram implementadas por desenvolvedores de software como código procedural embutido no sistema de verificação de código de construção (DING *et al.*, 2006; EASTMAN, C. *et al.*, 2009), ou seja, procedimentos que simplesmente contêm um conjunto de passos computacionais a serem executados, “caixas pretas” onde não se tem acessos ao procedimento sendo realizado. Na maioria dos casos, o código não é acessível para desenvolvedores de terceiros ou especialistas na área, causando desconfiança. Além disso, as modificações no código implementado em resposta a alterações nos documentos normativos só podem ser realizadas por especialistas em software (PREIDEL; BORRMANN, 2016).

Um exemplo dessa implementação de caixa preta é o sistema CORENET e-PlanCheck, que está sendo usado em Cingapura para verificar a conformidade de modelos de edifícios digitais com códigos de construção nas áreas de controle predial, acesso livre de

barreiras e proteção contra incêndio (SOLIHIN, Wawan, 2004). O componente principal do sistema de verificação de código é a biblioteca FORNAX que foi desenvolvida e mantida por uma empresa privada (PREIDEL; BORRMANN, 2016). Como consequência, as etapas individuais do processo de verificação não são visíveis, de modo que não há uma maneira para os usuários fazerem alterações.

Uma abordagem que vem sendo utilizado atualmente é o desenvolvimento de sistemas de checagem baseados Linguagens de Programação Visual (VPL). Estas são linguagens formais, baseadas em imagens, definidas por objetos gráficos constituídos de nós e conexões (SINGER; BORRMANN, 2015). Myers (1990) com base em uma pesquisa com 50 linguagens de programação visual, já mostrava que um estilo mais visual de programação poderia ser mais fácil de entender para não programadores ou programadores iniciantes (Arquitetos e Engenheiros normalmente encaixados nessas categorias). No contexto de BIM, as VPLs vem se tornado progressivamente importantes para lidar com processos de modelagem geométrica (ZARDO; DA SILVA; MUSSI, 2017).

Diversos autores pesquisaram o uso de algum tipo de VPL em alguma etapa do seu processo de checagem de regras (GHANNAD *et al.*, 2019; KHAN *et al.*, 2019; KIM *et al.*, 2018; PAUWELS *et al.*, 2011; PREIDEL; BORRMANN, 2016). A próxima seção discutirá mais a respeito de como esses sistemas foram empregados.

#### 2.2.4.1 Linguagens de Programação Visual em BIM

Uma enorme quantidade de dados são geradas nos modelos, e acessar, ler e gravar esses dados continua difícil para aqueles sem habilidades de programação tradicionais (MICHAEL, 2016). No contexto do BIM, uma alternativa são os scripts de programação visual. Os scripts podem usar dados do modelo para gerar novas soluções de design e, também, para analisar, avaliar e otimizar decisões de relacionadas à forma e ao desempenho de um edifício" (KENSEK, 2014).

Com a programação visual, *software* podem ser manipulados graficamente em vez de textualmente. Algumas pesquisas ilustram que não programadores e iniciantes possuem maior facilidade entendendo a programação visual do que a linguagem de programação convencional (ASL *et al.*, 2014).

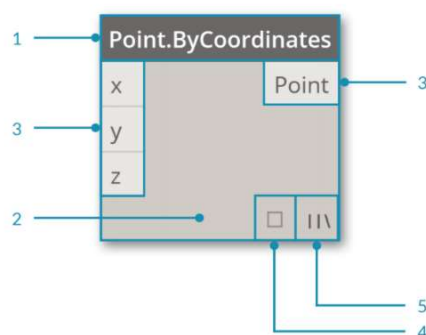
Interfaces de programação visual já estão disponíveis para os principais *software* de modelagem: para o Autodesk Revit a solução mais frequente é o Dynamo (DYNAMO, 2018); para o Rhinoceros se tem o Grasshoper (GRASSHOPPER, 2018) e para a Bentley os Componentes Generativos (BENTLEY, 2018). Com a ajuda de nós e conexões arquitetos e

engenheiros podem definir roteiros e regras para modelagem de elementos e projetos nestas plataformas. A geometria pode então ser modificada de acordo com ajustes nos valores de entrada dos parâmetros (SINGER; BORRMANN, 2015). Este trabalho estará focado na utilização do Dynamo para desenvolvimento das aplicações.

#### 2.2.4.2 *Dynamo*

Em Primer (2018), um guia de programação visual no Autodesk Dynamo, o Dynamo é um ambiente de programação para usuários de todos os níveis, admitindo a criação de lógica visual para automatizar tarefas, formatar fluxos de trabalho, e explorar conceitos paramétricos, direcionando a geometria e comportamento de modelos, bem como extrair e organizar informações do projeto. Estas operações são executadas através de nós e conexões. O nó consiste em uma rotina para executar uma tarefa, podendo esta ser simples, como guardar um número, ou complexa, como criar uma geometria tridimensional (MOUSIADIS; MENGANA, 2016). Com poucas exceções, a estrutura dos nós, (Figura1), na interface do Dynamo é composta por cinco partes: (1) Nome do nó; (2) Janela principal: Clicar com botão direito apresenta opções do nó; (3) Portas de entrada e saída onde acontecem as conexões; (4) Pré-visualização de dados onde se observa os resultados deste nó; (5) Ícone de amarra (laço) que indica a opção de amarra adotada.

Figura 7 Estrutura de um nó no Autodesk® Dynamo



Fonte: (MOUSIADIS; MENGANA, 2016)

Associado com o Revit, o Dynamo é uma ferramenta que pode aperfeiçoar diversos processos em um modelo (PRIMER, 2018) O Dynamo permite o planejamento de

processos customizados, e o controle de objetos do modelo por rotinas, ampliando capacidades do Revit (HUMPPI, 2015). Isto viabiliza novas possibilidades na criação e modificação do modelo e no gerenciamento de informação (tratamento de dados).

Se pode enumerar a facilidade na customização, permitindo a automatização de rotinas pelos usuários sem a necessidade de dominar a complexa Interface de Programação de Aplicação (API); Sistematização de relações entre elementos e parâmetros do modelo; e a capacidade de criar modelos a partir da preconcepção (MOUSIADIS; MENGANA, 2016), avaliando possibilidades, indicadores e viabilidade do projeto (ANDRADE; FERREIRA, 2017; NEMBRINI; SAMBERGER; LABELLE, 2014).

### **2.3 Aplicação dos sistemas de checagem**

Neste tópico serão analisadas aplicações de checagem de modelos desenvolvidas por diversos autores. O objetivo é demonstrar o potencial destas ferramentas e identificar abordagens de sucesso que poderão ser utilizadas nesta pesquisa, principalmente as voltadas a aplicação de checagem baseada em regras. Aqui serão destacadas a proposta da aplicação, a interpretação das regras, a implementação e os resultados encontrados de pesquisas que utilizaram as ferramentas de checagem na fase inicial de projeto.

Zhang et al. (2013) delineou uma estrutura para um sistema de verificação baseado em regras para planejamento e simulação de segurança, integrando BIM e segurança. Os autores mostram que possíveis perigos podem ser automaticamente identificados e os métodos de prevenção podem ser aplicados em uma abordagem automatizada. Uma lógica de conversão de regras de segurança baseada em tabela foi desenvolvida com base nas regras de proteção contra quedas e em práticas recomendadas de segurança e saúde. A viabilidade da abordagem foi mostrada ao integrar a implementação para prevenção de queda ao Tekla Structures, software BIM de engenharia que permite design e gerenciamento. Em Zhang et al. (2015) o framework é melhorado e um protótipo é desenvolvido no Solibri Model Checker. A contribuição é uma estrutura automatizada de verificação de regras que integra a segurança ao BIM de forma eficaz e fornece aos profissionais um método para detectar e prevenir os riscos relacionados a quedas.

O SmartCodes é uma nova iniciativa do International Code Council (ICC) que procura automatizar a verificação de conformidade de código. Nawari, (2012) explica que objetivo é criar uma lista de verificação de elementos de construção para procurar e inspecionar os componentes da construção. O Autor desenvolveu um sistema baseado em várias tecnologias de suporte importantes. Especificamente, eles são compostos por BIM,

XMLSmartCodes e Language Integrate Query (LINQ). Nawari, (2012) avaliou as capacidades funcionais da tecnologia e da estrutura dos SmartCodes, sugerindo uma nova estrutura para o desenvolvimento de sistemas automatizados de verificação de código que validará projetos de construções com relação a regulamentos, provisões e condições definidas pelo usuário.

O estudo desenvolvido por Liu e Issa (2014) mostra que é possível verificar capacidade de manutenção do edifício na fase de projeto. Os autores investigam, diferentes ferramentas BIM, o Solibri Model Checker (SMC) e um Revit Add-in, para encontrar problemas de manutenibilidade. O Microsoft Visual Studio é usado como uma ferramenta para desenvolver o aplicativo Add-in. Um estudo de caso envolvendo acessibilidade de manutenção de exaustores é usado para validar a solução proposta. Os autores relatam que o Add-in oferece mais possibilidades, na medida em que permite acessar mais livremente as informações geométricas no modelo. Este estudo fornece uma aplicação preliminar para um caso específico.

Ji e Leite (2018) integra BIM e a verificação baseada em regras para abordar o planejamento de guindaste de torre de forma automatizada. Os autores propõem um framework que integra modelagem 4D e verificação baseada em regras para revisão de planos de guindastes de torre. Este trabalho apresenta um sistema protótipo para o modelo é atualizado através do Dynamo e as regras específicas de guindastes são aplicadas no SMC.

A checagem baseada em regras foi utilizada no planejamento do layout de canteiro. Os autores demonstraram que os *Business* por Schwabe, Teizer e König, (2019) *Rule Management Systems* (BRMS) já existentes, como o mecanismo de regras de código aberto *Drools*, podem ser usados. Em seu trabalho os autores combinam o *Drools* com o IFC para coletar dados e usar dentro do mecanismo de regras. A definição dos requisitos para a aplicação do sistema se divide nos níveis de informação e geométricos.

Uma abordagem semelhante foi adotada por Khan et al. (2019) para avaliar conformidade automatizada de regras de segurança para trabalhos de escavação. Os autores focaram-se em três módulos: extração de informações e projeto lógico, conversão de informações e integração de processos e geração de planos automáticos e de segurança. Um conjunto de algoritmos baseados em regras foi desenvolvido em GrassHopper, usando linguagem de programação visual (VPL), que gera automaticamente condições geométricas no modelo, identifica potenciais de risco a segurança e instala recursos de proteção.



A checagem para avaliar códigos de construção tem várias propostas sendo desenvolvidas. Em um estudo recente, Ghannad et al. (2019) propõe uma nova estrutura modularizada que integra o padrão aberto emergente, LegalRuleML, com uma linguagem de programação visual. A abordagem proposta abrange todo o processo de verificação de conformidade, desde a interpretação de requisitos baseados em linguagem natural até regras legíveis por máquina, categorização de regras, parametrização de regras e a execução das regras no modelo de informações de construção padrão ISO.

Analisando as aplicações mencionadas anteriormente se nota alguns pontos convergentes. A grande maioria dos trabalhos utilizou as 4 etapas propostas por Eastman et al. (2009) no desenvolvimento do sistema de checagem, porém foi comum observar algumas adaptações para englobar especificidades. Um dos gargalos observados foi a padronização das regras, há propostas de formatos para *datasets* de regras em mais de um dos trabalhos avaliados, mas nenhum formato parece dominante. Principalmente na verificação de códigos de construção, que possui pesquisas relativamente mais avançadas, este tipo de problema começa a aparecer.

As linguagens em que os sistemas foram desenvolvidos variou bastante, o SMC como ferramenta dominante no mercado se mostrou como etapa inicial para as pesquisas. Logo os pesquisadores desenvolviam plugins para softwares comerciais de modelagem ou se utilizavam de linguagens computacionais baseadas em IFC. Um ponto interessante é a utilização de Linguagens de Programação Visual, em que mais de um autor verificou maior facilidade de interpretação dos algoritmos e implementou em alguma fase do sistema.

### 3 METODOLOGIA

Este tópico irá discutir como a pesquisa será conduzida. Inicialmente, a estratégia de pesquisa escolhida é apresentada, junto com os devidos delineamentos específicos para este estudo. Então, o método é estruturado como um fluxo e cada passo é detalhado.

#### 3.1 Estratégia de Pesquisa

O objetivo principal dessa pesquisa é desenvolver uma ferramenta prática, focada em solucionar o problema identificado, gerando a partir do processo um modelo teórico. Opta-se pela Design Science Research (DSR), pois esta envolve duas atividades principais: a criação de novos conhecimentos através do design de artefatos originais ou inovadores (objetos ou processos) e a análise do uso e desempenho do artefato com reflexão e abstração (HEVNER *et al.*, 2004). O foco da pesquisa é efetivamente direcionado ao projeto de artefatos que sustentem melhores soluções para os problemas existentes (DRESCH, Aline; LACERDA; JÚNIOR, 2015). Design Science Research é um conjunto de técnicas analíticas que permitem o desenvolvimento de pesquisas nas diversas áreas, em particular na engenharia (KUECHLER; VAISHNAVI, 2008). Assim, esta se encaixa neste trabalho por proporcionar uma forma de gerar um artefato para a solução do problema ao mesmo tempo que gera conhecimento teórico para a área. Hevner et al. (2004) define as diretrizes para a realização de uma DSR:

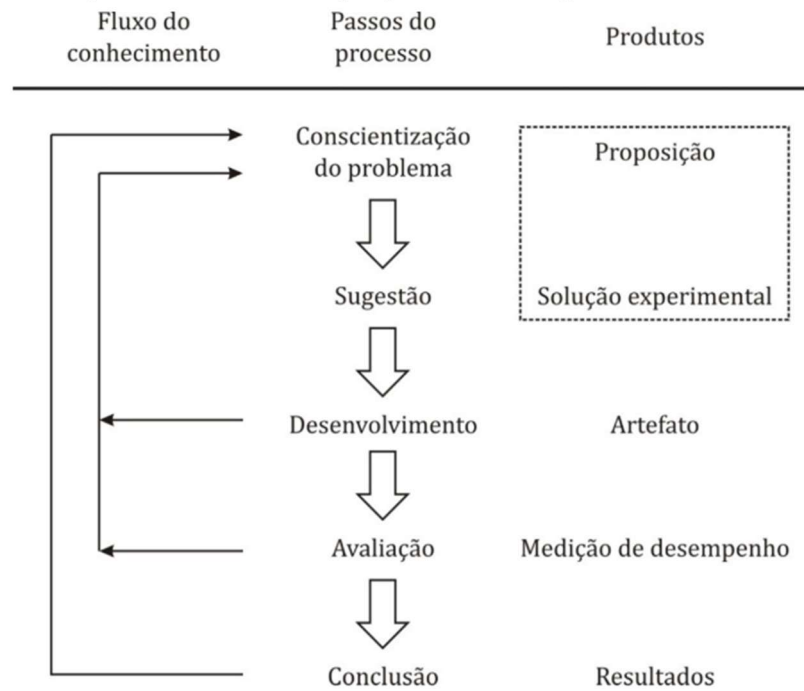
- Design como um artefato: a DSR deve produzir um artefato viável na forma de um construto, um modelo, um método ou uma instanciação.
- Relevância do problema: o objetivo da DSR é desenvolver soluções baseadas em tecnologia para problemas de negócios importantes e relevantes.
- Avaliação de projeto: a utilidade, a qualidade e a eficácia de um artefato de projeto devem ser rigorosamente demonstradas por meio de métodos de avaliação bem executados.
- Contribuições de pesquisa: a DSR deve fornecer contribuições claras e verificáveis nas áreas de artefato de projeto, fundamentos de projeto e / ou metodologias de projeto.
- Rigor da pesquisa: a DSR depende da aplicação de métodos rigorosos tanto na construção quanto na avaliação do artefato de projeto.

- Design como um processo de pesquisa: a busca por um artefato efetivo requer a utilização dos meios disponíveis para atingir os fins desejados enquanto satisfaz as leis do ambiente do problema.
- Comunicação da pesquisa: a DSR deve ser apresentada de forma eficaz tanto para públicos orientados para a tecnologia quanto para orientados para a gestão.

As atividades de muitos projetos DSR agrupam-se naturalmente em fases como as ilustradas na Figura 8. É possível perceber que existem etapas distintas para serem executadas em uma sequência determinada, porém o fluxo de conhecimento vai em ambas as direções. As sugestões feitas com base no conhecimento do problema são desenvolvidas e avaliadas em um processo iterativo até que o artefato demonstre os resultados necessários para. As etapas são:

- Conscientização do Problema: o processo de investigação começa quando o pesquisador toma conhecimento de um problema ou oportunidade de pesquisa (LACERDA *et al.*, 2013). Nesta etapa é definido o problema de pesquisa e o construída a fundamentação teórica (LIMA, 2016);
- Sugestões: segundo Manson (2006), está vinculada às atividades de desenvolver uma, ou mais, alternativas de artefato para a solução dos problemas. Os projetos preliminares estão ligados a uma proposta formal que inclui, normalmente, um projeto experimental. Essa etapa é totalmente criativa, e é nesta fase que diferentes pesquisadores irão chegar a diferentes modelos experimentais. Observa-se que as sugestões para o problema são delineadas a partir da existência de conhecimento/teoria de base sobre o problema (VAISHNAVI; KUECHLER; PETTER, 2004);
- Desenvolvimento: O Desenvolvimento corresponde ao processo de constituição do artefato em si (MANSON, 2006). Nesta etapa, o pesquisador construirá um ou mais artefatos. As técnicas utilizadas variam, dependendo do artefato a ser construído;
- Avaliação: A Avaliação é definida como o processo rigoroso de verificação do comportamento do artefato no ambiente para o qual foi projetado, em relação às soluções que se propôs alcançar (LACERDA *et al.*, 2013);
- Conclusões: Consiste na formalização geral do processo e sua comunicação às comunidades acadêmica e de profissionais (LACERDA *et al.*, 2013).

Figura 8 Diagrama da metodologia geral em Design Science Research



Fonte: Adaptado de (DRESCH, Aline; LACERDA; MIGUEL, 2015; HEVNER *et al.*, 2004 *apud* LIMA, 2016)

### 3.2 Delineamento da Pesquisa

Faz-se necessário antes de introduzir o método desta pesquisa realizar uma desambiguação e enquadrar este trabalho nos conceitos apresentados no referencial. Inicialmente é preciso entender que os insumos desta pesquisa serão: modelos de informações e critério que avaliem um projeto. Entende-se por critério, tanto regras de verificação com resultados booleanos, como indicadores que resultam em valores numéricos. Observando que cada tipo de critério atende uma necessidade específica.

Então, serão usadas as estratégias apresentadas para converter as regras ou indicadores em um formato que possa ser implementado por computador para avaliar os modelos BIM. Para realizar a avaliação serão utilizados os conceitos de *BIM Model Checking* (BMC), ou seja, a checagem do modelo de informações. A forma mais eficiente de checar o modelo é através de regras, serão usadas principalmente regras de classe 4 de Solihin e Eastman (2015) e orientação de Hjelseth e Nisbet (2010a). Esse tipo de regra engloba o cálculo de indicadores e normalmente apresenta como resultados nova informação sobre o projeto baseado no modelo.

Assim, será utilizada a checagem do modelo através de regras, em Dynamo para serem executadas no Revit, para avaliar aspectos do projeto da edificação que prevejam o desempenho do projeto. Busca-se uma metodologia modularizada, com etapas padronizadas que se adequem aos mais diversos tipos de critérios. Assim, propondo um método para o desenvolvimento da ferramenta de forma generalizada. Espera-se que ao realizar a implementação de vários critérios que avaliem desempenhos diferentes chegue-se a um modelo teórico que possa ser replicado.

### **3.3 Etapas Metodológicas**

#### **3.3.1 Conscientização**

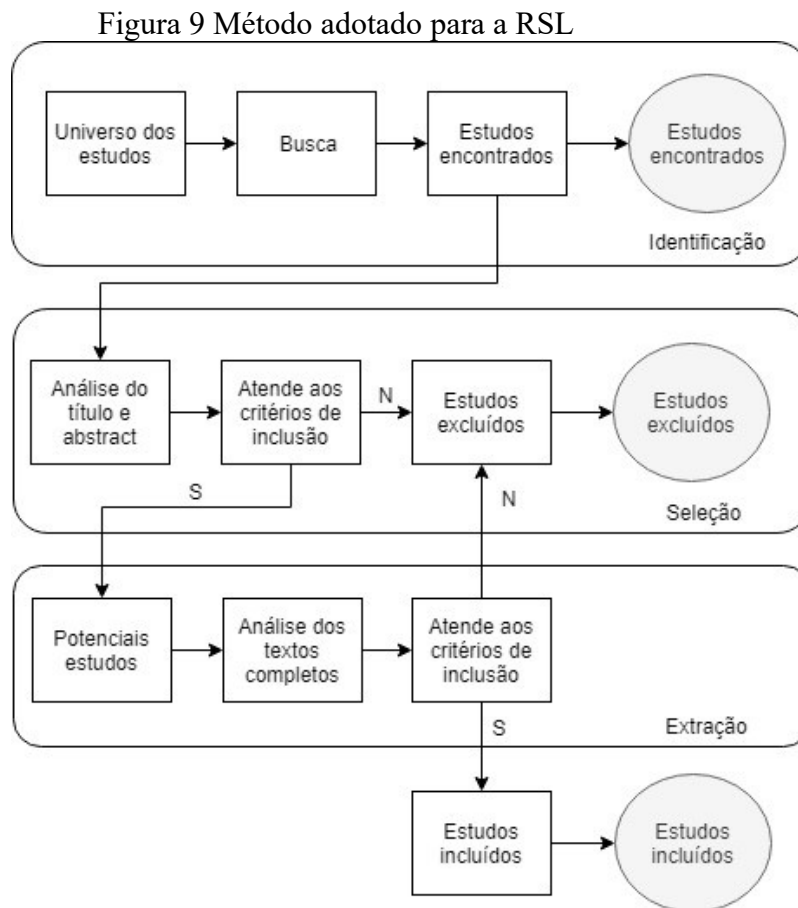
Esta etapa foi realizada para delineamento do problema de pesquisa, posicioná-lo no universo teórico existente e para municiar com ferramentas e conceitos as próximas etapas. O problema atacado nesta pesquisa é a dificuldade de se avaliar projetos de edificações de forma racional e quantitativa, de modo que esta pesquisa tem a intenção de propor uma ferramenta automatizada de avaliação de projetos em BIM. Para tal, a conscientização será realizada em três fases:

- (1) identificação da lacuna teórica e como esta pesquisa se insere nos avanços científicos da área;
- (2) buscar na literatura pesquisas que utilizaram a avaliação e checagem de modelos BIM, para identificar potenciais ferramentas para esta pesquisa.
- (3) realizar uma revisão sistemática da produção científica sobre formas de prever o desempenho de projetos. Identificando as dimensões que serão avaliadas;

Os resultados da primeira e segunda fases da etapa de conscientização se encontra capítulo 2 (referencial teórico), fez-se necessário entender o conceito de BIM e como ele é utilizado pela indústria da construção. Buscou-se apontar quais características são importantes para a pesquisa e em que etapa do empreendimento a ferramenta teria maior utilidade. Procedeu-se com uma revisão que estabelece a teoria para a checagem automatizada de modelo, mostrando o estado da arte da área. Abordou-se sistemas de classificação de regras, metodologias de desenvolvimento e aplicação de checagem automatizada de modelos.

Levantou-se diversos trabalhos que se utilizaram da checagem automatizada de modelos e ferramentas adotadas. Este esforço se justifica em organizar um método eficaz para o desenvolvimento da ferramenta proposta, absorvendo as técnicas que obtiveram os melhores resultados. Os conceitos presentes no Referencial foram fundamentais para a pesquisa.

A última fase de conscientização foi realizada por meio de Revisão Sistemática da Literatura, que consiste em identificar, avaliar e interpretar as pesquisas relevantes relacionadas a um específico problema, tópico, área ou fenômeno de interesse (SONEGO; ECHEVESTE, 2015). Esta segue um procedimento rigoroso, mostrada na Figura 9, onde serão buscados na literatura trabalhos que possam reponder a questão inicial.



Fonte: Adaptado de (DRESCH, A; LACERDA; ANTUNES, 2014)

A RSL tem como objetivo identificar formas de medir o desempenho do projeto de edificações. Assim, buscou-se trabalhos dentro dessa temática que apresentem indicadores de projeto, estes foram organizados de acordo com o aspecto do projeto que avaliam. Foram usadas variações de palavras como “Indicadores de projeto”, “Desempenho do projeto”, “Métricas de avaliação de projeto” e “Performance do projeto” para buscar nas principais bases científicas, com o intuito de conhecer o estado da arte do projeto e avaliação deste desempenho, para estabelecer uma base conceitual. Autores que abordaram este desempenho e, principalmente, como medi-lo foram incluídos nesta revisão. O resultado desta etapa é a

identificação das principais dimensões do projeto que podem ser avaliadas automaticamente e um conjunto de regras/indicadores que possam avaliar as dimensões.

### **3.3.2 Sugestão**

Para tornar a avaliação do projeto possível esta pesquisa desenvolverá uma ferramenta de checagem automatizada de modelos. A solução proposta para o problema é um conjunto de algoritmos implementados em Dynamo que podem ser executados no Autodesk Revit. Os programas serão agrupados em módulos, uma para cada característica avaliada e cada módulo contendo um conjunto de regras a serem automatizadas.

Para entender melhor e verificar a viabilidade da solução proposta foi desenvolvido um protótipo. Foi estudado o desempenho construtibilidade do projeto, como uma forma de trazer a expertise de construção para o processo de projeto. O método usado no protótipo foi uma simplificação do método proposto para a pesquisa: conceituação e revisão sobre o tema; identificação do conjunto de indicadores categorizadas por projeto; desenvolvimento, no Dynamo, de algoritmos para aplicação de regras em um modelo BIM e cálculo dos indicadores; aplicação em dois projetos diferentes para avaliar os resultados.

O protótipo apresentou resultados significativos, foram identificados 13 indicadores e o algoritmo foi capaz de calcular todos a partir de uma análise do modelo BIM. Ao avaliar dos projetos diferentes os indicadores demonstraram resultados diferentes e de acordo com o esperado, mostrando sensibilidade na análise. Assim, a proposta de solução inicial vai para a próxima etapa com uma maior confiabilidade.

### **3.3.3 Desenvolvimento**

A partir do arcabouço teórico construído na primeira etapa, pode-se buscar aplicações práticas para esta pesquisa. Para o desenvolvimento da ferramenta precisamos cumprir duas etapas bem diferenciadas, estas foram baseadas no item 2.2 Avaliação Automatizada de Modelos. Compreendem basicamente uma etapa de interpretação da métrica, onde sua descrição formal é traduzida para uma linguagem de verificação que possa ser codificada, e uma etapa de desenvolvimento do algoritmo onde as regras são automatizadas para avaliar o modelo. A seguir são explicitados os procedimentos em dois tópicos Medir e Implementar.

### 3.3.3.1 Medir

Nessa primeira etapa o texto da regra ou a fórmula dos indicadores precisam ser escritos em termos de parâmetros do modelo ou ainda na forma de operações, como operações matemáticas com os dados ou operações geométricas nos objetos 3D. Avaliar o projeto para medir o desempenho é o objetivo final e para isso é preciso estabelecer unidades e padrões. Assim, para cada regra será estabelecido um padrão de comparação, este pode ser uma condição para aprovar ou reprovar o projeto, mas pode também ser um valor de comparação que indique a situação do projeto avaliado.

O conjunto de regras, se necessário, passou por um processo baseado em RASE, TIO e Tx3, que prepara cada regra para o desenvolvimento baseado no trabalho de Hjelseth (2015), este processo foi descrito no tópico 2.2.3 Transformação de regras. As regras após o processo estão tabuladas e organizadas de uma forma que possam ser implementadas por computador. Sem nenhuma ambiguidade nas descrições ou falta de precisão nos parâmetros.

Assim, com a regra devidamente especificada pode-se identificar os requisitos que o modelo a ser analisado precisará cumprir. Estes correspondem as características geométricas, paramétricas e de dados necessários para alimentar a checagem da regra que irá compor a avaliação do projeto. Um checklist de requisitos será elaborado para cada módulo da ferramenta, para garantir que não haja erros por falta de informação.

O exposto nesta seção compreende a primeira e segunda etapa de Eastman et al. (2009), Tradução da Regra e Preparação do Modelo. Esta etapa tem três produtos: lista de regras prontas para serem implementadas, parâmetros de comparação para cada regra e checklist de requisitos do modelo.

### 3.3.3.2 Implementar

Nesta etapa serão realizados os procedimentos necessários para cumprir os dois últimos passos propostos por Eastman *et al.* (2009), execução da regra e relatórios. Inicia com o desenvolvimento dos algoritmos e a codificação em si. Neste trabalho elaboraram-se rotinas com o uso de programação visual no Autodesk® Dynamo, assim, automatizando a aquisição de informações do modelo, checagem de regras e cálculo dos devidos critérios. A automatização dos cálculos foi elaborada na interface entre o Revit e o Dynamo, as regras foram interpretadas na etapa anterior referenciando as necessidades da análise aos dados disponíveis para análise no modelo BIM.

Escolheu-se uma abordagem paramétrica para a análise, de tal forma que se prioriza a avaliação dos parâmetros e informações contidas nos elementos do modelo em detrimento a



uma análise geométrica. Esta escolha foi feita devido ao grande custo computacional de operações geométricas, porém regras mais complexas ainda fizeram uso desse tipo de abordagem.

Buscou-se tornar a rotina de cálculo o mais simples possível, suas principais tarefas são ler o modelo para coletar dados (paramétricos e geométricos), então manipular e utilizar estes dados para calcular os indicadores ou compará-los com requisitos pré-estabelecidos. Por fim, compilaram-se os resultados escrevendo-os de forma estruturada em um formato de saída pré-definido, onde estes são expostos graficamente para facilitar a interpretação dos indicadores.

Destaca-se que a rotina final será implementada no Reprodutor do Dynamo, uma interface dentro do Revit que permite a utilização de rotinas sem a necessidade conhecimentos sobre programação visual. Tornando, assim, a ferramenta proposta aqui acessível para todo tipo de usuário, pois funcionará através de formulários e botões programados. Permite, ainda, que os cálculos aconteçam iterativamente as modificações do usuário, podendo este avaliar instantaneamente o impacto das alterações.

#### **3.3.4 Avaliação**

Esta etapa compreende um processo de validação, este será dividido em duas formas de avaliação. A primeira uma Avaliação Experimental formado por um Experimento Controlado, onde estuda-se o artefato em um ambiente controlado para verificar suas qualidades, e uma Simulação, onde executa-se o artefato com dados artificiais (HEVNER *et al.*, 2004). Os programas serão executados em modelos de teste, onde os resultados são conhecidos. Assim, simula-se os resultados com dados artificiais para comparação direta, podendo, assim, avaliar o desempenho do programa. Essa avaliação alimenta diretamente a Sugestão e o Desenvolvimento, pois a cada resposta negativa do resultado do programa é necessário sugerir uma nova abordagem e desenvolver um novo código. Este processo se repete no desenvolvimento de cada regra até obter-se um conjunto maduro de códigos. Esta etapa visa aprimorar o processo interno do programa, otimizando algoritmos e cálculos.

A avaliação será um Teste Funcional (Black Box): executar as interfaces do artefato para descobrir possíveis falhas e identificar defeitos (HEVNER *et al.*, 2004). O processo é cíclico e iterativo, pois a cada teste do uso da ferramenta usado para alimentar melhorias no programa.

#### **3.3.5 Conclusão**

A conclusão é o fechamento do ciclo de desenvolvimento do artefato, nela foi reavaliado se cada objetivo específico conseguiu ser cumprido e as contribuições práticas e teóricas dessa pesquisa. Nessa etapa também foi indicado a maneira recomendada de utilização dessa ferramenta, o intuito é apontar o que cada módulo pode contribuir e em que ponto do desenvolvimento do projeto ele pode ser executado.

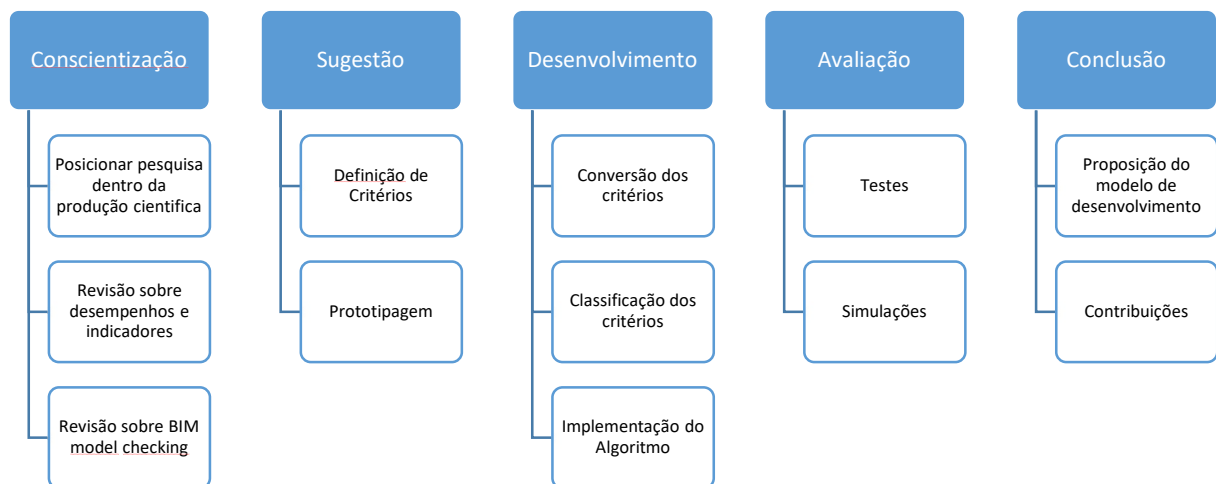
Nesta etapa o processo utilizado para criar os algoritmos foi generalizado para que pudesse abranger qualquer tipo de desempenho em haja regras de medição disponíveis. O intuito foi gerar um modelo teórico para a criação de uma ferramenta de avaliação de projeto baseado na checagem automatizada de modelos BIM. Esta foi a contribuição teórica final desta pesquisa.

É preciso, ainda, ao final da DSR comunicar os resultados da pesquisa (PEFFERS *et al.*, 2008), para esta se propõe a formalização do conhecimento em forma de artigos científicos a serem publicados ao final do desenvolvimento.

### 3.4 Delineamento geral

Neste tópico busca-se relacionar a metodologia exposta anteriormente com os objetivos que norteiam todo o trabalho. Para facilitar a compreensão geral do método a Figura 10, resume as etapas e mostras a subatividades necessárias.

Figura 10 Resumo do Método



Fonte: Autor

O primeiro objetivo é “Identificar dimensões do desempenho de uma edificação para serem avaliadas por meio do modelo de informação”, este foi cumprido pela etapa de

Conscientização, quando se conceitua características de projeto que afetam o desempenho. Neste trabalho foram selecionadas características que serão exploradas no decorrer da pesquisa.

O segundo objetivo é “Selecionar medidas relativas as dimensões especificadas a serem avaliadas”. Este é respondido pela lista de regras que foram implementadas, um produto da etapa de Sugestão. Estas regras se baseiam na conceituação feita para representar uma forma de quantificar os desempenhos do projeto na característica avaliada.

O objetivo “Desenvolver algoritmo para a avaliação das medidas através programação visual”. É plenamente atingido na etapa de desenvolvimento, pois, baseado nos estudos realizados na RSL da conscientização, foi implementado um conjunto de programas em programação visual que automatizam os sistemas de medida de desempenho proposto.

O último objetivo é “propor modelo para desenvolvimento de algoritmos de avaliação de projetos”. Foi discutido durante os testes na avaliação e respondido em uma reflexão sobre o desenvolvimento e o uso da ferramenta na conclusão. Assim, propõe-se um modelo que possa ser generalizado para desenvolver ferramentas de avaliação automatizada de projetos.

Em conclusão, por meio da metodologia proposta alcança-se cada um dos objetivos específicos e dessa forma atinge-se o objetivo geral de “desenvolver uma ferramenta automatizada para avaliação de desempenhos em tempo real do projeto de edificações”. A Tabela 7 sintetiza o método, as atividades da DSR e o resultados de cada uma.

Tabela 7 Síntese do Método

<b>Etapa da DSR</b>	<b>Atividades</b>	<b>Resultado Esperado</b>
<b>Conscientização</b>	Posicionar pesquisa dentro da produção científica	Principais conceitos BIM e sua utilização para checagem automatizada de projetos
	Revisão sobre desempenhos e indicadores	Conjunto de indicadores de projeto categorizados
	Revisão sobre BIIM model checking	Compilação sistemas de checagem, seus usos e ferramentas
<b>Sugestão</b>	Definição de indicadores	Proposição das dimensões do projeto a serem avaliadas e quais os indicadores
	Prototipagem	Algoritmo de avaliação de construtibilidade de projeto
<b>Desenvolvimento</b>	Medir: Identificação e preparação dos	Indicadores e métricas parametrizados

	critérios por meio da metodologia Tx3.	em formato de regras tabulares, prontos para a implementação
	Medir: Propor classificação de critérios combinando a visão deste trabalho com o discutido na literatura	Sistema de classificação de critérios que oriente o desenvolvimento dos algoritmos de avaliação.
	Implementar: Desenvolver, por meio de programação visual, rotinas que calculem os indicadores selecionadas	Programas intuitivos e eficientes que avaliem as métricas automaticamente.
<b>Avaliação</b>	Realizar simulações e testes controlados do programa em um modelo com resultados conhecidos	Ciclo de melhoria do programa, identificando e corrigindo erros relativos ao processo interno do programa
	Utilizar o programa em um modelo diferente	Apontar erros em variações do modelo, readequando os algoritmos para generalizar o uso
<b>Conclusão</b>	Proposição do modelo de desenvolvimento	Reflexão sobre o processo de desenvolvimento para generalização
	Contribuições	Contribuições teóricas e práticas desta pesquisa

Fonte: Autor

## 4 RESULTADOS

Os resultados desta pesquisa mostram as etapas de desenvolvimento do Artefato. Serão apresentados nesta seção seguindo a estrutura definida. Assim, se constrói logicamente, partindo do levantamento dos critérios, avaliando a viabilidade na prototipagem, definindo os critérios e indicadores a serem avaliados, desenvolvendo os algoritmos a serem utilizados na ferramenta, testando e validando o artefato. Então, finalmente se constrói o modelo de desenvolvimento.

### 4.1 Conscientização: Levantamento dos Critérios

#### 4.1.1 *Revisão Sistemática: Métricas de Avaliação de Projeto*

A Revisão Sistemática utilizou variações das entradas: “Indicadores de projeto”, “Desempenho do projeto”, “Métricas de avaliação de projeto” e “Performance do projeto”. Foram utilizadas estas entradas em inglês, pois é o idioma mais representativo nas bases pesquisadas. As bases pesquisadas foram: Engineering Village, Scopus, Web of Science e Science Direct. Optou-se por não restringir por tipo ou data de publicação, pois trabalhos antigos poderiam conter propostas de métodos de avaliação úteis que não foram implementadas devido a complexidade, o que pode ser contornado utilizando BIM. As buscas retornaram um total de 3051 artigos, sendo 199 aceitos para a seleção e destes 20 aceitos na etapa de extração.

Durante a Identificação removeu-se os artigos duplicados, os não duplicados foram para a próxima etapa. Na Seleção foi analisado o título e resumo dos trabalhos, apenas os que demonstravam formas de avaliar a performance do empreendimento baseado no seu projeto ou características que podem ser extraídas do design foram selecionados. Nesta etapa foram retirados da amostra artigos que estavam fora do tema do trabalho, grande parte deles nesta etapa eram de outras áreas do conhecimento ou não se enquadravam em trabalho sobre análise ou avaliação de projeto.

A última etapa foi a Extração, nessa os 199 artigos foram lidos na íntegra. Foi buscado nos trabalhos como o autor propôs avaliar os projetos, como ele descreve as métricas e regras e, baseando-se, nessas informações, categorizando o indicador de forma mais abrangente de acordo com qual aspecto do projeto ele avalia. Aqui, a maior parte dos artigos se propunha a analisar a performance do empreendimento, porém foram rejeitados trabalhos que apresentavam métricas que não podiam ser analisadas baseando-se no projeto diretamente. Entre os artigos rejeitados estavam avaliações de performance da empresa construtora ou de algum sistema da empresa, além disso, muitos artigos abordavam eficiência

energética, porém essa avaliação exigia informações do uso da edificação, não disponível diretamente no modelo.

Os critérios encontrados nos trabalhos serão analisados e selecionados na etapa de Desenvolvimento deste trabalho.

Tabela 8 Artigos analisados

<b>Título</b>	<b>Journal</b>	<b>Ano</b>	<b>Autores</b>
A Building Information Management (BIM) Framework and Supporting Case Study for Existing Building Operations, Maintenance and Sustainability	Procedia Engineering	2015	McArthur, J. J.
A method of optimizing fenestration design for daylighting to reduce heating and cooling loads in offices	Journal of Civil Engineering and Management	2014	Seinre, Erkki; Voll, Hendrik
A new systemic approach to improve the sustainability performance of office buildings in the early design stage	Energy and Buildings	2015	Kreiner, H.; Passer, A.; Wallbaum, H.
A review of indices for assessing visual comfort with a view to their use in optimization processes to support building integrated design	Renewable and Sustainable Energy Reviews	2015	Carlucci, Salvatore; Causone, Francesco; De Rosa, Francesco; Pagliano, Lorenzo
Assessing environmental performance in early building design stage: An integrated parametric design and machine learning method	Sustainable Cities and Society	2019	Feng, Kailun; Lu, Weizhuo; Wang, Yaowu
BIM application to building energy performance visualisation and management Challenges and potential	Energy and Buildings	2017	Cook, Malcolm; Gerrish, Tristan; Johnson, Mark; Lowry, Christine; Phillip, Mark; Ruikar, Kirti
BIM-based assessment metrics for the functional flexibility of building designs	Automation in Construction	2019	Cavalliere, Carmine; Dell'Osso, Guido Raffaele; Favia, Fausto; Lovicario, Marco
Building Envelope Thermal Performance Assessment Using Visual Programming and BIM, based on ETVV requirement of Green Mark and GreenRE	International Journal of Built Environment and Sustainability	2017	Ahmad, Mohd Hamdan; Lim, Yaik Wah; Samuel, Williams Opeyemi; Seghier, Taki Eddine
Building Information Modeling (BIM) and Safety: Automatic Safety Checking of Construction Models and Schedules	Automation in Construction	2013	Eastman, Charles M.; Lee, Jin Kook; Teizer, Jochen; Venugopal, Manu; Zhang, Sijie
Computer representation of building codes for automated compliance checking	Automation in Construction	2017	Günaydın, H. Murat; Macit İlal, Sibel
Green maintainability performance indicators for highly sustainable and maintainable buildings	Building and Environment	2019	Asmone, Ashan Senel; Chew, Michael Y.L.; Conejos, Sheila
Implementation of an interoperable process to optimise design and construction phases of a residential building: A BIM Pilot Project	Automation in Construction	2016	Ciribini, A. L.C.; Mastrolemba Ventura, S.; Paneroni, M.
Indoor Walkability Index: BIM-enabled approach to Quantifying building circulation	Automation in Construction	2019	Lee, Jin Kook; Shin, Jaeyoung

Título	Journal	Ano	Autores
Integration of parametric design into modular coordination: A construction waste reduction workflow	Automation in Construction	2018	Banihashemi, Saeed; Hosseini, M. Reza; Tabadkani, Amir
Investigation of the Sustainability and Resilience Characteristics of Buildings Including Existing and Potential Assessment Metrics	AEI 2017: Resilience of the Integrated Building - Proceedings of the Architectural Engineering National Conference 2017	2017	Bilec, M. M.; Chhabra, J. P.S.; Hasik, V.; Warn, G. P.
Offsite construction: Developing a BIM-Based optimizer for assembly	Journal of Cleaner Production	2019	Aigbavboa, Clinton; Akinade, Olugbenga O.; Gbadamosi, Abdul Quayyum; Mahamadu, Abdul Majeed; Mahdjoubi, Lamine; Manu, Patrick; Oyedele, Lukumon O.
Optimising building sustainability assessment using BIM	Automation in Construction	2019	Bragança, Luís; Carvalho, José Pedro; Mateus, Ricardo
Reducing embodied carbon during the design process of buildings	Journal of Building Engineering	2015	Häkkinen, Tarja; Jung, Nusrat; Kuittinen, Matti; Ruuska, Antti
Ventilation, thermal and luminous autonomy metrics for an integrated design process	Building and Environment	2018	Brager, Gail; Ko, Won Hee; Levitt, Brendon; Schiavon, Stefano
Waste minimisation through deconstruction: A BIM based Deconstructability Assessment Score (BIM-DAS)	Resources, Conservation and Recycling	2015	Ajayi, Saheed O.; Akinade, Olugbenga O.; Alaka, Hafiz A.; Bello, Sururah A.; Bilal, Muhammad; Owolabi, Hakeem A.; Oyedele, Lukumon O.

Fonte: Autor

Os artigos selecionados abordavam diversas facetas do projeto de uma edificação, todos tem em comum uma formalização através de indicadores ou regras para medir o desempenho baseado no projeto. Para melhor entendê-los foram levantados todos os critérios propostos pelos autores e estes foram divididos em categorias de acordo com a dimensão que avalia e classificados de acordo com e uma classificações proposta neste trabalho baseado na metodologia de desenvolvimento em Dynamo. Estes resultados estão expostos no Apêndice

Alguns dos trabalhos lidos apresentavam critérios explícitos, porém exigiam dados de planejamento ou dados externos do empreendimento o que inviabilizou a adição destes critérios ao artefato. Este foram utilizados em discussões ou definições da revisão, pois abordavam diretamente o tema de avaliação automatizada de modelos. De maneira geral

percebeu-se como uma dificuldade encontrar critérios que utilizassem apenas informações do projeto, pois esta é uma premissa deste artefato, muitos envolviam coletar informações da construtora, do ambiente da obra ou dos stakeholders envolvidos no projeto, fugindo do escopo.

#### 4.1.2 Consulta aos especialistas

Buscou-se entender o posicionamento dos profissionais sobre os critérios de avaliação, identificando dificuldades práticas que pudessem ser solucionadas através do artefato proposto nessa pesquisa. Para tal, foram consultados especialistas, foi pedido para os mesmos indicarem que critérios seriam úteis para serem automatizados na ferramenta. O perfil dos especialistas consultados está na Tabela 9.

Tabela 9 Perfil dos Especialistas

Especialista	Experiencia
Arquiteto	Diretor executivo de escritório de arquitetura, 50 anos de experiência no desenvolvimento de projetos residenciais, comerciais e públicos
Arquiteto	Coordenação de Desenvolvimento Urbano de Fortaleza, núcleo elaboração do plano diretor da cidade
Arquiteto e Engenheiro de Produção	Gerente de projetos, especialista em BIM com 16 anos de atuação em incorporadora.
Engenheiro Civil	Gerente Técnica especializada em Lean e Green, 7 anos de experiência
Arquiteto	20 anos de experiência, coordenação de projetos

Fonte: Autor

Os especialistas mostraram uma visão voltada para a adequação do projeto a regulações e normas, justificada pela necessidade de atender a essas necessidades em cada projeto que participam. Logo, os critérios sugeridos pelos especialistas eram provenientes da regulação municipal de aprovação, da norma de desempenho NBR 15575, de selos de sustentabilidade (Fator Verde), da regulação do corpo de bombeiros e da norma de



acessibilidade. As sugestões destes foram avaliados na etapa de definição dos critérios, a adicionados ao estudo.

## **4.2 Protótipo: Construtibilidade**

### **4.2.1 Conceituação**

As decisões dos arquitetos condicionam, de alguma maneira, o comportamento e o desempenho de todo o edifício, tanto funcional como economicamente (MASCARÓ, 2010). No entanto nenhuma outra indústria a responsabilidade pelo projeto está tão distante da reponsabilidade pela produção (EMMERSON, 2008). Nesta visão, os aspectos construtivos na concepção dos projetos deveria ser um dos tópicos abordados veemente por arquitetos e engenheiros. A construtibilidade depende de que projetistas e construtoras sejam capazes de visualizar todo o processo de construção com o olhar um do outro. Esta percepção parece não ter mudado durante os anos, visto que Rodriguez e Heineck (2003) destacam a necessária integração entre projeto e obra para a melhoria de construtibilidade. Isto é confirmado no estudo de Barbosa e Andery (2015), que concluem que a melhoria da execução das obras também está vinculada à introdução progressiva de conceitos de engenharia simultânea e a uma postura de coordenação de projetos pró ativa.

Na busca pelo entendimento do fator construtivo, organizações e estudiosos discutiram seu conceito. Para CIRIA (1983) a construtibilidade seria a amplitude na qual o projeto de um edifício favorece a facilidade de construção, considerado requisitos do edifício concluído. Para Oliveira (1994), a construtibilidade é entendida como a habilidade ou facilidade de construção. A Construction Industry Institute (CII, 2012) compreende construtibilidade como o uso ótimo do conhecimento e experiência em planejamento, contratação, projeto e atividades de canteiro para atingir os objetivos globais do empreendimento. Percebe-se que o conceito foi concebido como sinônimo de facilitar a construção através do projeto, e se vem ampliando. Assim, a construtibilidade envolve a integração do conhecimento e experiência construtiva durante as fases de concepção, o planejamento, o projeto e a execução da obra, visando a simplificação das operações construtivas através do conhecimento da tecnologia construtiva a ser adotado no empreendimento (MYDIN *et al.*, 2011).

O objetivo da construtibilidade é melhorar a eficiência dos processos de construção ao desenvolver projetos sensitivos, que consideram aspectos da execução (HON; GAIRNS; WILSON, 1988). Este processo pode ser feito pela alteração de atributos em

qualquer um dos projetos, como no caso de um projeto estrutural, que pode promover uma solução de laiaute que corrobora com menos congestionamentos na execução, com maiores tolerâncias e menores densidades de armaduras (MYDIN *et al.*, 2011). Dentre alguns fatores que influem na construtibilidade, se pode citar o grau de simplificação do projeto; a extensão da padronização adotada na empresa, a sequência executiva e interdependência entre atividades; a acessibilidade aos espaços de trabalho; e a comunicação entre projeto e obra (OLIVEIRA, R. R., 1994).

Os resultados esperados da implementação da construtibilidade é uma construção eficiente e efetiva, gerando economia mantendo os requisitos de qualidade especificados pelos clientes (GRIFFITH; SIDWELL, 1997). É certo que o aumento deste fator traz benefícios para critérios de custo, prazo e qualidade da obra (LAM; WONG, 2009), destacando-se o aumento de produtividade, redução de retrabalhos e trabalho intensivo, e satisfação de *stakeholders* (ICC, 2012). A construtibilidade também pode ser tratada como fator estratégico, como na redução de dependência de mão de obra ostensiva na indústria da construção (LAM; WONG; TIONG, 2006).

Os sistemas de medição de desempenho são especialmente importantes na indústria da construção (CÂNDIDO; LIMA; BARROS NETO, 2016) onde a primeira etapa parte da seleção de indicadores (FRANCO-SANTOS *et al.*, 2007). Indicadores podem ter o papel de demonstrar o desempenho de uma organização, atuar no controle de um processo em relação à um padrão, no estabelecimento de metas, e agindo na motivação de pessoas (FORMOSO *et al.*, 1996).

Os indicadores são amplamente utilizados na medição de produtividade na construção civil (COSTA *et al.*, 2006) a qual se relaciona diretamente com aspectos construtivos dos projetos. Sendo baseada em projetos, é adequado que a medição de desempenho da construção seja concentrada nas suas performances. Permitindo assim que os seus atributos possam ser aplicados na fase de pré-construção, particularmente para funcionalidade orientada para a construção (LAM; WONG, 2009).

Informações espaciais são necessárias para a análise de construtibilidade, onde cálculos complexos na obtenção destas são evitados com o uso de dados extraídos facilmente do modelo BIM (KHEMLANI, 2004). Portanto, o modelo BIM facilita testes de projeto e sequenciamento de atividades para alcançar melhor construtibilidade (ZHANG, C. *et al.*, 2016), e com a adoção de métodos de programação, é possível automatizar a extração da construtibilidade destes projetos. A presente avaliação propõe uma ferramenta com o uso do

VPL para a criação de rotinas na extração de indicadores que avaliam a construtibilidade de projetos de construção civil antes da fase de execução.

#### 4.2.2 Sugestão de Indicadores

Uma revisão da literatura foi conduzida para buscar indicadores e critérios. Resultando em 13 itens que suportam a avaliação da construtibilidade do projeto. Estes são apresentados na Tabela 10.

Tabela 10 Indicadores para construtibilidade

N	Autor	Nome	Equação	
<b>ARQUITETURA</b>				
1	(LANTELME, E. M. V., 1994; MASCARÓ, 2010)	Índice de Compacidade	$IC = 2 * \sqrt{\pi * Apav} / Pp$	<i>Apav: Área do pavimento</i> <i>Pp: Perímetro do pavimento</i>
		Representa inversamente a complexidade geométrica do perímetro do pavimento. Quanto mais distante de um quadrado (0,84), menor será o índice, e menor a construtibilidade.		
2	(LANTELME, E.; OLIVEIRA; FORMOSO, 1995; NARLOCH, 2015)	Índice de Área Molhada	$IAm = Am / Apav$	<i>Am: Área molhada no pavimento</i>
		Áreas molhadas exigem maior quantidade de serviços devido a impermeabilização, testes e uso de cerâmicas nas alvenarias.		
3	(LANTELME, E.; OLIVEIRA; FORMOSO, 1995))	Densidade de Paredes	$DP = Apar / Apavt$	<i>Apar: Área de paredes</i>
		Este índice tem o objetivo de verificar o grau de otimização da compartimentação do pavimento tipo		
4	(MASCARÓ, 2010; NARLOCH, 2015; OLIVEIRA, M.; FREITAS, 1996)	Índice de Fachadas	$IF = Af / Apavt$	<i>Af: Área fachadas do pav. tipo</i>
		Os planos verticais das fachadas são mais difíceis e custosos de serem construídos. O indicador revela a proporção de fachadas em relação ao pavimento tipo.		
5	(LANTELME, E.; OLIVEIRA; FORMOSO, 1995)	Densidade de Esquadrias	$DE = Aes / Apar$	<i>Aes: Área de esquadrias (portas e janelas)</i>
		Esquadrias demandam maior número de serviços e aumentam a complexidade construtiva		
6	(LANTELME, E.; OLIVEIRA; FORMOSO, 1995)	Índice de Padronização de Esquadrias	$IE = Edif / Etot$	<i>Edif: Número de Esquadrias Diferentes</i> <i>Etot: Número Total de Esquadrias</i>
		A maior diversidade de esquadrias influi na complexidade do projeto, na compra, e no planejamento e operação de execução.		
<b>ESTRUTURA</b>				
7	(JARKAS, 2010)	Densidade de Pilares	$DPi = Abp / Np$	<i>Abp: Área da base de pilares</i> <i>Np: Número Pilares ajustado</i>
		Pilares restringem a movimentação na obra e aumentam pontos de fundações.		
8	(JARKAS, 2010)	Densidade de Vigas	Se $Cv / Apav \leq 0.45$ (1) Se não (2) (1) $DPv = Cv / (0.45 * Apav)$ ou (2) $DPv = 2 - Cv / (0.45 * Apav)$ ou	<i>Cv: Comprimento de Vigas</i> <i>Apav: Area do Pavimento</i>
		Este índice representa a eficiência do projeto. Quanto menor este valor, menores serão as complexidades dos serviços de formas e concretagem, também reduzindo interferências.		
9	(JARKAS, 2010)	Índice de Padronização de Pilares	$IPp = Pdif / Ptot$	<i>Pdif: Pilares diferentes</i> <i>Ptot: Pilares totais</i>
		Este índice considera a complexidade na individualidade de tipos estruturais, através		

		da relação de quantidades de pilares diferentes em suas seções transversais e o número total de pilares.	
<b>10</b>	(JARKAS, 2010)	Índice de Padronização de Vigas $IPv = Vdif / Vtot$	<i>Vdif: Vigas diferentes</i> <i>Vtot: Vigas totais</i>
		Este índice considera a complexidade na individualidade de tipos estruturais, através da relação de quantidades de vigas diferentes em suas seções transversais e o número total de vigas.	
<b>11</b>	(JARKAS, 2010)	Índice de Padronização de Lajes $IPl = \frac{Ldif}{Ltot}$	<i>Ldif: Lajes diferentes</i> <i>Ltot: Lajes totais</i>
		Este índice considera a complexidade na individualidade de tipos estruturais, através da relação de quantidades de lajes diferentes em suas seções transversais e o número total de lajes.	
<b>INSTALAÇÕES</b>			
<b>12</b>	(MATTOS, 2015; PALIARI; SOUZA, 2008)	Índice de Tubulações $ITub = Ctub / Apriv$	<i>Ctub: Comprimento de Tubulações</i> <i>CApriv: Comprimento de Tubulações</i>
		Este índice relaciona com a densidade de tubulações na área privativa do pavimento. De modo que quanto maior este índice, menores são as produtividades destes serviços.	
<b>13</b>	(LANTELME, E. M. V., 1994; MATTOS, 2015)	Índice de Conduítes $ICond = Pontos / Ccond$	<i>Pontos: Pontos de Elétricos</i> <i>Ccond: Comprimento de Conduítes</i>
		Lantelme propõe o índice de pontos por comprimento de conduíte, devendo representar a performance do projeto e melhor aproveitamento dos eletrodutos.	

Fonte: Autor

### 4.2.3 Implementação

A estrutura desenvolvida se tornou um conjunto massivo de rotinas, Figura 11, o que dificultou sua visualização. Então, serão destacadas as principais rotinas utilizadas e em quais indicadores elas foram necessárias.

**Rotina de coleta de parâmetros:** O nó “Categories” é utilizado para selecionar a categoria de interesse, ele alimenta o nó “All Elements of Category” que coleta todos os elementos da categoria escolhida. Os elementos, então, abastecem o nó “Element.GetParameterValueByName” que também necessita da especificação textual do parâmetro desejado para retornar uma lista com os parâmetros. Esta rotina coleta parâmetros de instância, se for necessário a coleta de parâmetros de tipo o nó “Element.ElementType” precisa ser aplicado antes do nó de coleta.

**Rotina de filtragem:** Partindo da lista que precisa ser filtrada, ela é conectada a um nó de verificação que retorna uma lista de booleanos, esta juntamente com a lista a ser filtrada alimentam o nó “List.FilterByBoolMask” que retorna duas novas listas, uma com os valores em que o teste é verdadeiro e outra para os testes falsos.

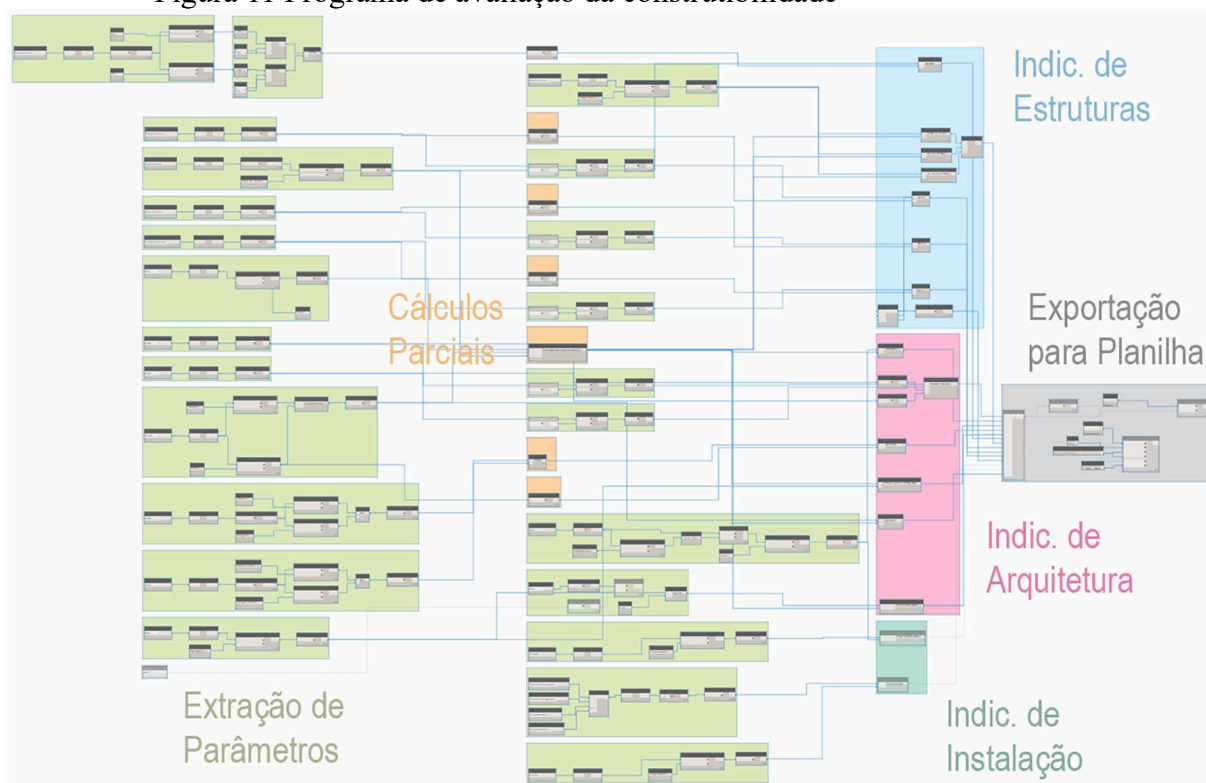
**Rotina de soma e contagem:** O nó “Math.Sum” recebe uma lista de valores e retorna a soma cumulativa. Já o nó “List.Count” conta a quantidade de valores em uma lista.

**Rotina condicional:** O nó “If” permite fazer testes por condição, ele precisa ser alimentado com um “Code Block” contendo o teste e as respostas para verdadeiro e para falso.

**Rotina de cálculo:** O Dynamo permite através do nó “Code Block” a implementação de equações específicas. Este foi utilizado para calcular os indicadores com a alimentação dos parâmetros coletados.

**Rotina de exportação:** Os resultados dos indicadores alimentam o nó “List Create” que os une em uma lista, que é conectada ao nó “Data.ExportExcel”. Para escrever a lista em uma planilha Excel é preciso fornecer ao nó o número da linha e coluna, o nome da planilha, o caminho do arquivo no sistema e um booleano para autorizar que os dados seja sobrescritos. O programa também permite a exportação via arquivo .csv, implementada na rotina destes indicadores resultou em um programa extenso

Figura 11 Programa de avaliação da construtibilidade



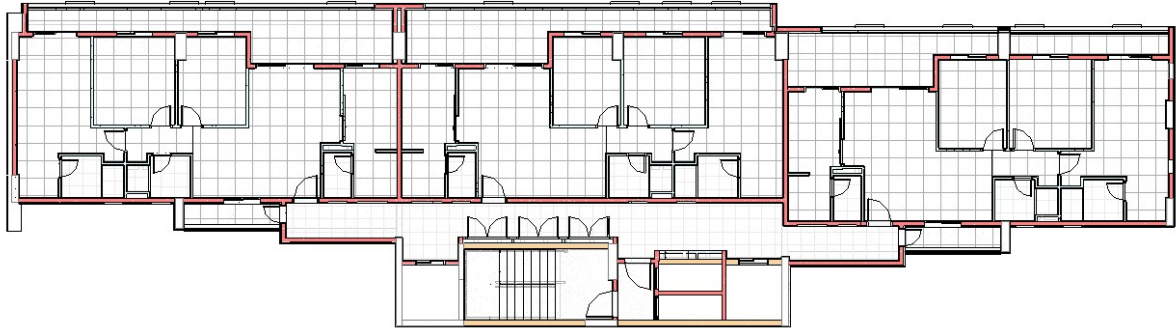
Fonte: Autor

#### 4.2.4 Avaliação

A rotina foi aplicada em dois projetos distintos para avaliação. O primeiro projeto analisado (Figura 12) é um edifício residencial de luxo de uma única torre com dois pisos de garagem, 15 pavimentos tipos e 1 cobertura. Possui três apartamentos por andar, com

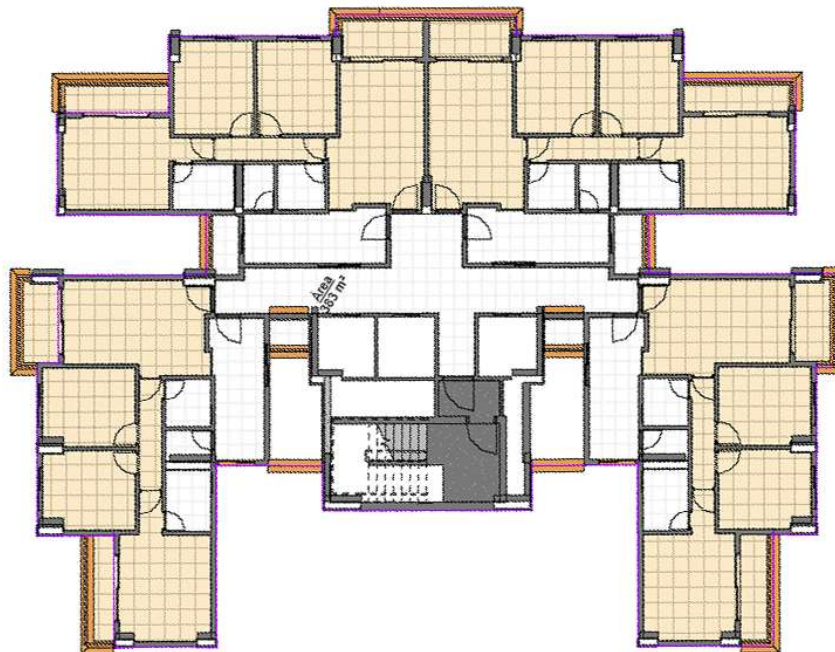
aproximadamente 90 m<sup>2</sup> cada. O segundo projeto analisado (Figura 13) possui duas torres com 22 andares cada e 4 apartamentos por andar. Apenas uma das torres foi selecionada, possuindo apartamentos com 95m<sup>2</sup>.

Figura 12 Projeto 1



Fonte: Autor

Figura 13 Projeto 2



Fonte: Autor

Fez-se uma comparação entre os resultados, levando em conta a natureza de cada indicador. Os procedimentos de coleta e cálculo foram realizados com baixo esforço e em reduzido tempo, em torno de 1 min por execução. Os resultados são ilustrados na Tabela 11, no artigo em si estes são discutidos com mais profundidade.

Tabela 11 Resultados dos indicadores de construtibilidade

N	Indicador	Orientação	Projeto 1	Projeto 2	Diferença	%
<b>ARQUITETURA</b>						
1	Índice de Compacidade	↑	0,55	0,51	0,05	8,4%
2	Índice de Área molhada	↑	0,22	0,14	0,08	36,8%
3	Densidade de Paredes	↓	0,10	0,12	-0,01	-14,3%
4	Índice de Fachadas	↓	1,02	1,10	-0,08	-8,1%
5	Densidade de Esquadrias	↓	0,20	0,15	0,05	25,8%
6	Índice de Padronização de Esquadrias	↑	0,26	0,18	0,09	33,3%
<b>ESTRUTURA</b>						
7	Densidade de Pilares	↓	0,30	0,23	0,07	23,0%
8	Densidade de Vigas	↓	0,83	0,54	0,29	35,2%
9	Índice Padronização Pilares	↓	0,65	0,27	0,37	57,9%
10	Índice Padronização Vigas	↓	0,29	0,13	0,16	55,5%
11	Índice Padronização Lajes	↓	0,29	0,15	0,14	47,5%
<b>INSTALAÇÕES</b>						
12	Índice Tubulações por Área Molhada	↓	1,91			
13	Índice de Pontos por Eletroduto	↑	0,24			

Fonte: Autor

Os indicadores de estrutura apresentaram uma diferença perceptível entre os projetos. O segundo projeto apresenta simetria na direção vertical, e a solução estrutural adotada é mais compacta que a do projeto 1, que não apresenta simetria. Além disso, a tipologia das lajes dos projetos é diferente, enquanto no primeiro são lajes maciças e o segundo nervuradas.

O projeto 2 apresenta resultados melhores no que tange a construtibilidade estrutural. Nos indicadores de padronização este aproveita a simetria e reduz as variações nas seções dos elementos estruturais. Os indicadores de densidades mostram que os pilares apresentam resultados mais aproximados aos das vigas, isto devido à similaridade do carregamento vertical. Já no primeiro projeto, suas vigas vencem vãos maiores, algumas modeladas como protendidas, o que afeta o indicador negativamente devido sua dimensão.

#### 4.2.5 Avaliação como protótipo

A programação da rotina se deu de maneira fluida e rápida, comprovando a simplicidade na sua composição. Esta característica permite que usuários proponham soluções mais abrangentes do que com o uso apenas do software de modelagem autoral, ou que demandariam muitas operações, sem ainda a capacidade de automatizar tais processos. A interface também avisa sobre erros na rotina, facilitando o trabalho na sua construção, e promovendo maior confiabilidade à execução. Sugere-se a utilização desse módulo para testar

soluções de projeto, pois pode ser usado em simultâneo a modelagem, espera-se assim incorporar o conhecimento de construtibilidade no processo de projeto.

### **4.3 Desenvolvimento**

#### **4.3.1 Medir**

Suportando o desenvolvimento do artefato têm-se critérios provenientes de três fontes: critérios definidos no protótipo abrangendo construtibilidade, revisão sistemática da literatura sobre métodos de avaliação de projeto e sugestões a partir de consulta a especialistas. Esses formaram um conjunto amplo de formas para avaliar projetos, foram separados por dimensão de análise e classificados de acordo com inputs, complexidade e intenção.

Nas etapas a seguir serão explicados como foram classificados os critérios, propõe-se uma nova forma de classificação em duas dimensões, e então define-se cada categoria dos critérios, está se referindo diretamente ao desempenho da edificação cujo projeto está sob análise do artefato.

##### *4.3.1.1 Proposta de Classificação quanto a Implementação*

Neta seção apresenta-se uma proposta de classificação de critérios quanto a implementação. Essa classificação se faz necessária devido à complexidade inerentes às próprias regras e a amplitude das condições às quais elas precisam se aplicar. Devido ao grande número de regras que podem ser definidas, é essencial sistematizar as regras para tornar a tarefa de verificação de regras tratável.

O artefato desenvolvido trabalha com regras de verificação voltados para o conceito de Orientação, como visto na seção Classificação das regras, proposto por Hjelseth e Nisbet (2010a). Assim, as regras avaliam a solução adotada no projeto em relação as outras possíveis. Porém, não se atinge o conceito ideal, onde o sistema de verificação deveria propor soluções previamente testadas e aprender com o desenvolvimento do projeto.

Então, baseado no conceito de orientar o desenvolvimento do projeto, têm-se diversos critérios compostos por regras a serem verificadas no modelo BIM. Essas regras implementadas em Dynamo tem diferentes níveis de complexidade. Com relação ao seu desenvolvimento computacional identificou-se duas dimensões que impactam essa complexidade. A primeira foi identificada por Solihin e Eastman (2015) que consiste na complexidade dos processos necessários para se chegar ao objetivo da regra, baseando-se nas relações entre os dados e os tratamentos necessários. São 3 níveis propostos que foram incorporadas nessa classificação, estes foram discutidos na seção 2.2.2.

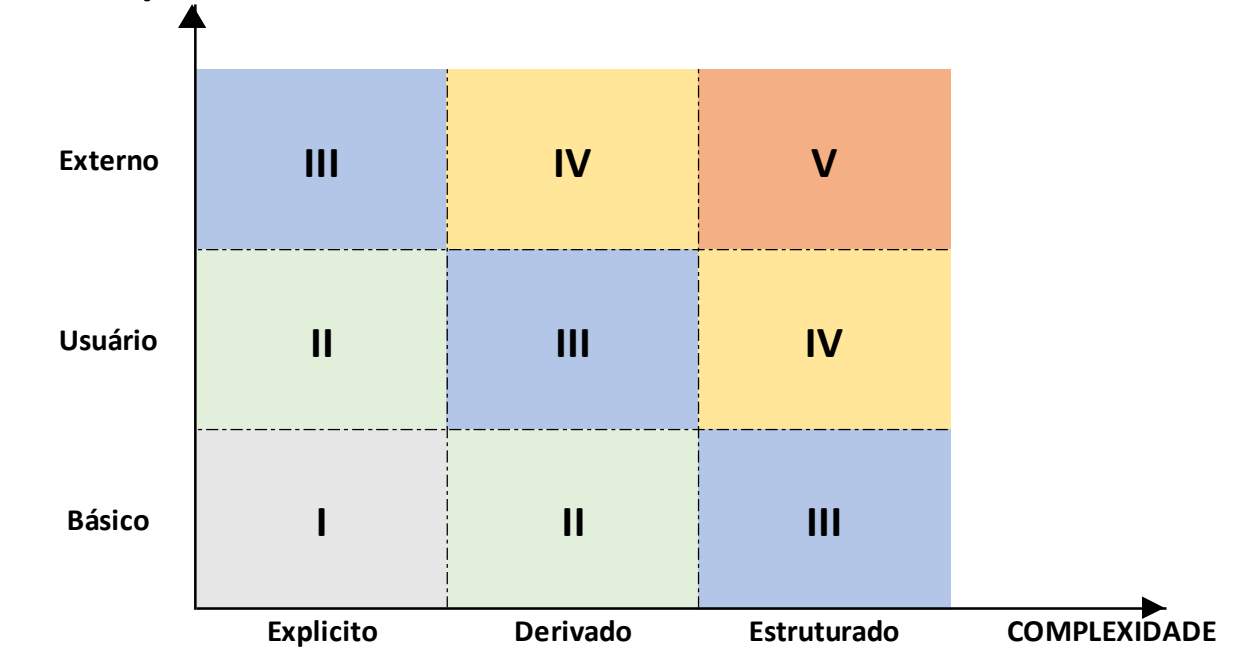


A segunda dimensão que faz parte dessa classificação é a origem dos dados utilizados na implementação. Este conceito se mostrou importante durante a modelagem das regras, pois informações nem sempre estavam disponíveis diretamente no modelo. Se propõe 3 tipos de origem das informações: Básico, informações disponíveis em parâmetros padrão dos modelos analisados; Usuários, informações não padrões que podem ser inseridas através de parâmetros customizados; e Externo, informações complexas que precisam consultar bases de dados externas.

Foi montado o gráfico da Figura 14 que mostra como essas duas dimensões se relacionam, de acordo com a percepção de desenvolvimento do artefato utilizando o Revit e o Dynamo. Optou-se por montar uma escala de cinco níveis:

- Nível I: Para a avaliação utiliza-se dados de parâmetros explícitos nos modelos, ou seja, já previamente calculados e disponíveis. Sendo estas informações padrão para todos os modelos.
- Nível II: Estes critérios aumentam a complexidade em um nível em uma das dimensões individualmente. Os dados precisam ser definidos pelos usuários, ou ainda, o cálculo envolve um dado derivados que precisa ser calculado pelo programa.
- Nível III: No nível 3 sobe-se dois níveis de complexidade individualmente, ou um nível em ambos. São critérios que exigem a consulta de bancos de dados externos; que usam dados fornecidos pelo usuário para um cálculo derivado; ou ainda, que usam dados padrões que necessitam de uma solução estruturado devido à complexidade de cálculo
- Nível IV: Atinge-se o nível máximo em pelo menos uma das dimensões e o segundo nível na outra. São critérios que usam dados externos e cálculos derivados das informações do modelo ou critérios que envolvem cálculos complexos utilizando dados dos usuários e que precisam ser estruturados.
- Nível V: Alanca-se o último estágio em termos de complexidade. São regras de maior dificuldade de serem implementadas, pois utilizam dados externos para executar cálculos estruturados.

Figura 14 Classificação dos critérios informações X complexidade



Fonte: Autor

O critério é classificado previamente baseado no estudo das regras, de modo que se tem uma estimativa quanto ao esforço de desenvolvimento do mesmo. Porém, apenas após sua finalização que este pode ser completamente definido, podendo ainda variar de classificação dependendo da solução empregada na verificação. Neste trabalho os critérios implementados serão classificados na etapa de desenvolvimento.

#### 4.3.1.2 Categorias

Nessa seção introduz-se uma breve descrição do motivo desta categoria estar representadas e como ela é definida neste trabalho. Posteriormente é apresentado o conjunto de critérios identificados dentro da categoria e que serão utilizados nas próximas etapas.

- Construtibilidade
- Sustentabilidade
- Flexibilidade
- Conformidade

A **Construtibilidade** já foi discutida amplamente durante o desenvolvimento do protótipo, tópico 4.2.1. Para esta pesquisa toma-se o conceito de facilidade de construir através de três bases: padronização da construção, acessibilidades dos espaços, simplificação de passos e desconstrução. A Tabela 12 apresenta os critérios avaliados.

Tabela 12 Critérios de Construtibilidade

<b>N</b>	<b>Dimensão</b>	<b>Indicador</b>	<b>Autor</b>	<b>Proposta</b>
C1	Construtibilidade	Índice de Compacidade	Lantelme (1994) Mascaró (2010)	Representa inversamente a complexidade geométrica do perímetro do pavimento. Quanto mais distante de um quadrado (0,84), menor será o índice, e menor a Construtibilidade.
C2	Construtibilidade	Índice de Área Molhada	Oliveira, Lantelme e Formoso (1995) Narloch (2015)	Áreas molhadas exigem maior quantidade de serviços devido a impermeabilização, testes e uso de cerâmicas nas alvenarias
C3	Construtibilidade	Densidade de Paredes	Oliveira, Lantelme e Formoso (1995)	Este índice tem o objetivo de verificar o grau de otimização da compartimentação do pavimento tipo
C4	Construtibilidade	Índice de Fachadas	Oliveira e Freitas (1996) Mascaró (2010) Narloch (2015)	Os planos verticais das fachadas são mais difíceis e custosos de serem construídos. O indicador revela a proporção de fachadas em relação ao pavimento tipo.
C5	Construtibilidade	Densidade de Esquadrias	Oliveira, Lantelme e Formoso (1995)	Esquadrias demandam maior número de serviços e aumentam a complexidade construtiva
C6	Construtibilidade	Índice de Padronização de Esquadrias	Oliveira, Lantelme e Formoso (1995)	A maior diversidade de esquadrias influi na complexidade do projeto, na compra, e no planejamento e operação de execução.
C7	Construtibilidade	Densidade de Pilares	Jarkas (2010) Franco (2014)	Pilares restringem a movimentação na obra e aumentam pontos de fundações.
C8	Construtibilidade	Densidade de Vigas	Jarkas (2010)	Este índice representa a eficiência do projeto. Quanto menor este valor, menores serão as complexidades dos serviços de formas e concretagem, também reduzindo interferências.
C9	Construtibilidade	Índice de Padronização de Pilares	Jarkas (2010)	Este índice considera a complexidade na individualidade de tipos estruturais, através da relação de quantidades de pilares diferentes em suas seções transversais e o número total de pilares.

C10	Construtibilidade	Índice de Padronização de Vigas	Jarkas (2010)	Este índice considera a complexidade na individualidade de tipos estruturais, através da relação de quantidades de vigas diferentes em suas seções transversais e o número total de vigas.
C11	Construtibilidade	Índice de Padronização de Lajes	Jarkas (2010)	Este índice considera a complexidade na individualidade de tipos estruturais, através da relação de quantidades de lajes diferentes em suas seções transversais e o número total de lajes.
C12	Construtibilidade	Índice de Tubulações	Mattos (2015); Paliari (2010)	Relaciona a quantidade de tubulação com a área do pavimento, quanto maior o índice mais complexo
C13	Construtibilidade	Índice de Conduítes	Mattos (2015); Paliari (2010)	Representa o desempenho do projeto elétrico na otimização da distribuição dos conduítes
C14	Construtibilidade	Índice de Desconstrução	AKINADE <i>et al.</i> , (2015)	Determina até que ponto um edifício pode ser desmontado para reutilização ou realocação
C15	Construtibilidade	Índice de Recuperação	AKINADE <i>et al.</i> , (2015)	Representa a facilidade de recuperação e reutilização de materiais após o fim da vida útil do edifício

A **Flexibilidade** das edificações está se tornando cada vez mais importante a medida em que proprietários percebem que a longevidade da edificação pode levar a sua obsolescência. Este conceito na construção de edificações está fortemente ligada a flexibilidade do produto, ou seja, a facilidade com que a empresa consegue adaptar seus produtos a um grupo restrito de clientes ou a um cliente individualmente (BARROS NETO; FENSTERSEIFER; FORMOSO, 2003). A Tabela 13 apresenta os critérios de flexibilidade desenvolvidos no artefato. As adaptações do produto vem por meio de modificações ao longo do ciclo de vida que são de 3 tipos de acordo com Slaughter (2001):

- **Função:** conjunto de atividades realizadas pela edificação ou seus ocupantes, podendo as funções originais serem aprimoradas ou redirecionadas, ou ainda, a edificação pode incorporar novas funções.
- **Capacidade:** alterar a capacidade da edificação para uma alteração na sua performance ou ainda devido ao aumento do volume de atividades. Exigindo readequação do espaço.

- Fluxo: uma alteração nos requisitos de movimentação, passagem ou transporte de pessoas ou objetos durante as diferentes situações de uso da edificação.

Tabela 13 Critérios de Flexibilidade

N	Dimensão	Indicador	Autor	Proposta
F1	Flexibilidade	Modularidade da Estrutura	CAVALLIERE, C. et al. (2018)	Se refere aos módulos repetíveis da estrutura, as distâncias entre dois próximos elementos de suporte de carga. Quanto mais modular, mais flexível.
F2	Flexibilidade	Localização do Shaft	CAVALLIERE, C. et al. (2018)	Localização de serviços técnicos (banheiros, cozinhas, salas técnicas etc.), considerados elementos quase invariantes durante o ciclo de vida do edifício devido à dificuldade de modificar as posições das prumadas
F3	Flexibilidade	Elementos construtivos removíveis	CAVALLIERE, C. et al. (2018)	Considera quatro categorias de elementos de construção removíveis, nomeadamente paredes divisórias, paredes externas, pisos e elementos de suporte de carga (vigas e pilares), avalia a percentagem de elementos de construção removíveis em comparação com o total
F4	Flexibilidade	Razão e orientação de janelas	CAVALLIERE, C. et al. (2018)	Estabelece se o número e a extensão das aberturas são suficientes para ajudar na redistribuição dos espaços internos.
F5	Flexibilidade	Partições internas móveis	CAVALLIERE, C. et al. (2018)	Considera a percentagem de partições móveis em comparação com a área total de paredes divisórias internas

A **Sustentabilidade** é um conceito amplo e com variações nas suas definições, uma das primeiras e mais importantes é “desenvolvimento que atenda às necessidades do presente sem comprometer a capacidade das gerações futuras de atender às suas próprias necessidades” (WCDE, 1987). Quando se trata de edificações os conceitos não apresentam

uma forma fixa, porém de acordo com (BERARDI, 2013), uma edifício sustentável deve promover:

- a demanda por construção segura, flexibilidade, valor econômico e de mercado;
- a neutralização dos impactos ambientais, incluindo seu contexto e sua regeneração;
- o bem-estar humano, satisfação dos ocupantes e partes interessadas;
- a equidade social, melhorias estéticas e preservação dos valores culturais.

E é com base nessas definições que os critérios foram alocados nesta categoria, assumindo a edificação como um consumidor de recursos naturais, bem com um espaço de convivência e geração de riquezas culturais e sociais. A Tabela 13 apresenta os critérios de Sustentabilidade desenvolvidos neste artefato.

Tabela 14 Critérios de Sustentabilidade

<b>N</b>	<b>Dimensão</b>	<b>Indicador</b>	<b>Autor</b>	<b>Proposta</b>
S1	Sustentabilidade	Índice de permeabilidade	Carvalho, Bragança, Mateus (2019); SBTool-H	Razão que representa a parte da área do terreno que está impermeabilizada devido ao empreendimento
S2	Sustentabilidade	Uso de materiais com conteúdo reciclado	Carvalho, Bragança, Mateus (2019); SBTool-H	Participação de materiais reciclados dentro da massa total dos materiais utilizados na edificação
S3	Sustentabilidade	Uso de materiais orgânicos certificados	Carvalho, Bragança, Mateus (2019); SBTool-H	Todos os produtos à base de madeira e orgânicos devem ter sua certificação identificada
S4	Sustentabilidade	Uso de substitutos de cimento no concreto	Carvalho, Bragança, Mateus (2019); SBTool-H	Participação de substitutos de cimento no concreto utilizado no empreendimento
S5	Sustentabilidade	Acessibilidade ao transporte público	Carvalho, Bragança, Mateus (2019); SBTool-H	Avaliar as distâncias de viagem do prédio até as paradas de transporte público próximas
S6	Sustentabilidade	Veículos elétricos	Fator Verde; Especialistas	Defini quantidade de vagas para veículos necessárias
S7	Sustentabilidade	Iluminação	Fator Verde; Especialistas	Projetar ambientes com área mínima de aberturas e adotar princípios básicos de iluminação natural.
S8	Sustentabilidade	Ventilação	Fator Verde; Especialistas	Projetar ambientes com área mínima de aberturas de

				ventilação natural e adotar estratégias de conforto térmico passivo.
--	--	--	--	--

A **Conformidade** está diretamente ligado as regulamentações governamentais, que por sua vez são ativos importantes da sociedade, estendem as leis que governam o país com orientações específicas para ações corporativas e públicas (CHENG; LAU; LAW, 2010). Uma das principais preocupações dos profissionais de qualquer setor é garantir a conformidade de seu trabalho com a multiplicidade de requisitos estatutários, contratuais e de desempenho que regem suas disciplinas (NAWARI, 2012).

A finalidade dos códigos de construção é estabelecer os requisitos mínimos para fornecer um nível razoável de segurança, saúde pública e bem-estar geral através de especificações atribuídas ao ambiente construído. O código de construção se torna lei de uma jurisdição particular quando formalmente promulgada pela autoridade governamental ou privada apropriada (CHING; WINKEL, 2016).

Na indústria de AEC, os projetos de construção devem ser verificados em relação a vários códigos de construção para fins de conformidade. Eles podem ser executados somente quando a conformidade com todas as regras aplicáveis do código de construção tiver sido garantida. A falha em avaliar corretamente os projetos para fins de conformidade também pode ter efeitos negativos no desempenho da construção e permitir erros que são caros de corrigir (MACIT İLAL; GÜNAYDIN, 2017). Os desenvolvidos, Tabela 15, aqui foram apontados por especialistas de Fortaleza-CE baseados em normas que os mesmos precisam atender ao desenvolver seus projetos.

Tabela 15 Critérios de Conformidade

<b>N</b>	<b>Dimensão</b>	<b>Indicador</b>	<b>Autor</b>	<b>Proposta</b>
N1	Conformidade	Índice de aproveitamento	Lei de uso e Ocupação de Fortaleza; Especialistas	Número que multiplicado pela área do terreno, resulta na área de construção computável, estabelecendo as condições de utilização dos instrumentos urbanísticos, jurídicos e tributários
N2	Conformidade	Taxa de ocupação	Lei de uso e Ocupação de Fortaleza; Especialistas	Percentual utilizado pela edificação em relação a área total do terreno, considerando apenas sua projeção horizontal, não sendo computados nesta projeção os elementos

				componentes das fachadas
N3	Conformidade	Recuo edificação	Lei de uso e Ocupação de Fortaleza; Especialistas	É a distância medida entre o limite externo da projeção da edificação no plano horizontal, e a divisa do lote.
N4	Conformidade	Recuo subsolo	Lei de uso e Ocupação de Fortaleza; Especialistas	É a distância medida entre o limite externo da projeção do subsolo da edificação no plano horizontal, e a divisa do lote.
N5	Conformidade	Recuo entre torres	Lei de uso e Ocupação de Fortaleza; Especialistas	É a distância medida entre as projeções das edificações em um mesmo terreno no plano horizontal
N6	Conformidade	Gabarito	Lei de uso e Ocupação de Fortaleza; Especialistas	É o limite máximo, expresso em metros, estabelecido pela legislação urbanística para a altura das edificações de uma determinada área;
N7	Conformidade	Número de vagas	Lei de uso e Ocupação de Fortaleza; Especialistas	Quantidade de vagas de estacionamento mínimas para o tipo de edificação
N8	Conformidade	Tamanho de vagas	Lei de uso e Ocupação de Fortaleza; Especialistas	Largura, comprimento e altura necessários para as vagas de estacionamento
N9	Conformidade	Rota de Fuga/ Saída de emergência	NBR 15.575; NT Nº 005; Especialistas	Identificar a rota de fuga em um momento de emergência, conferir a largura de portas e das escadas de emergência. Atendimento a norma de desempenho
N10	Conformidade	Pé direito	NBR 15.575; Especialistas	Altura livre necessária em ambientes da edificação
N11	Conformidade	Separação entre edificações	NT Nº 001/2008; Especialistas	Identificar a distância entre as edificações vizinhas é maior que a mínima
N12	Conformidade	Rampas	NBR 9050; Especialistas	As características de rampas de acesso destinadas a cadeirantes garantem a funcionalidade destas verificar principalmente inclinação da rampa



N13	Conformidade	Vagas acessíveis	NBR 9050; Especialistas	Reserva de vagas para veículos que transportam pessoa com deficiência com comprometimento de mobilidade; essas vagas devem estar localizadas próximas aos acessos de circulação de pedestres
N14	Conformidade	Dimensão mínima (Portas e Ambientes)	NBR 9050; Especialistas	Verificar dimensões mínimas de aberturas destinadas a ambientes e edificações acessíveis

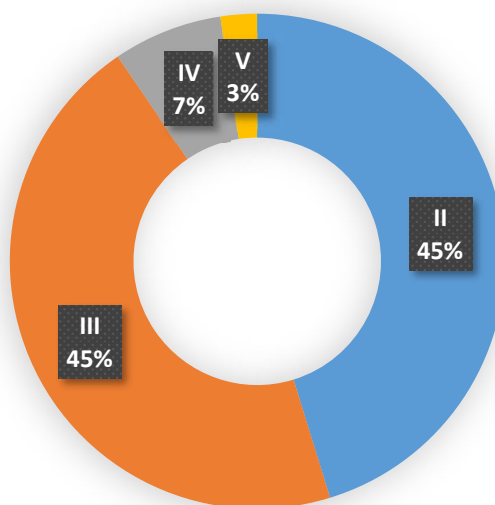
#### 4.3.1.3 Análise dos critérios

Analisando os critérios quanto ao processo e desenvolvimento foi possível classificá-los. A partir do posicionamento nas duas dimensões avaliadas: Informação e Complexidade (Figura 14). Assim, foi possível inferir quais algoritmos demandaram mais esforço para implementação estudando a forma como eles se adequaram na classificação.

Inicialmente observa-se a classificação geral dos critérios analisados, Figura 15. É possível perceber que o número de critérios diminui ao subirmos na classificação, sendo a grande maioria dos critérios classificados em II ou III. Demonstrando que é possível avaliar extensivamente um modelo sem entrar em critérios de implementação complexa, já que IV e V somaram 10%. Outra informação relevante é que nenhum critério foi classificado como I, o que assinala que checagens com regras simples podem não atender ao objetivo de avaliar o desempenho dos projetos.

Figura 15 Classificação Geral dos Critérios

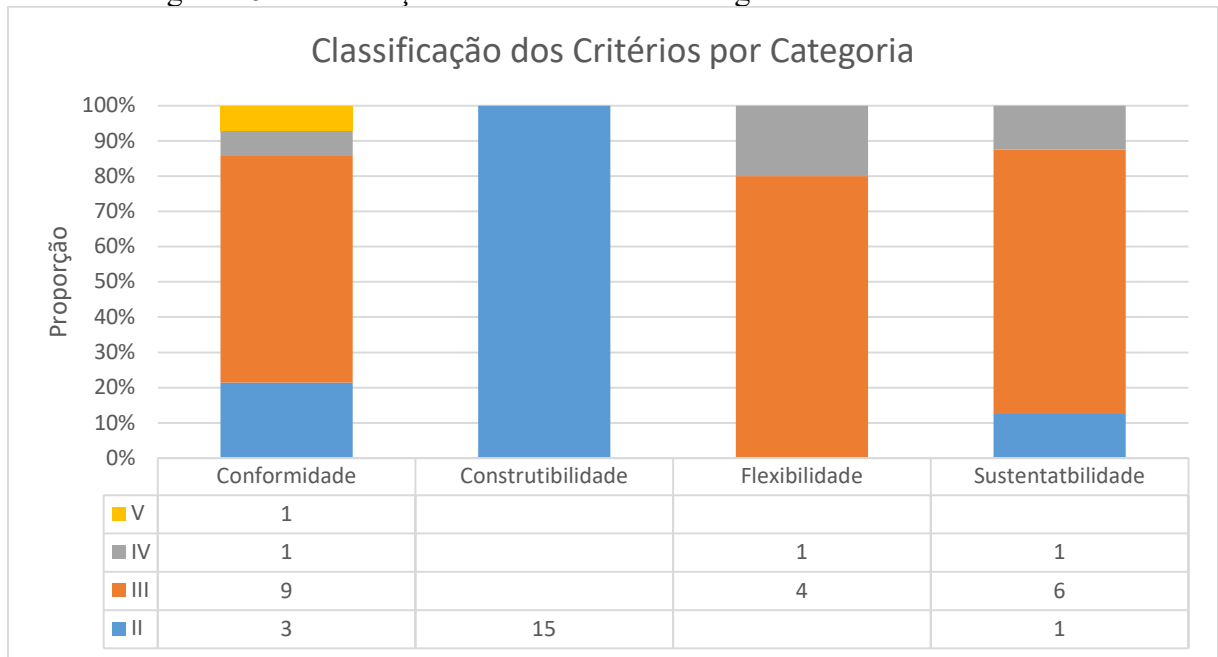
### Critérios por Dimensão



Analisado a Figura 16, mostra distribuição da classificação nas categorias, e Figura 17, analisa a posição em cada dimensão, percebe-se que a classificação dentro das categorias tem uma variação a ser avaliada. Construtibilidade ficou totalmente classificada como “II”, grande parte devido a seus critérios serem puramente geométricos não demandando informações extras dos usuários ou consultas externas, e quando necessita dessas informações a avaliação é de baixa complexidade. A categoria conformidade apresenta uma classificação mais alta, predominantemente “III”, isso se dá aos cálculos mais complexos para adaptar os programas as flexibilizações das normas ou ainda a necessidade de informações extra modelo. De tal forma, que apresenta alguns critérios de classificação “IV”

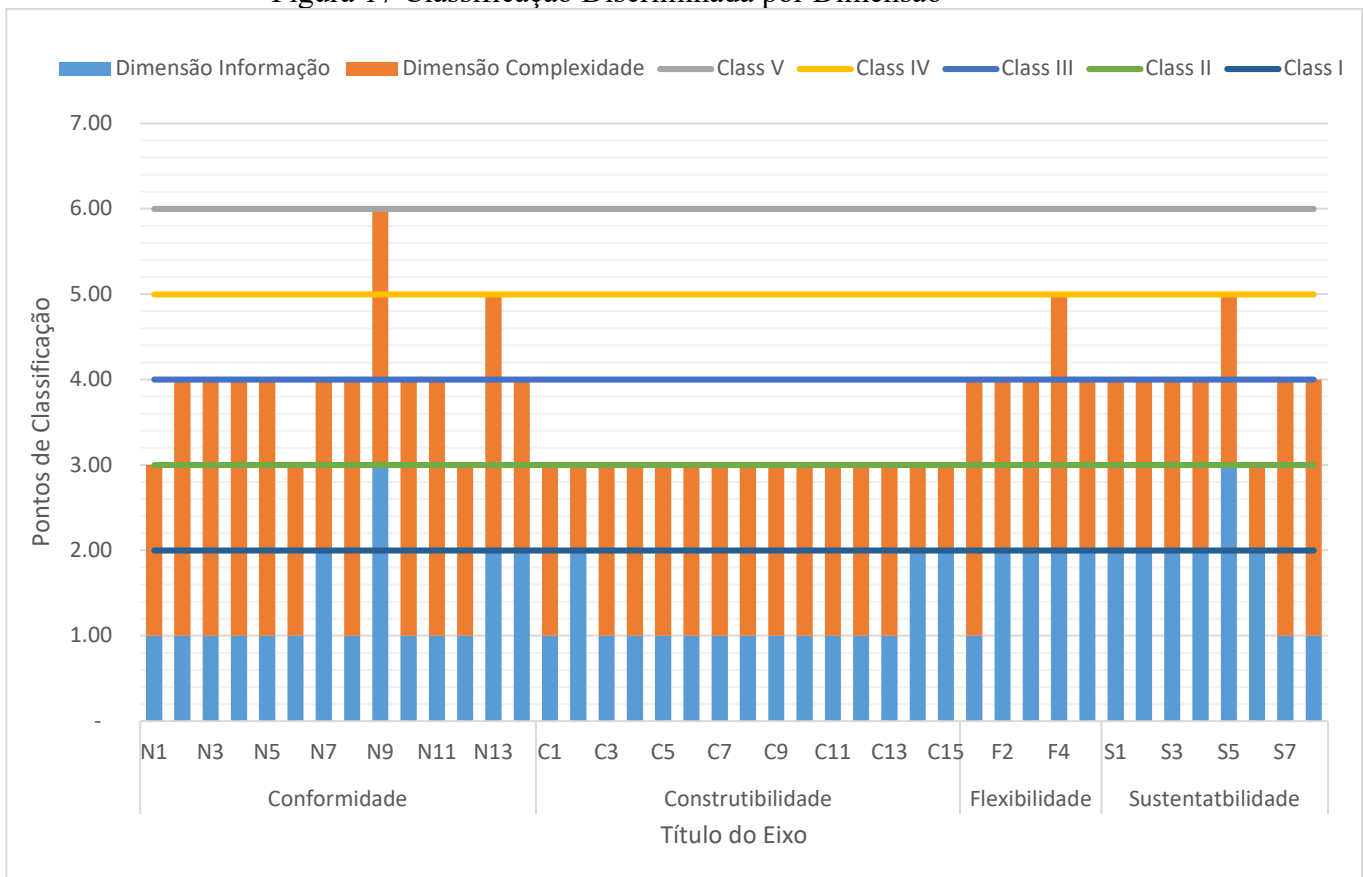
A categoria flexibilidade apresenta os critérios de mais alta classificação apresenta uma certa homogeneidade na classificação “III”, são normalmente critérios geométricos complexos, mas que não demandam consulta de informações. Apenas um dos critérios sobre de nível ao exigir que o usuário identifique um ponto de interesse. Na categoria sustentabilidade têm-se diversos níveis de critérios, na medida em que há verificações simples de contagem de elementos que contenham parâmetros até critérios com alto grau de complexidade de processamento utilizando bancos de dados externos.

Figura 16 Classificação dos Critérios Por Categoria



Fonte: Autor

Figura 17 Classificação Discriminada por Dimensão



Fonte: Autor

### 4.3.2 Implementar

O processo de desenvolver rotinas no Dynamo corresponde em orientar o fluxo de informação através de funções, chamadas de nós. A plataforma dispõe de uma gama de nós padrões que compreendem as funções básicas como: lógica, listas, matemática, geometria e interações com o Revit. No entanto, para funções customizadas se faz necessários desenvolver novos nós ou formular scripts em linguagem Python. Assim, uma vantagem da plataforma é a possibilidade de customização que aliada a forte comunidade de usuários que gera novos nós e os compartilha ativamente.

Este comportamento é distribuído na forma de “Pacotes” que são um conjunto de nós customizados que são disponibilizados na ferramenta, de forma podem ser adicionados a sua biblioteca. Estes adicionam novas funcionalidades e permitem a criação de soluções melhores. Muitos destes nós melhoram funções padrões e permitem gerenciar as informações processadas de forma mais eficiente.

Uma alternativa disponível é a utilização do Grasshopper e sua interação com o Rhinoceros 3D que se conecta a modelos BIM desenvolvidos no ArchiCAD. Este apresenta uma interface de programação visual que funciona de maneira semelhante ao Dynamo. De modo que a maioria dos conceitos desenvolvidos nessa pesquisa podem ser aplicados para avaliações utilizando o Grasshopper.

O processo de desenvolvimento do algoritmo para verificação do critério envolve um ambiente complexo de definições a serem tomadas, de forma a tornar a regra o mais abrangente possível. Os algoritmos possuem seções definidas como:

- Coleta: consiste em realizar uma consulta no modelo BIM para buscar dados para abastecer a análise. Encontrando os elementos que farão parte da análise e coletando seus parâmetros e/ou geometria;
- Consulta: Buscar informações pertinentes a verificação em uma base de dados externa ao modelo BIM, podendo esta ser local ou mesmo consultas a bases online;
- Processamento: os dados são interpretados e sobre eles se aplica a lógica definida no critério, na forma operações e decisões lógicas que orientam o fluxo de informações. Etapa principal e mais importante do algoritmo, é aqui que se aplica inteligência pré-definida e utiliza os recursos disponíveis no Dynamo para obter os resultados, complexidade é variável.

- **Exportação:** Compreender os resultados do processamento e formatar os dados para atender as exigências do critério, exportando para documentação ou visualização;

Alguns, critérios seguem lógicas semelhantes e seu algoritmo pode ser reutilizados com revisões pontuais. De forma que o algoritmo final de cada critério é composto por partes independentes, aqui chamadas de grupos de nós, que são criadas a partir das necessidades e identificadas através de “caixas coloridas” no programa. Estes grupos são o equivalente a definir funções em linguagens de programação tradicionais. Assim, quanto maior e mais diversificado os critérios desenvolvidos, maior é a gama de grupos que poderão ser reutilizadas. Logo, nota-se uma diminuição no esforço de desenvolvimento com o avanço da codificação, devido a construção de um conjunto de ferramentas.

Durante todo o processo de desenvolvimento teve-se como mentalidade tornar as verificações amplas e genéricas de forma a permitir sua utilização em uma maior gama de modelos. Porém, devido ao modo como as regras são definidas, através de um código pré-estabelecido, exige que todas as possibilidades sejam exauridas para evitar problemas futuros. Já que o programa não permite alterações dentro da estrutura que não pelo código, limitando ao usuário que não domina o Dynamo a customizar apenas os inputs.

#### 4.3.2.1 Exemplo de aplicação comentado

Para demonstrar o processo de desenvolvimento selecionou-se o critério Razão e Orientação de Janelas da categoria Flexibilidade para mostrado seu passo a passo e funcionamento. Este foi escolhido por conter uma boa variedade de funções, apresenta uma complexidade por utilizar a geometria do modelo para gerar novas informações para a análise, mostrando um fluxo de informação importante para as avaliações desenvolvidas, além de abordar todas as etapas de desenvolvimento.

O critério em questão busca avaliar se as aberturas externas permitem a redistribuição dos ambientes. Ele é definido como a média da razão entre o comprimento das aberturas e o comprimento das paredes em cada uma das quatro direções (Norte, Oeste, Sul e Leste). Dada por:

$$\sum_j \frac{L_w + L_{cw}}{L_w} \times \frac{1}{4}$$

j = Orientação da construção

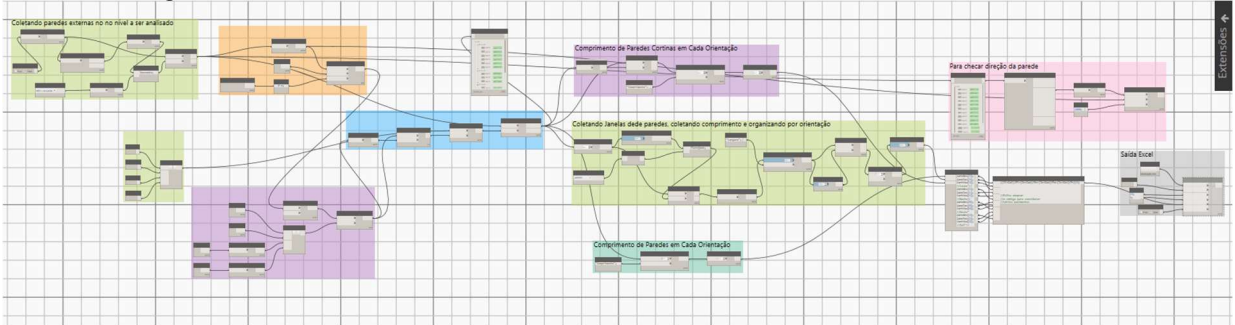
L<sub>w</sub> = Largura total das janelas na direção j

L<sub>cw</sub> = Largura total das paredes cortina na direção j

$LW =$  Comprimento total das paredes na direção  $j$

A implementação deste critério resultou na rotina mostrada na Figura 18, completamente desenvolvido no Dynamo para a partir de um modelo BIM retornar o critério do tipo indicador em análise. Este será analisado a seguir em 4 etapas.

Figura 18 Critério Razão e Orientação de Janelas

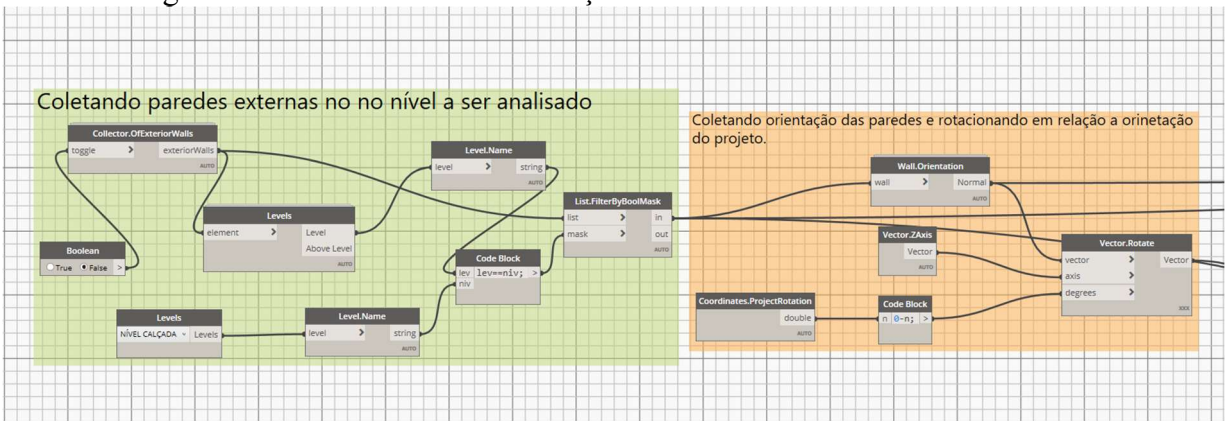


Fonte: Autor

Na primeira parte da rotina, Figura 19, está a etapa de Coleta. Inicia-se coletando todas as paredes externa do modelo. Nota-se que é preciso que o modelo esteja configurado corretamente para que as paredes corretas tenham a função externa. E então, estas são filtradas quanto ao nível em análise. Este grupo (verde) do programa retorna uma lista com todas as paredes, incluindo paredes cortinas, no nível escolhido.

O segundo grupo (laranja), ainda na Figura 19, é a interface entre a etapa de coleta e de processamento. É necessário identificar qual a orientação das paredes em análise. A função `Wall.Orientation` calcula um vetor normal a face externa das paredes, notar que a parede deve ter sido modelada com sua face externa no exterior da edificação. Este vetor é rotacionado caso o projeto apresente alguma rotação em relação ao norte. Este grupo retorna o conjunto dos vetores normais associado as paredes.

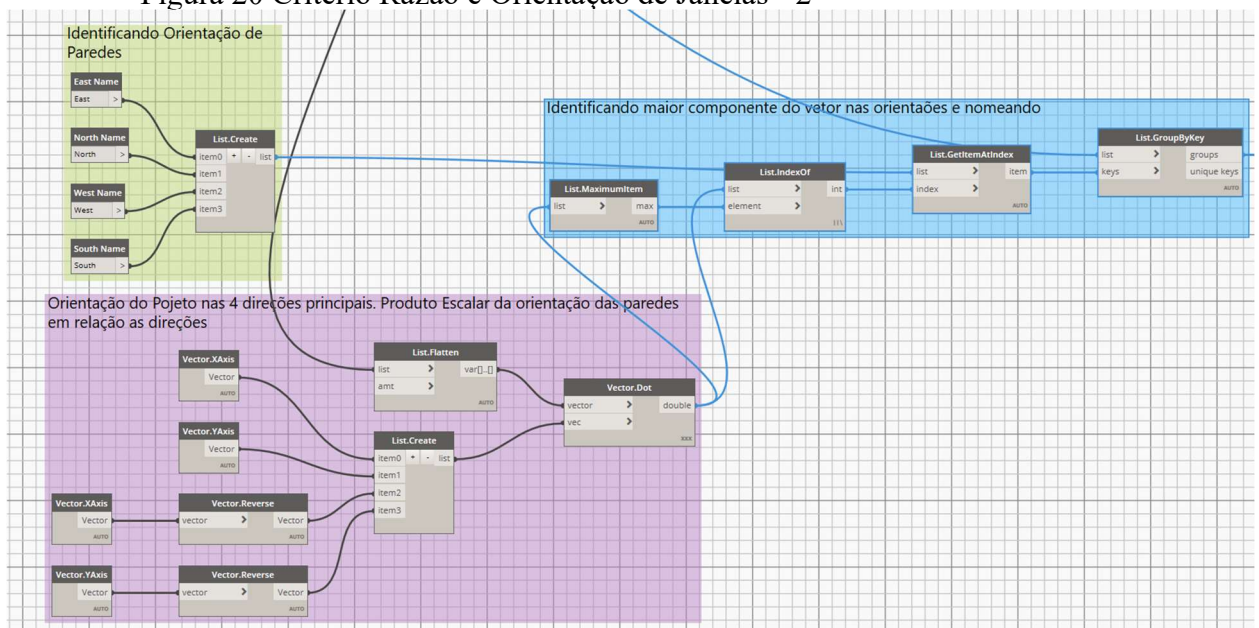
Figura 19 Critério Razão e Orientação de Janelas - 1



Fonte: Autor

A segunda etapa é mostrada na Figura 20. Consiste em um primeiro grupo que cria uma lista com as indicações de orientação norte, sul, leste e oeste (verde). O segundo grupo (roxo) utiliza os vetores X e Y, juntamente com seus reversos para identificar as orientações do projeto no espaço geométrico. Então realiza o produto escalar de cada vetor normal as paredes com as 4 direções principais. Assim, o terceiro grupo (azul) consegue identificar qual a orientação predominante para a parede, já que quanto maior o produto escalar maior a projeção deste vetor na direção principal, de modo que o retorno deste grupo é uma lista com 4 sublistas, agrupando as paredes com a mesma orientação.

Figura 20 Critério Razão e Orientação de Janelas - 2

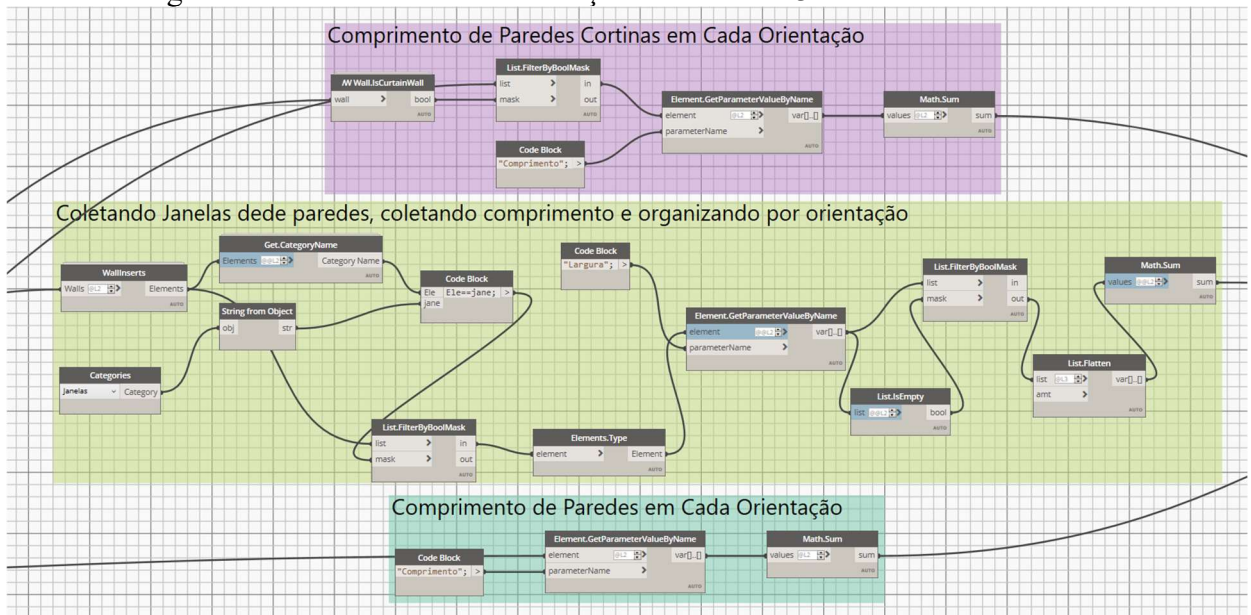


Fonte: Autor

A terceira etapa do programa, Figura 21, processa as informações estruturadas geradas na etapa anterior. O grupo verde escuro calcula o comprimento das paredes, mantendo a estrutura da lista agrupada por orientação, assim, ao somá-las têm-se o comprimento total em cada uma das direções principais. O grupo roxo identifica quais paredes são cortinas, filtra-as e soma seus comprimentos em cada direção.

O grupo verde claro, coleta os elementos inseridos nas paredes e filtra o resultado para apenas elementos da categoria Janelas. Então, coleta o parâmetro largura para todas as janelas e soma o resultado em cada direção. Assim, ao final desta etapa todos os parâmetros necessários para o cálculo do indicador estão disponíveis.

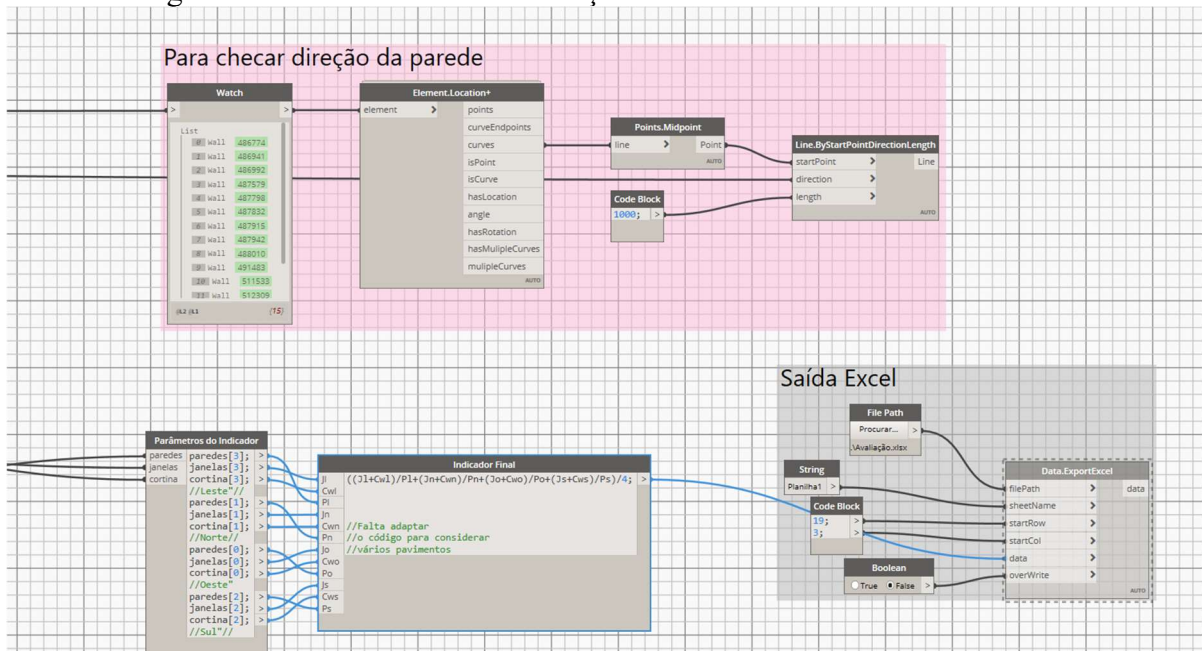
Figura 21 Critério Razão e Orientação de Janelas - 3



Fonte: Autor

A Parte 4, Figura 22, representa a seção de exportação do programa. Primeiro o indicador é devidamente calculado, então no grupo cinza exporta-se o resultado para uma planilha de Excel. O grupo rosa é uma exportação de suporte, este gera uma visualização dos vetores, tanto no Dynamo quanto no Revit, Figura 23. Completa-se a implementação.

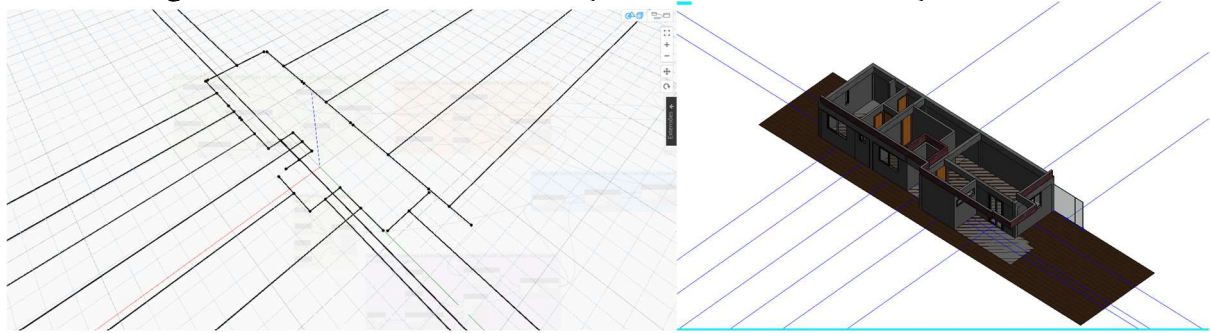
Figura 22 Critério Razão e Orientação de Janelas - 4



Fonte: Autor



Figura 23 Critério Razão e Orientação de Janelas - Visualização



Fonte: Autor

Cada critério desenvolvido teve sua abordagem única para a partir do modelo BIM para gerar resultados que avaliem o projeto dentro da especificidade deste critério. Assim, com o intuito de condensar a informação e permitir entendimento geral do artefato desenvolvido, cada critério foi explicitado em uma tabela resumo, na qual constam informações sobre o que se está sendo medido, qual o resultado esperado, coleta, consulta, processamento e exportação do algoritmo. A Tabela 16 contém explicações de como esta tabela foi utilizada, as cores são um indicativo visual a categoria que o item pertence.

Tabela 16 Modelo de Tabela Resumo

Categoria	N	Autor	Nome	Equação	Parâmetros		
A qual categoria pertence	<b>Codificação</b>	Referencial científico	Identificação	Definição Matemática	<i>Definição das variáveis do critério</i>		
<b>Descrição</b>	Descreve o critério quanto a seu propósito e indicação de uso						
<b>Etapa</b>	<b>Entrada</b>			<b>Processo</b>	<b>Saída</b>		
	<i>Tipo</i>	<i>Fonte</i>	<i>Entrada</i>	<i>Etapas</i>	<i>Resultado</i>	<i>Und</i>	
Algoritmo	Como o input é definido	Forma de acesso à informação	Identificação Da entrada	Passo a passo do algoritmo do critério	mostra a etapa final do cálculo	Indicativo da unidade do indicador	
<i>Classificação</i>	Podendo ser categorias do modelo	Podendo ser informações		Para cada etapa é explicitado como as variáveis são coletadas.			
Posição na classificação proposta	Informações fornecidas pelo usuário ou bancos de dados para consulta	Padrão do Revit, informada pelo Usuários ou consulta extra modelo BIM		Como é processada a Informação Como ela é exportada			

Fonte: Autor

A seguir têm-se a tabela resumo do critério do exemplo anterior, Tabela 17. Esta condensa toda a informação dos critérios. No Apêndice está disponível a tabela de todos os critérios, relatando as informações de todos os critérios avaliados.

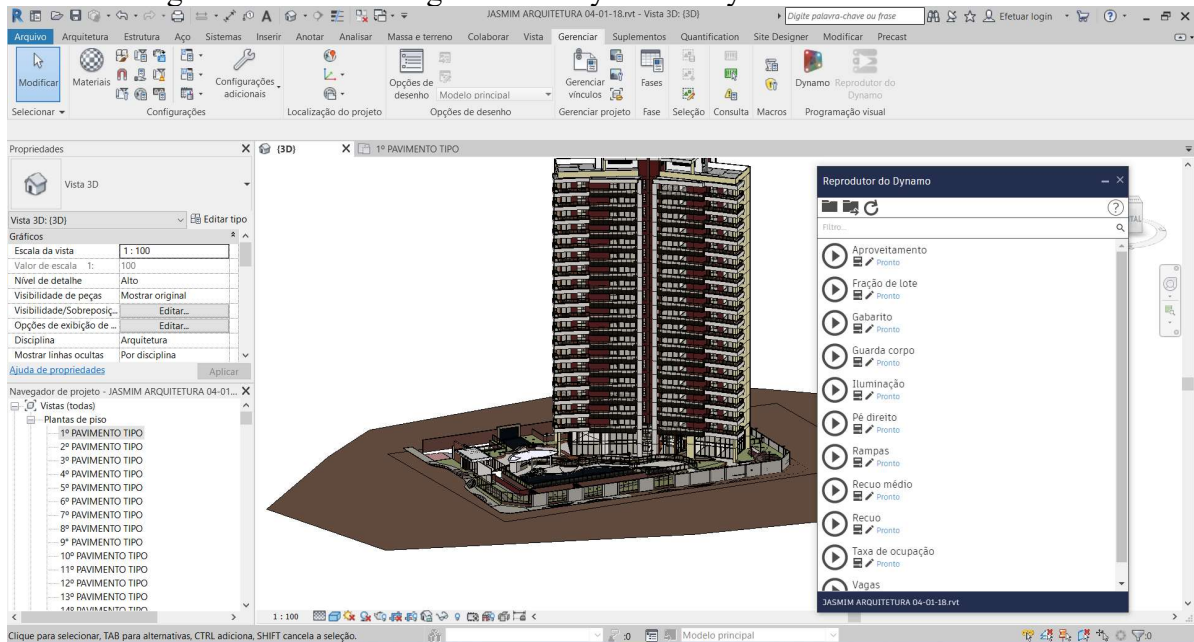
Tabela 17 Resumo Implementação - Orientação de Aberturas

<b>Categoria</b>	<b>N</b>	<b>Autor</b>	<b>Nome</b>	<b>Equação</b>	<b>Parâmetros</b>	
Flexibilidade	F5	(CAVALLIERE <i>et al.</i> , 2019)	Porcentagem e orientação das janelas	$\sum_j \frac{L_w + L_{cw}}{L_w} \times \frac{1}{4}$	<i>j</i> = Orientação da construção <i>L<sub>w</sub></i> = Largura total das janelas <i>L<sub>cw</sub></i> = Largura total das paredes cortina <i>LW</i> = Largura total das paredes	
<b>Descrição</b>	Estabelecer um indicador para a relação entre o número e a extensão das aberturas em cada orientação, apontando sua suficiência em ajudar na redistribuição dos espaços internos.					
<b>Etapa</b>	<b>Entrada</b>			<b>Processo</b>	<b>Saída</b>	
	<i>Tipo</i>	<i>Fonte</i>	<i>Entrada</i>	<i>Etapas</i>	<i>Resultado</i>	<i>Und</i>
Algoritmo	Categoria	Elementos do Modelo	Paredes	Calcular orientação das paredes externas (J)	Calcular indicador, aplicando parâmetros na equação	%
	Categoria	Elementos do Modelo	Janelas	Coletar parâmetro largura das janelas (L <sub>w</sub> ) por orientação		
Classificação	Categoria	Elementos do Modelo	Paredes Cortina	Coletar parâmetro largura das paredes cortina (L <sub>cw</sub> ) por orientação		
	III	Categoria	Elementos do Modelo	Função das Paredes		

Fonte: Autor

Um ponto a ser analisado é a integração dos critérios desenvolvidos com a interface do software de modelagem. O Dynamo nativamente apresenta a interface Dynamo Player que permite a criação de inputs e outputs para o usuário do Revit sem a necessidade de abrir o programa. Esta funcionalidade é útil para aumentar a adoção de usuários a ferramentas, pois facilita sua utilização com uma interface simples e intuitiva. No Revit se aponta a localização dos programas no computador, e o Dynamo Player apresenta-os em formato de lista (Figura 24). Há a opção de executar diretamente, editar o programa ou entrar na tela de inputs.

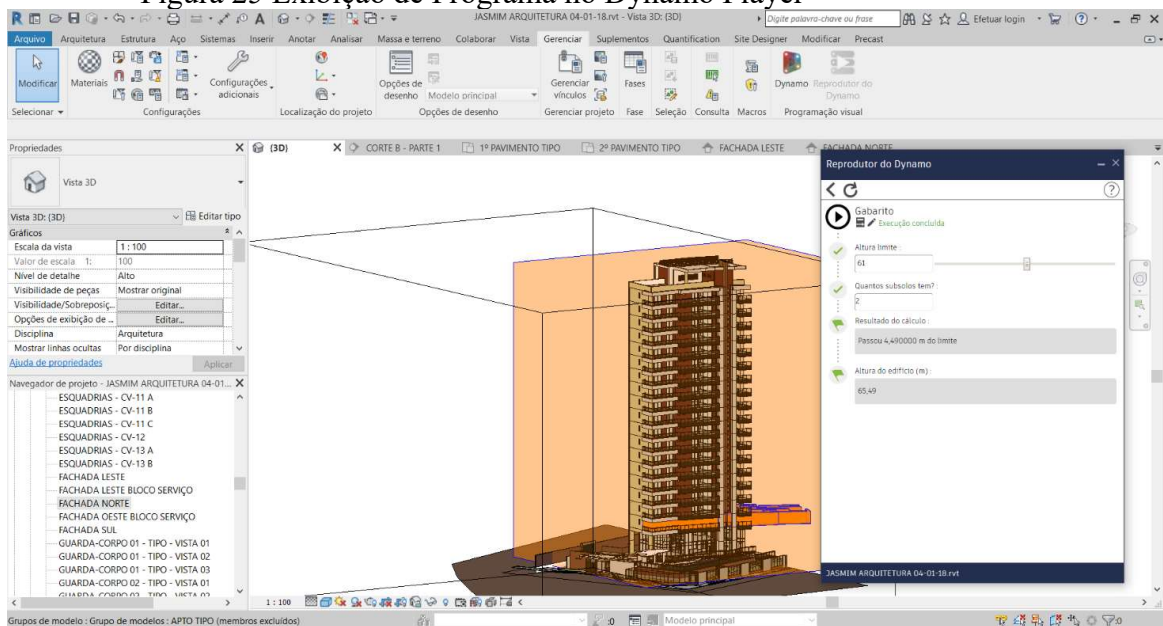
Figura 24 Lista de Programas no Dynamo Player



Fonte: Autor

No desenvolvimento dos programas existem funções específicas que podem ser configuradas como “entradas” ou “saídas”. Existe uma limitação destas funções, porém estão disponíveis entradas com caracteres, valores, booleanos e seleções de categorias ou tipo no Revit. Já como saída, o padrão é o nó “Watch” que exibe o que é passado para ele na interface. Ao entrar no programa através do Dynamo Player este exibe as entradas configuradas e ao ser executado retorna as saídas. A Figura 25 mostra para o programa de verificação de altura da edificação ou gabarito.

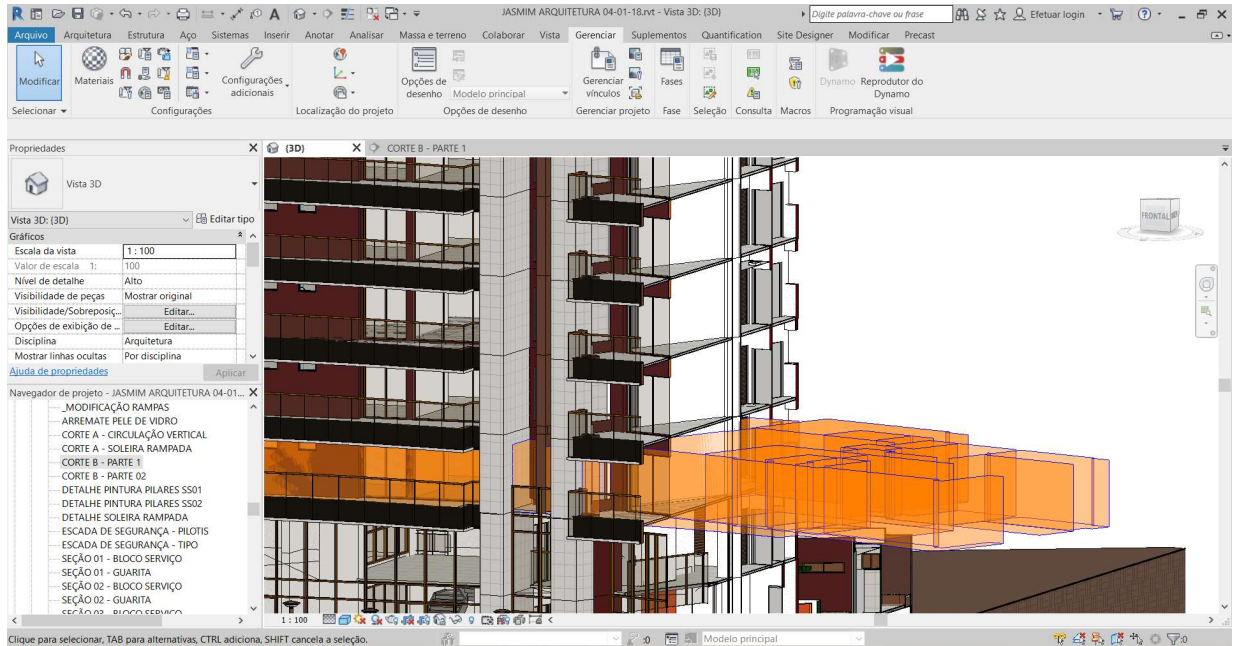
Figura 25 Exibição de Programa no Dynamo Player



Fonte: Autor

Outra opção para apresentar os resultados da verificação é utilizar a geometria gerada no ambiente do Dynamo para mostrar pré-visualizações de resultados no Revit. A Figura 26 mostra a geometria gerada a partir dos ambientes para verificar altura livre dos ambientes. Torna evidente para os usuários o cálculo sendo executado na verificação, facilitando o entendimento e aumentando a confiança no programa

Figura 26 Exemplo de visualização de geometria Dynamo no Revit - Pé Direito

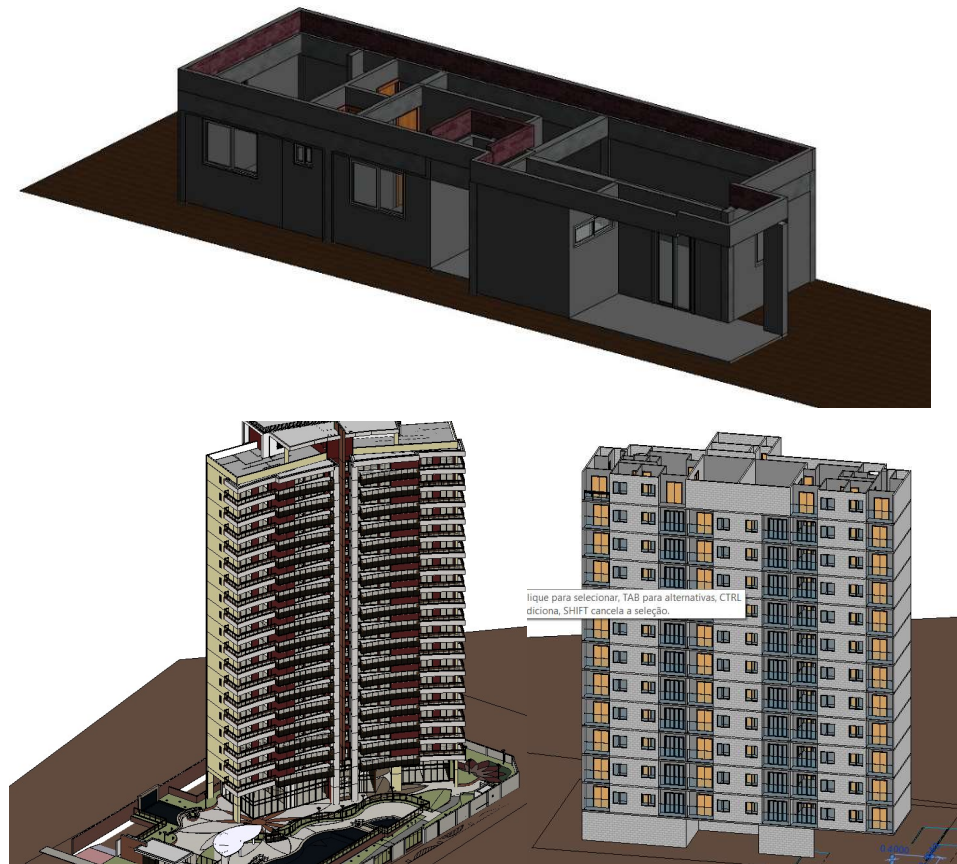


Fonte: Autor

#### 4.4 Avaliação

Iniciou-se o processo de avaliar o artefato através de uma simulação de uso com modelos em que os resultados de cada critérios fora previamente calculado de forma manual através das ferramentas de medição do Revit. Foram utilizados três modelos, em níveis de detalhamento diferente, estes foram cedidos por uma empresa parceira. A Figura 27 mostra os modelos utilizados, por serem diferentes facilitaram o estudo da variação da modelagem em algumas regras.

Figura 27 Modelos Utilizados nos Testes



Fonte: Autor

Faz-se assim um comparativo direto entre o resultado do artefato e o calculado manualmente para estudar as divergências. Em caso de diferenças relevantes o critério retorna a etapa de desenvolvimento é ajustado para um novo teste.

Tabela 18 Síntese dos resultados de avaliação

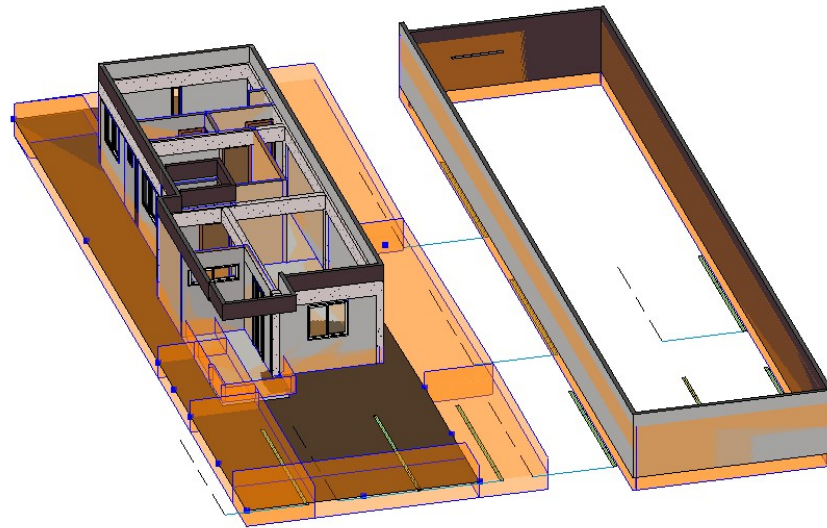
N	Critério	Artefato	Medida	Varição	UND	OBS
C1	Índice de Compacidade	0,73	0,73	-0,03%	-	
C2	Índice de Área Molhada	19,0%	18,98%	-0,01%	%	
C3	Densidade de Paredes	9,63%	9,63%	0,00%	%	
C4	Índice de Fachadas	149,9%	149,88%	0,00%	%	
C5	Densidade de Esquadrias	13,06%	13,06%	0,03%	%	
C6	Índice de Padronização de Esquadrias	75,00%	75,00%	0,00%	%	
C7	Densidade de Pilares	5,11%	5,11%	0,00%	%	
C8	Densidade de Vigas	59,46%	59,46%	0,00%	%	
C9	Índice de Padronização de Pilares	66,7%	66,67%	0,00%	%	
C10	Índice de Padronização de Vigas	71,4%	71,43%	0,00%	%	
C11	Índice de Padronização de Lajes	50,0%	50,00%	0,00%	%	
C12	Índice de Tubulações	0,108	0,108	0,00%	-	
C13	Índice de Conduítes	0,30	0,30	-0,05%	-	
C14	Desconstrução	0,20	0,20	0,00%	%	

C15	Recuperação	0,15	0,15	0,00%	%	
F1	Modularidade da Estrutura	0,58823	0,588	0,04%	-	
F2	Localização do Shaft	80,40%	80,32%	0,10%	%	
F3	Elementos construtivos removíveis	0,79%	0,79%	0,00%	%	
F4	Razão e orientação de janelas	19%	19%	-0,27%	%	
F5	Partições internas móveis	55,2%	55,0%	0,27%	%	
S1	Índice de permeabilidade	58,98%	59,76%	-1,31%	%	
S2	Uso de materiais com conteúdo reciclado	0,09	0,09	0,00%	%	
S3	Uso de materiais orgânicos certificados	0,04	0,04	0,00%	%	
S4	Uso de substitutos de cimento no concreto	0,07	0,07	0,00%	%	
S5	Acessibilidade ao transporte público	103,89	104	-0,11%	%	
S6	Veículos elétricos	2	2	0,00%	und	
S7	Iluminação	14,62%	14,58%	0,27%	%	Média dos ambientes
S8	Ventilação	13,62%	13,80%	-1,30%	%	Média dos ambientes
N1	Índice de aproveitamento	41,1%	42,0%	-2,26%	-	Paredes Externas modeladas no terreno
N2	Taxa de ocupação	45,46%	45,42%	0,09%	%	
N4	Recuo edificação	-	-	-	m	Realiza verificação de acordo com requisitos
N5	Recuo subsolo	-	-	-	m	Realiza verificação de acordo com requisitos
N6	Recuo entre torres	-	-	-	m	Realiza verificação de acordo com requisitos
N7	Gabarito	3,2	3,2	0,00%	m	
N8	Número de vagas	3	3	0,00%	und	
N9	Tamanho de vagas	11,52	11,52	0,00%	m <sup>2</sup>	
N10	Rota de Fuga/Saída de Emergência	14,105	14,4	-2,05%	%	
N12	Pé direito	2,5	2,5	0,00%	%	
N15	Separação entre edificações	4,8	4,8	0,00%	%	
N16	Rampas	20%	20%	0,00%	%	Inclinação
N19	Vagas acessíveis	6	6	0,00%	und	
N20	Dimensão mínima portas	1	1	0,00%	und	

Fonte: Autor

A tabela mostra uma alta precisão do artefato desenvolvido, variação média de -0,17% entre o artefato e a medição manual. Comprovando a eficiência dos algoritmos para os modelos testados. A verificação dos recuos foi um caso especial, já que devido à grande variação do que afeta esta regra, ela foi desenvolvida para acusar ou não a violação dos recuos mínimos. A Figura 28 mostra a geometria utilizada na verificação, o programa responde o status de cada recuo individualmente.

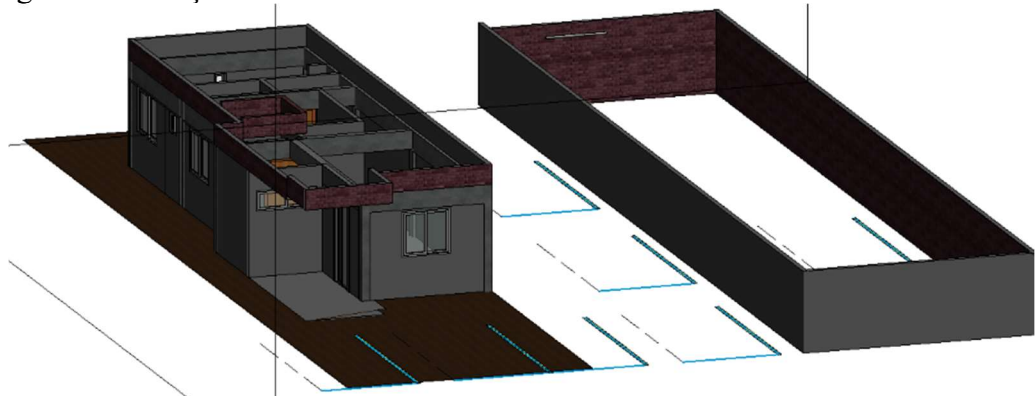
Figura 28 Verificação dos Recuos



Fonte: Autor

Havia, ainda, a necessidade de garantir abrangência das regras desenvolvidas. Utilizou-se de variações nos modelos como forma de estressar o algoritmo em situações mais complexas. A Figura 29 mostra uma dessas variações, onde se estudou como paredes fora da edificação afetariam as regras, em algumas delas foram precisos adicionar condicionais específicos para este caso.

Figura 29 Variação do Modelo



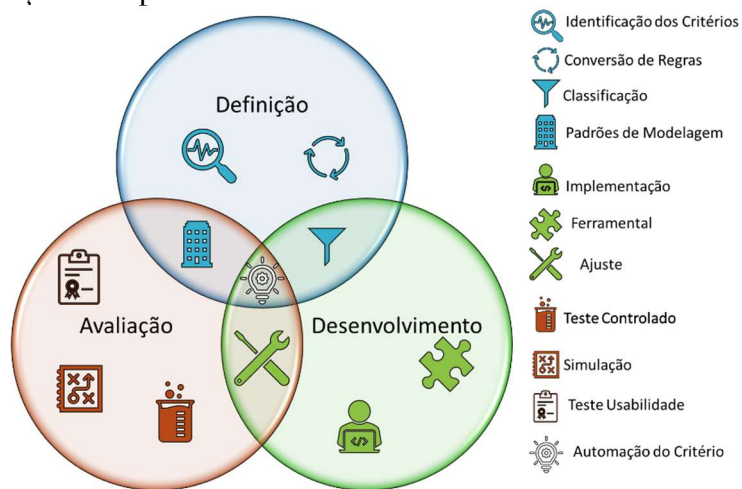
Fonte: Autor

É importante apontar que no processo de validação foi possível não apenas apontar problemas no artefato para correção, mas também variações na modelagem que afetavam os resultados. Podendo então redefinir os padrões de modelagem para o critério avaliado. Logo o processo de avaliação sedimenta as definições estabelecidas no início do processo e controla o avanço do desenvolvimento. Sendo, então, imprescindível para o desenvolvimento do artefato.

### 4.5 Proposição de Modelo de Desenvolvimento de Critério de Avaliação de Projeto Automatizados

O desenvolvimento do artefato gerou evidências e experiências que foram as bases de um modelo para automatização de critérios ou indicadores de projeto baseados em um modelo BIM e apoiado por programação visual. Esta sessão define as etapas a serem seguidas e expõe as ferramentas recomendadas. A Figura 30 ilustra as etapas do modelo proposto e a interação entre etapas

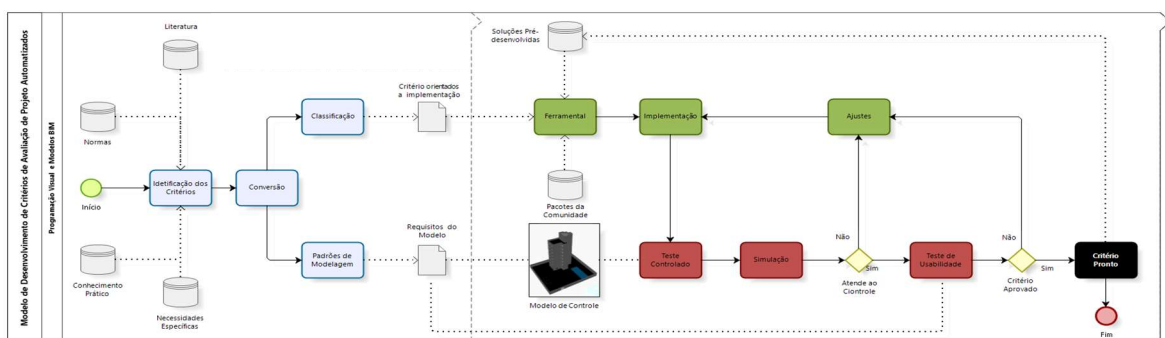
Figura 30 Interação e Etapas do Modelo de Desenvolvimento



Fonte: Autor

Propõe-se 3 etapas para a automatização de um critério de desempenho: Definição, Desenvolvimento e Avaliação. As etapas têm objetivos específicos a serem alcançados para garantir que o processo obtenha sucesso. Estas, embora específicas, apresentam uma interface de transferência de informação importante entre elas. A Figura 30 explicita estas etapas principais e as atividades necessárias. É possível ver também a interação entre as etapas, já que a atividades que se encaixam em mais de uma, estas usam subprodutos das atividades de modo a integrar o processo.

Figura 31 Modelo de Desenvolvimento de Critério de Avaliação de Projeto Automatizados





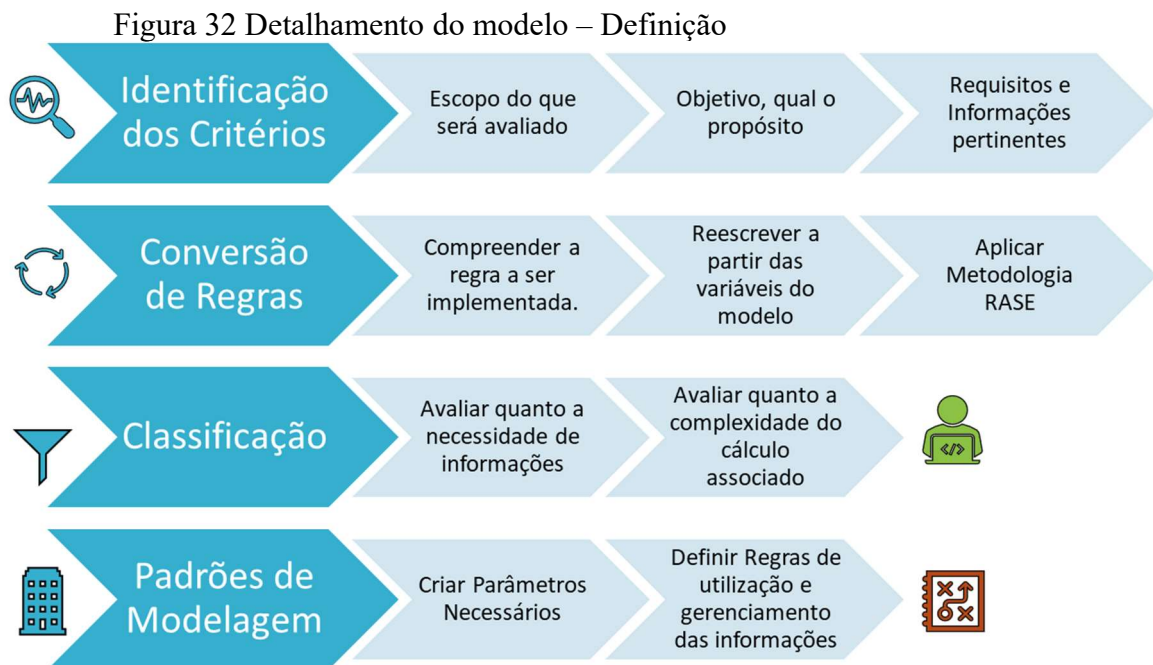
Definição   Desenvolvimento   Avaliação

Fonte: Autor

Nas próximas seções entra-se em detalhes nas atividades propostas, mostrando ferramentas, insumos e produtos destas. Enfatiza-se a especificidade do desenvolvimento utilizando a programação visual e as atividades necessárias para obter o sucesso.

#### 4.5.1 Definição

Esta primeira etapa é de preparação, é a decisão do que se pretende avaliar. É uma etapa crítica na medida em que estabelece os objetivos que vão guiar todo o desenvolvimento, definindo tanto a complexidade do cálculo como a utilidade do critério. A Figura 32 sintetiza esta etapa, referenciando as interfaces com as outras etapas do modelo.



Fonte: Autor

O escopo de todo o desenvolvimento inicia com a definição de quais regras serão automatizadas. É importante basear essa escolha em critérios consolidados pela literatura ou pelo uso prático dos usuários. Recomenda-se estabelecer estas definições em conjunto com os usuários finais, pois o entendimento das expectativas do dono do produto é de suma importância para o processo.

Após decidir o critério a ser estudado e especificar o escopo de avaliação, é preciso se voltar para o modelo BIM para mapear as informações que serão utilizadas. Ao estudar o cálculo necessário se define as variáveis de dentro do modelo. Estas podem ser

parâmetros em elementos, geometria de elementos ou ainda relação entre eles. Caso a informação não exista no modelo, ela pode ser consultada em bancos de dados externos.

A definição das variáveis é um dos processos para a Conversão de Regras explicada na seção 2.2.3 da revisão deste trabalho. Esta possui um conjunto de ferramentas que permitem converter as regras escritas em uma lógica apropriada para automatização. Assim, identifica-se como se realizará o desenvolvimento do critério.

Pode-se, então, classificar o critério para entender o esforço que será necessário na próxima etapa. Utilizando a regra convertida pode-se entender a complexidade dos cálculos definindo uma dimensão da classificação. Ao mesmo tempo que observando as variáveis necessárias entende-se como será a fonte de informações do critério. Obtendo, assim, informações importantes para classificá-lo. A classificação nessa fase é preliminar e tem uma interface com o processo de desenvolvimento, pois apenas quando o mesmo estrutura o algoritmo pode se entender completamente a regra.

Assim, com as regras devidamente definidas, é preciso aplicar essas definições a modelagem. Um critério pode necessitar de um parâmetro complementar, por exemplo, o critério de desconstrução exigia que na modelagem fosse apontado os elementos desconstruíveis. A melhor abordagem para esse tipo de critério é alterar na biblioteca de famílias, criando os parâmetros e o preenchendo corretamente, previamente a sua utilização. Entre outros casos que exigem um padrão de modelagem claro a ser seguido pela equipe de projetos para alimentar a automatização

#### **4.5.2 *Desenvolvimento***

Esta etapa compreende o esforço de implementação necessário para tornar o critério automatizado, resumido na Figura 33. Segmentou-se em: Implementação, Ajuste e Ferramental. Esta etapa traduz o esforço de desenvolvimento, sabe-se que este engloba uma grande variedade de possibilidades, devido à complexidade inerente do processo de criação de códigos e algoritmos.

Figura 33 Detalhamento do modelo - Desenvolvimento



Fonte: Autor

Inicia-se pela atividade de Ferramental que dá suporte a implementação, esta compreende formas de organizar funções a serem utilizadas para agilizar e potencializar o desenvolvedor. Assim, buscar as soluções já implementadas na comunidade abre um leque de novas ferramentas, as ferramentas de programação visual facilitam essa troca de informações como um ambiente de código aberto através do compartilhamento de pacotes. Estes contêm um conjunto de “nós” desenvolvidos por membros da comunidade.

A segunda forma de suportar a implementação é através do estudo das soluções desenvolvidas nas regras anteriores. Pois, ao finalizar o desenvolvimento deve-se organizar os nós em grupos que correspondem a funções específicas do programa, então identificá-los por sua utilidade e orientá-los no espaço de desenvolvimento de uma forma lógica, adicionando notas de explicação quando necessário. Estes grupos identificados serão registrados no conjunto de soluções disponíveis, formando o ferramental de desenvolvimento. Este processo de limpeza e organização, muitas vezes deixado de lado, é importante para facilitar futuras alterações no programa e garantir que o esforço despendido no desenvolvimento poderá ser utilizado para agilizar os próximos projetos.

A etapa seguinte corresponde a parte criativa de codificação do processo, deve-se utilizar das capacidades máximas da ferramenta de programação visual para otimizar o funcionamento. Há uma interface com a atividade de conversão de regras, pois o ela fornece o insumo para o desenvolvimento que é o critério estruturado para implementação, há ainda interface com a atividade de classificação que prever o esforço a ser despendido.

Inicia-se por coletar as informações do modelo, é interessante explorar o modelo a partir da coleta de Categorias específicas e então filtrar os elementos a partir de condicionais

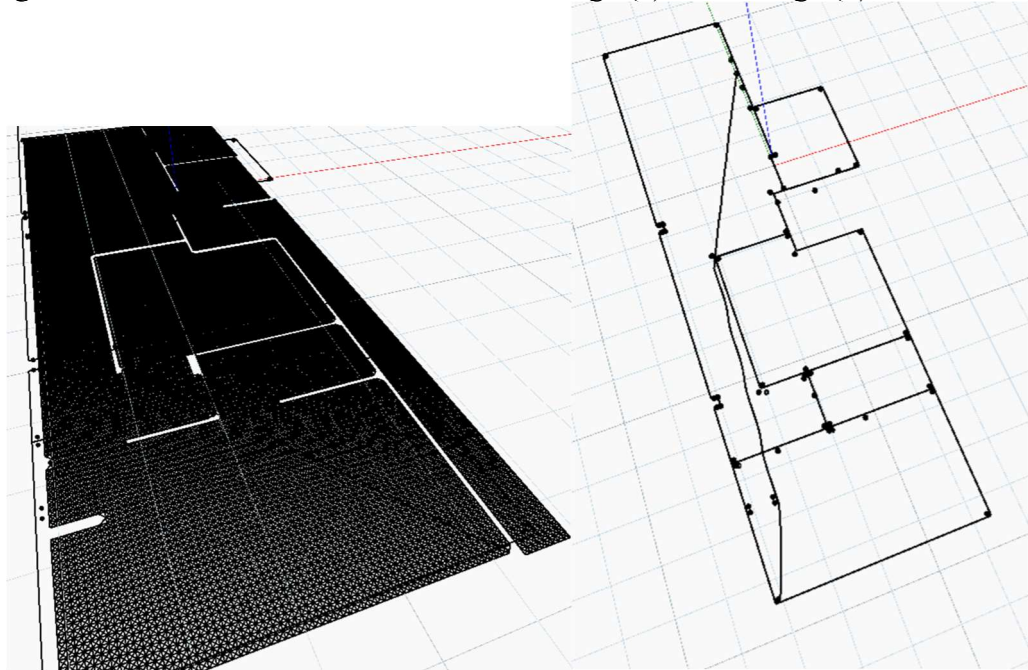
para montar a base de informações necessárias. Estas condicionais podem ser parâmetros, tipos, localização ou característica específica definida na modelagem. Em regras mais complexas é preciso conectar bancos de dados ao algoritmo para a utilização destas informações.

Então deve-se utilizar os dados obtidos para gerar as informações utilizadas no critério. Assim, cada input terá um fluxo específico para alimentar o critério, cálculos simples atendem alguns e são suficientes para atingir os objetivos, por exemplo: utilizar uma função de contagem para quantificar elementos específicos; aplicar operações matemáticas em parâmetros extraídos do modelo, verificar proporção entre quantidade de elementos com um parâmetro de marcação; entre outras. Porém, em vários casos é preciso gerar novas informações com as baseado no que foi coletado.

Informações advindas do modelo são utilizadas para criar dados, normalmente envolvendo propriedades geométricas. Alguns exemplos, são a distância entre objetos; Áreas específicas; Translação ou rotação da geometria; Verificação de intersecção; Operações booleanas entre geometrias; entres outras aplicações. Estes dados derivados exigem maior conhecimento das ferramentas geométricas. As ferramentas de programação visuais aplicadas a modelos BIM possuem ambientes geométricos definido por pontos, curvas, superfícies e sólidos, estas geometrias possuem propriedades e operações próprias que são úteis na avaliação do modelo, já que estas ferramentas têm a capacidade de extrair a geometria de elementos no modelo BIM para geometria do ambiente.

Um exemplo é o critério de Rota de Fuga que para sua análise exigia geração de geometria complexa e posterior análise para seleção do maior caminho. A Figura 34 (a) mostra a geometria necessária para a análise deste critério, são milhares de pontos e conexões que são avaliados para encontrar o caminho mais longo, Figura 34 (b). Esse tipo de critério exige desenvolver estruturas de dados para guardar essas informações para as diversas necessidade, atingindo o grau mais alto de complexidade na escala de classificação apresentada neste trabalho.

Figura 34 Geometria do Critério de Rota de Fuga (a) rota de fuga (b)



Fonte: Autor

Cálculos são realizados para atingirem o resultado proposto pelo critério, estes precisam ser formatados e organizados de forma a fornecer a informação necessária para a tomada de decisão do usuário. Isso envolve trabalhar a lógica das listas de resultados para o formato específico escolhido, pois ao gerenciar a estrutura das listas pode-se mostrar os resultados resumidos, assim como exibir resultados intermediários. Destacam-se aqui 3 opções de exportação dos resultados dos critérios:

1. Mostrar o resultado parcial e final diretamente no software de modelagem, útil para verificações rápidas e correções no curso do projeto;
2. Exportar os resultados salvando um arquivo de texto, normalmente em formato .csv ou .json que podem ser lidos em outros sistemas;
3. Exportar os resultados em uma planilha através da integração com softwares específicos, destaca-se aqui a possibilidade de criar um template personalizado que pode agrupar os resultados de diversas regras.

Pode-se utilizar mais de um tipo de forma de exportação dos resultados adequando as necessidades da tarefa. Há ainda, diversas outras possibilidades através da integração de bibliotecas do Python, como por exemplo envio por e-mail. O importante é atender ao formato especificado na Definição do critério. Sugere-se aqui a combinação do método 2 ou 3 com uma interface de *Business Intelligence* para análises mais avançadas.

A atividade de ajustes segue todos os processos do desenvolvimento, pois cada pedaço de código adicionado precisa ser validado e, por vezes, corrigido. Essa atividade é fortemente dependente dos resultados de ambas as atividades da Avaliação. Deve-se ajustar o programa para atender completamente a avaliação. Assim, esta etapa é composta por diversas iterações até que ambas as avaliações estejam satisfeitas.

#### 4.5.3 Avaliação

Esta etapa checa se o programa atende ao que foi previamente definido, inspecionando os resultados da implementação. Identificando desvios do objetivo e apontando pontos de correção. Imprescindível para garantir a qualidade do que se está sendo desenvolvido.

Figura 35 Detalhamento do modelo - Avaliação



Fonte: Autor

Sugere-se um modelo BIM controlado para testes, este corresponde a um modelo em que os resultados do critério são conhecidos. De modo que, o resultado conhecido se comporta como uma meta para o código atender, servindo para identificar possíveis erros no programa. Assim, o processo iterativo de desenvolvimento, teste e ajuste surge e continua até o programa atingir a meta estabelecida. Em cada teste realizado, volta-se para o programa para entender a causa do desvio e realizar o ajuste.

De modo a avançar nos ajustes é preciso variar o modelo controlado, realizando simulações através da colocação de erros propositais ou aumentando a complexidade do modelo para entender como o programa se comporta nesse ambiente. Novamente, desvios da meta (agora alterada) devem ter as causas investigadas e corrigidas. É neste momento que o programa se torna mais genérico, pois a cada simulação com erro corrigido o código se torna mais abrangente, e assim, mais eficiente para verificação de modelos de fontes diferentes.

Então, com o critério atendendo ao modelo controlado e comportando variações no tipo de modelo, leva-se o produto desenvolvido ao usuário final. Apresenta-se a ele o funcionamento do programa como um todo, detalhando entradas, processo e saída. Realiza-se uma utilização assistida do critério automatizado. Por fim, questiona-se o usuário a respeito do produto, avaliando se as expectativas foram atendidas e se há sugestões de melhoria. Ao final confirma-se os padrões de modelagem, já que as alterações realizadas no Ajuste podem exigir uma abordagem nova na construção do modelo. Os padrões são revisados e, se necessário, readequados.

## 5 CONCLUSÕES

Este trabalho desenvolveu uma ferramenta de checagem de modelos capaz de avaliar características do projeto que afetam seu desempenho. Visa prover ao projetista a capacidade de avaliar seu produto em diversas características e avaliar o desempenho em categorias distintas, em menos tempo e de modo mais assertivo. Assim, o artefato proposto aqui pode aprimorar o processo de desenvolvimento dos projetos e pode resultar em melhores produtos. Efeito esperado pela possibilidade de realizar verificações de conformidade e desempenho de forma automatizada.

A pesquisa se orienta a partir de uma Design Science e apresenta etapas bem definidas. Há um fluxo de informação iterativo entre as etapas, atividades a serem seguidas e ferramentas para superar dificuldades. Foi mostrada a capacidade da programação visual (Dynamo) como interface entre projetistas e a *Application Programming Interface* do software de modelagem (Revit) de ser utilizada de forma diversa e eficaz. Embora, aponte-se que em problemas mais complexos conhecimentos de programação são importantes para o sucesso do projeto e que os algoritmos desenvolvidos continuam seguindo regras rígidas e previamente definidas.

Assim, durante desenvolvimento da pesquisa foi proposto uma ferramenta de classificação de critérios quanto ao desenvolvimento, baseando-se na complexidade da regra e a necessidade de informações do modelo. Já os critérios de verificação desenvolvidos foram agrupados em categorias para agrupar critérios de verificação: Construtibilidade, Flexibilidade, Sustentabilidade e Conformidade.

Ao final da construção do artefato foi gerado um modelo robusto para fomentar a verificação automatizada de modelos. Foi proposto um modelo para desenvolvimento de critérios automatizados para verificação de projetos constituído de 3 etapas com interligações fundamentais e um conjunto de atividades a serem realizadas. A Definição identifica o escopo, formaliza o critério e o classifica. O Desenvolvimento promove a construção do algoritmo com base em conhecimento prévio e ajustes constantes baseados no resultado esperado. A Avaliação controla o que foi desenvolvido por meio de testes controlados e simulações, para então apresentá-lo ao usuário.

Algumas limitações foram identificadas no decorrer da pesquisa. Destaca-se o fato de as rotinas desenvolvidas serem regidas pelas definições iniciais, podendo não suportar variações não previstas. O modelo foi testado com uma ferramenta de programação visual (Dynamo), podendo ter de sofrer algumas adaptações para correta eficiência em outras



ferramentas. A última atividade da avaliação, teste de usabilidade, não foi realizada no desenvolvimento do artefato proposto nesta pesquisa, pois não havia cliente final que tenha estabelecido expectativas e necessidades para o artefato.

Uma dificuldade que ficou clara durante a pesquisa é a necessidade de utilizar-se de padrões de modelagem para garantir conformidade nos dados do modelo. A possível utilização de uma biblioteca padronizada para o Brasil, com um sistema de classificação de objetos poderia ser utilizado para vencer esta barreira. Já que a interpretação dos modelos por algumas regras exige a presença de parâmetros específicos.

Este trabalho apresenta uma série de desdobramentos que podem ser abordados em trabalhos futuros. Destaca-se a capacidade de utilizar o modelo para validar critérios de avaliação de projetos, uma vez que ao serem automatizados pode-se agilizar o processo de geração de dados para análise de projetos variados. Outra possível aplicação é criação de um indicador geral para projetos ou categoria de projetos, utilizando ferramentas multicritérios de avaliação de desempenhos.

Há ainda a possibilidade de utilizar os critérios automatizados como pontos de contornos para processos de design generativos. Definindo o resultado do critério como uma restrição no modelo a ser gerado ou como indicador de escolha do design gerado. Podendo ser utilizado para dar suporte a geração de um conjunto de opções de projetos otimizadas a partir do conjunto de critérios desenvolvidos seguindo o modelo proposto neste trabalho.

Outro possível trabalho futuro seria avaliar se o modelo de desenvolvimento proposto funcionaria com modelos federados IFC e plataformas de checagem open source. Utilizando-se de ferramentas abertas disponíveis para desenvolver os critérios. Já que esse tipo de ambiente permite uma maior cooperação entre empresas e plataformas de modelagem diferentes,

Os objetivos propostos foram atendidos em plenitude. “Identificar dimensões de desempenho de uma edificação para serem avaliadas por meio do modelo de informação”, foram identificadas 4 dimensões para serem avaliadas nesta pesquisa. “Selecionar critérios relativos às dimensões especificadas a serem avaliadas”, foram selecionados 42 critérios, estes foram categorizados e classificados. “Desenvolver algoritmo para a avaliação dos critérios através programação visual”, todos os critérios foram automatizados, sendo sua avaliação realizada através de ferramentas de programação visual. “Propor modelo para desenvolvimento de algoritmos de avaliação de projeto”, foi apresentado um modelo que pode ser seguido para a partir de uma necessidade chegar-se a um critério automatizado.

Considera-se que este artefato mostra que as ferramentas de programação visual, em especial o Dynamo, podem ser utilizadas para desenvolver sistemas de verificação de critérios baseados em modelos BIM. Apresentando uma solução satisfatórias para diversos propostos.

## REFERÊNCIAS

- AKINADE, O. O. *et al.* Waste minimisation through deconstruction: A BIM based Deconstructability Assessment Score (BIM-DAS). **Resources, Conservation and Recycling**, [s. l.], v. 105, p. 167–176, 2015. Available at: <https://doi.org/10.1016/j.resconrec.2015.10.018>
- ALARCÓN, L. F.; MARDONES, D. A. Improving the design-construction interface. *In: , 1998. Proceedings of the 6th Annual Meeting of the International Group for Lean Construction*. [S. l.: s. n.], 1998. p. 1–12.
- ANDRADE, E. De; FERREIRA, M. Estimativas De Custos Com Base Em Bim E Algoritmos. **Simpósio brasileiro de gestão e economia da construção - SIBRAGEC**, Fortaleza, n. November, 2017.
- ANDRADE, M. L.; RUSCHEL, R. C. BIM: Conceitos, Cenário das Pesquisas Publicadas no Brasil e Tendências. **Simpósio Brasileiro de Qualidade do Projeto no Ambiente Construído**, [s. l.], p. 602–613, 2009. Available at: <https://doi.org/10.4237/sbqp.09.166>
- ARAYICI, Y. *et al.* Technology adoption in the BIM implementation for lean architectural practice. **Automation in Construction**, [s. l.], v. 20, n. 2, p. 189–195, 2011. Available at: <https://doi.org/10.1016/j.autcon.2010.09.016>
- ASL, M. R. *et al.* BIM-based Parametric Building Energy Performance MultiObjective Optimization. **32nd eCAADe Conference**, [s. l.], v. 224, p. 10, 2014. Available at: <http://autodeskresearch.com/pdf/bimparametric.pdf>
- ASSAF, S. A.; AL-HEJJI, S. Causes of delay in large construction projects. **International Journal of Project Management**, [s. l.], v. 24, n. 4, p. 349–357, 2006. Available at: <https://doi.org/10.1016/j.ijproman.2005.11.010>
- AZEVEDO, O. **Metodologia BIM - Building Information Modeling na Direcção Técnica de Obras**. 114 f. 2009. - Universidade do Minho, [s. l.], 2009.
- AZHAR, S. Building Information Modeling (BIM): Trends, Benefits, Risks, and Challenges for the AEC Industry. **Leadership Manage. Eng.**, [s. l.], v. 11, p. 241–252, 2011.
- BARBOSA, P.; ANDERY, P. Contribuição ao estudo de medidas para melhoria da construtibilidade no processo de projeto em empresas construtoras. **Construindo**, [s. l.], v. 7, n. 1, 2015.
- BARROS NETO, J. de P.; FENSTERSEIFER, J. E.; FORMOSO, C. T. Os critérios competitivos da produção: um estudo exploratório na construção de edificações. **Revista de Administração Contemporânea**, [s. l.], v. 7, n. 1, p. 67–85, 2003. Available at: <https://doi.org/10.1590/s1415-65552003000100004>
- BENTLEY. **Generative Components Design and explore the unimaginable**. [S. l.], 2018.
- BERARDI, U. Clarifying the new interpretations of the concept of sustainable building. **Sustainable Cities and Society**, [s. l.], v. 8, n. 2013, p. 72–78, 2013. Available at:

<https://doi.org/10.1016/j.scs.2013.01.008>

BUILDINGSMART ALLIANCE. **National BIM Standard - United States - Version 3**. [S. l.: s. n.], 2015. Available at: [https://doi.org/10.1016/S0040-4039\(01\)82124-9](https://doi.org/10.1016/S0040-4039(01)82124-9).

CÂNDIDO, L. F.; LIMA, S. H. de O.; BARROS NETO, J. de P. Análise de sistemas de medição de desempenho na indústria da construção. **Ambiente Construído**, [s. l.], v. 16, n. 2, p. 189–208, 2016.

CAVALLIERE, C. *et al.* BIM-based assessment metrics for the functional flexibility of building designs. **Automation in Construction**, [s. l.], v. 107, n. November 2018, p. 102925, 2019. Available at: <https://doi.org/10.1016/j.autcon.2019.102925>

CHAN, C. T. W. Barriers of Implementing BIM in Construction Industry from the Designers' Perspective: A Hong Kong Experience. **(Online) Journal of System and Management Sciences**, [s. l.], v. 4, n. 2, p. 1818–0523, 2014.

CHEN, L.; LUO, H. A BIM-based construction quality management model and its applications. **Automation in Construction**, [s. l.], v. 46, p. 64–73, 2014. Available at: <https://doi.org/10.1016/j.autcon.2014.05.009>

CHENG, C. P.; LAU, G. T.; LAW, K. H. Mapping regulations to industry-specific taxonomies. [s. l.], p. 59, 2010. Available at: <https://doi.org/10.1145/1276318.1276329>

CHING, F. D. K.; WINKEL, S. R. **Building Codes Illustrated: A Guide to Understanding the 2015 International Building Code**. [S. l.]: Wiley, 2016. (Building Codes Illustrated).*E-book*.

CII. **CII best practices guide: improving project performance**. Estados Unidos, 2012. Available at: [http://www.sig.org/docs2/CII\\_Best\\_Practices\\_Guide\\_Improving\\_Project\\_Performance.pdf?docID=9415](http://www.sig.org/docs2/CII_Best_Practices_Guide_Improving_Project_Performance.pdf?docID=9415).

CIRIA. **Buildability: An Assessment Special publication no. 26**. London: [s. n.], 1983.

COELHO, S. S.; NOVAES, C. C. Modelagem de Informações para Construção (BIM) e ambientes colaborativos para gestão de projetos na construção civil. **WORKSHOP BRASILEIRO - GESTÃO DO PROCESSO DE PROJETOS NA CONSTRUÇÃO DE EDIFÍCIOS**, [s. l.], 2008.

COSTA, J. M. da *et al.* Sistemas de indicadores de desempenho e produtividade para a construção civil. **Lnec Qic2006**, [s. l.], p. 1–12, 2006.

DIMYADI, J.; SOLIHIN, W.; HJELSETH, E. Classification of BIM-based Model checking concepts. **Journal of Information Technology in Construction**, [s. l.], v. 21, n. July, p. 354–370, 2016.

DING, L. *et al.* **Automating code checking for building designs-DesignCheck Automating code checking for building designs-DesignCheck Lan Ding INFORMATION AND**

**COMMUNICATION TECHNOLOGIES IMPROVING EFFICIENCIES** Refereed Paper **AUTOMATING CODE CHECKING FOR BUILDING DESIGNS-Des.** [S. l.: s. n.], 2006. Available at: <http://ro.uow.edu.au/engpapers/4842>.

DRESCH, A; LACERDA, D. P.; ANTUNES, J. A. V. **Design Science Research: A Method for Science and Technology Advancement.** [S. l.]: Springer International Publishing, 2014. *E-book*.

DRESCH, Aline; LACERDA, D. P.; JÚNIOR, J. A. V. A. **Design science research: método de pesquisa para avanço da ciência e tecnologia.** [S. l.]: Bookman Editora, 2015.

DRESCH, Aline; LACERDA, D. P.; MIGUEL, P. A. C. Uma análise distintiva entre o estudo de caso, a pesquisa-ação e a design science research. **Revista Brasileira de Gestao de Negocios**, [s. l.], v. 17, n. 56, p. 1116–1133, 2015. Available at: <https://doi.org/10.7819/rbgn.v17i56.2069>

DYNAMO. **Dynamo BIM: Open source graphical programming for design.** [S. l.: s. n.], 2018.

EASTMAN, C. *et al.* Automatic rule-based checking of building designs. **Automation in Construction**, [s. l.], v. 18, n. 8, p. 1011–1033, 2009. Available at: <https://doi.org/10.1016/j.autcon.2009.07.002>

EASTMAN, C. M. *et al.* Exchange Model and Exchange Object Concepts for Implementation of National BIM Standards. **Journal of Computing in Civil Engineering**, [s. l.], v. 24, n. 1, p. 25–34, 2010. Available at: [https://doi.org/10.1061/\(asce\)0887-3801\(2010\)24:1\(25\)](https://doi.org/10.1061/(asce)0887-3801(2010)24:1(25))

EASTMAN, Chuck *et al.* **BIM Handbook: A Guide to Building Information Modeling for Owners, Managers, Designers, Engineers and Contractors.** [S. l.: s. n.], 2011. ISSN 1837-9133.v. 2 Available at: <https://doi.org/10.1002/9780470261309>

EL-OMARI, S.; MOSELHI, O. Integrating automated data acquisition technologies for progress reporting of construction projects. **Automation in Construction**, [s. l.], v. 20, n. 6, p. 699–705, 2011. Available at: <https://doi.org/10.1016/j.autcon.2010.12.001>

EMMERSON, H. Survey of Problems Before the Construction Industry. *In*: MOODLEY, K.; PREECE, C. (org.). **Construction Reports 1944–98.** [S. l.]: Blackwell Science, 2008.

FABRICIO, M. M.; MELHADO, S. B. Desafios para integração do processo de projeto na construção de edifícios. *In*: , 2001. **WORKSHOP NACIONAL: gestão do processo de projeto na construção de edifícios.** [S. l.: s. n.], 2001.

FORMOSO, C. T. *et al.* **As perdas na construção civil: conceitos, classificações e seu papel na melhoria do setor** -Relator: Núcleo Orientado para a Inovação da Edificação (NORIE), 1996. p. 12.

FRANCO-SANTOS, M. *et al.* Towards a Definition of a Business Performance Measurement System. **International Journal of Operations & Production Management**, [s. l.], 2007. Available at: <https://doi.org/10.1108/01443570710763778>

GHAFFARIANHOSEINI, A. *et al.* Building Information Modelling (BIM) uptake: Clear benefits, understanding its implementation, risks and challenges. **Renewable and Sustainable Energy Reviews**, [s. l.], v. 75, n. October 2015, p. 1046–1053, 2017. Available at: <https://doi.org/10.1016/j.rser.2016.11.083>

GHANNAD, P. *et al.* Automated BIM data validation integrating open-standard schema with visual programming language. **Advanced Engineering Informatics**, [s. l.], v. 40, n. March, p. 14–28, 2019. Available at: <https://doi.org/10.1016/j.aei.2019.01.006>

GRASSHOPPER. **Grasshopper algorithmic modeling for Rhino**. [S. l.: s. n.], 2018.

GRIFFITH, A.; SIDWELL, A. C. Development of constructability concepts, principles and practices. **Engineering, Construction and Architectural Management**, [s. l.], v. 4, n. 4, p. 295–310, 1997.

HEIGERMOSER, D. *et al.* BIM-based Last Planner System tool for improving construction project management. **Automation in Construction**, [s. l.], v. 104, n. January, p. 246–254, 2019. Available at: <https://doi.org/10.1016/j.autcon.2019.03.019>

HEVNER, A. R. *et al.* Design Science in Information Systems Research. **Mis Quarterly**, [s. l.], v. 32, n. 1, p. 725–730, 2004.

HJELSETH, E. BIM-based model checking (BMC). *In: BUILDING INFORMATION MODELING: APPLICATIONS AND PRACTICES*. [S. l.: s. n.], 2015a. p. 33–61. Available at: <https://doi.org/10.1061/9780784413982.ch02>

HJELSETH, E. Converting performance based regulations into computable rules in BIM based model checking software. **eWork and eBusiness in Architecture, Engineering and Construction**, [s. l.], n. August 2012, p. 461–469, 2012a. Available at: <https://doi.org/10.1201/b12516-73>

HJELSETH, E. Experiences on converting interpretative regulations into computable rules. *In:* , 2012b. **CIB W78 2012:**. [S. l.: s. n.], 2012.

HJELSETH, E. **Foundations for BIM-based model checking systems: Transforming regulations into computable rules in BIM-based model checking systems**. 1–198 f. 2015b. - Norwegian University of Life Sciences, [s. l.], 2015. Available at: [http://ibim.no/Eilif\\_Hjelseth\\_PhD\\_Foundations\\_for\\_BIM-based\\_model\\_checking\\_systems\\_2015-11-06.pdf](http://ibim.no/Eilif_Hjelseth_PhD_Foundations_for_BIM-based_model_checking_systems_2015-11-06.pdf)

HJELSETH, E.; NISBET, N. Exploring Semantic Based Model. *In:* , 2010a, Cairo, Egypt. **CIB W78 2010: 27th International Conference**. Cairo, Egypt: [s. n.], 2010.

HJELSETH, E.; NISBET, N. Overview of concepts for model checking. **Proceedings of the CIB W78 2010: 27th International Conference –Cairo, Egypt**, [s. l.], n. December 2002, p. 16–18, 2010b.

HON, S. L.; GAIRNS, D. A.; WILSON, O. D. Buildability: A Review of Research and Practice. **Australian Institute of Building**, [s. l.], v. 3, 1988.

HUMPPI, H. **ALGORITHM-AIDED BUILDING INFORMATION MODELING Connecting Algorithm-Aided Design and Object-Oriented Design**. 176 f. 2015. - Tampere University of Technology, [s. l.], 2015.

JARKAS, A. M. Buildability factors affecting formwork labour productivity of building floors. **Canadian Journal of Civil Engineering**, [s. l.], v. 37, n. 10, p. 1383–1394, 2010. Available at: <https://doi.org/10.1139/L10-063>

JI, Y.; LEITE, F. Automated tower crane planning: leveraging 4-dimensional BIM and rule-based checking. **Automation in Construction**, [s. l.], v. 93, n. May, p. 78–90, 2018. Available at: <https://doi.org/10.1016/j.autcon.2018.05.003>

KASIM, T. *et al.* Automated Sustainability Compliance Checking Process: Proof of Concept. **Proceedings of the 13th International Conference on Construction Applications of Virtual Reality (ConVR)**, [s. l.], n. October, p. 11–21, 2013.

KENSEK, K. Integration of Environmental Sensors with BIM: Case studies using Arduino, Dynamo, and the Revit API. **Informes de la Construcción**, [s. l.], v. Vol. 66, 5, p. 31 – 39., 2014.

KHAN, N. *et al.* Excavation Safety Modeling Approach Using BIM and VPL. **Advances in Civil Engineering**, [s. l.], v. 2019, p. 1–15, 2019. Available at: <https://doi.org/10.1155/2019/1515808>

KHEMLANI, L. The IFC Building Model : A Look Under the Hood. **AECbytes Magazine**, [s. l.], 2004.

KIM, H. *et al.* Visual language approach to representing KBimCode-based Korea building code sentences for automated rule checking. **Journal of Computational Design and Engineering**, [s. l.], v. 6, n. 2, p. 143–148, 2018. Available at: <https://doi.org/10.1016/j.jcde.2018.08.002>

KO, C. H. Lean Building Design Model. **Procedia Engineering**, [s. l.], v. 182, p. 329–334, 2017. Available at: <https://doi.org/10.1016/j.proeng.2017.03.104>

KOO, B. *et al.* Using support vector machines to classify building elements for checking the semantic integrity of building information models. **Automation in Construction**, [s. l.], v. 98, n. October 2018, p. 183–194, 2019. Available at: <https://doi.org/10.1016/j.autcon.2018.11.015>

KUECHLER, B.; VAISHNAVI, V. On theory development in design science research: Anatomy of a research project. **European Journal of Information Systems**, [s. l.], v. 17, n. 5, p. 489–504, 2008. Available at: <https://doi.org/10.1057/ejis.2008.40>

LACERDA, D. P. *et al.* Design science research: a research method to production engineering. **Gestão Produção**, [s. l.], v. 20, n. 4, p. 741–761, 2013. Available at: <https://doi.org/10.1590/S0104-530X2013005000014>

LAM, P. T. I.; WONG, F. W. H. Improving building project performance: How buildability benchmarking can help. **Construction Management and Economics**, [s. l.], v. 27, n. 1, p.

41–52, 2009. Available at: <https://doi.org/10.1080/01446190802570498>

LANTELME, E.; OLIVEIRA, M.; FORMOSO, C. Análise da implantação de indicadores de qualidade e produtividade na construção civil. **Encontro Nacional de Tecnologia do Ambiente Construído**, [s. l.], p. 20–22, 1995.

LANTELME, E. M. V. **Proposta de um sistema de indicadores de qualidade e produtividade para a construção civil**. 111 f. 1994. - Universidade Federal do Rio Grande do Sul, Porto Alegre, 1994.

LATIFFI, A. A. *et al.* Building Information Modeling (BIM) Application in Malaysian Construction Industry. **International Journal of Construction Engineering and Management**, [s. l.], v. 2, p. 1–6, 2013. Available at: <https://doi.org/10.5923/s.ijcem.201309.01>

LEE, J. K. *et al.* Development of space database for automated building design review systems. **Automation in Construction**, [s. l.], v. 24, p. 203–212, 2012. Available at: <https://doi.org/10.1016/j.autcon.2012.03.002>

LIMA, M. M. X. de. **Metamodelo para integração de multidesempenhos em projeto de arquitetura**. 348 f. 2016. - UNICAMP, [s. l.], 2016.

LIU, R.; ISSA, R. R. A. Design for maintenance accessibility using BIM tools. **Facilities**, [s. l.], v. 32, n. 3, p. 153–159, 2014. Available at: <https://doi.org/10.1108/F-09-2011-0078>

MACIT İLAL, S.; GÜNAYDIN, H. M. Computer representation of building codes for automated compliance checking. **Automation in Construction**, [s. l.], v. 82, n. April, p. 43–58, 2017. Available at: <https://doi.org/10.1016/j.autcon.2017.06.018>

MANSON, N. Is operations research really research? **ORiON**, [s. l.], v. 22, n. 2, p. 155–180, 2006. Available at: <https://doi.org/10.5784/22-2-40>

MASCARÓ, J. L. **O custo das decisões arquitetônicas**. 5<sup>a</sup>ed. [S. l.]: MASQUATRO, 2010.

MATTOS, A. D. **Sistema de indicadores de construções prediais**. [S. l.: s. n.], 2015.

MICHAEL, T. **Current application of graphical programming in the design phase of a BIM project: Development opportunities and future scenarios with 'Dynamo**. 2016. - Northumbria University, [s. l.], 2016.

MOUSIADIS, T.; MENGANA, S. **Parametric BIM: Energy Performance Analysis Using Dynamo for Revit**. 1–55 f. 2016. - KTH Royal Institute of Technology, [s. l.], 2016.

MYDIN, S. H. *et al.* ARTIGO 2B\_Buildability attributes at design phase in malaysian building construction. **International Journal of Sustainable Construction Engineering**, [s. l.], v. 2, n. 1, p. 24–43, 2011.

MYERS, B. A. Taxonomies of visual programming and program visualization. **Journal of Visual Languages and Computing**, [s. l.], v. 1, n. 1, p. 97–123, 1990. Available at: [https://doi.org/10.1016/S1045-926X\(05\)80036-9](https://doi.org/10.1016/S1045-926X(05)80036-9)



NARLOCH, T. B. **Modelo indicador da construtibilidade a partir da análise geométrica do projeto**. 2015. - Universidade Federal de Santa Catarina, [s. l.], 2015.

NAWARI, N. O. Automating Codes Conformance. **Journal of Architectural Engineering**, [s. l.], v. 18, n. 4, p. 315–323, 2012. Available at: [https://doi.org/10.1061/\(asce\)ae.1943-5568.0000049](https://doi.org/10.1061/(asce)ae.1943-5568.0000049)

NEMBRINI, J.; SAMBERGER, S.; LABELLE, G. Parametric scripting for early design performance simulation. **Energy and Buildings**, [s. l.], v. 68, n. PART C, p. 786–798, 2014. Available at: <https://doi.org/10.1016/j.enbuild.2013.09.044>

OLIVEIRA, M.; FREITAS, H. M. R. **Indicadores de qualidade de projeto para edificações: o caso do Campus Universitário da UFRGS**. [S. l.]: Universidade Federal do Rio Grande do Sul, 1996.

OLIVEIRA, R. R. Sistematização e listagem de fatores que afetam a construtibilidade das alvenarias estruturais. **INTERNATIONAL SEMINAR ON STRUCTURAL MASONRY FOR DEVELOPING COUNTRIES**, [s. l.], 1994.

PALIARI, J. C.; SOUZA, U. E. L. de. **Método simplificado para prognóstico do consumo unitário de materiais e da produtividade da mão-de-obra: sistemas prediais hidráulicos**. São Paulo: Boletim Técnico Da Escola Politécnica Da USP Departamento de Engenharia De Construção Civil, 2008.

PAUWELS, P. *et al.* A semantic rule checking environment for building performance checking. **Automation in Construction**, [s. l.], v. 20, n. 5, p. 506–518, 2011. Available at: <https://doi.org/10.1016/j.autcon.2010.11.017>

PEFFERS, K. *et al.* A Design Science Research Methodology for Information Systems Research. **Journal of Management Information Systems**, [s. l.], v. 24, n. 3, p. 45–77, 2008. Available at: <https://doi.org/10.2753/mis0742-1222240302>

PREIDEL, C.; BORRMANN, A. Automated Code Compliance Checking Based on a Visual Language and Building Information Modeling. *In*: , 2015. **Proceedings of the 32nd International Symposium of Automation and Robotics in Construction**. [S. l.: s. n.], 2015. p. 256–263. Available at: <https://doi.org/10.13140/RG.2.1.1542.2805>

PREIDEL, C.; BORRMANN, A. Towards Code Compliance Checking on the basis of a Visual Programming Language. **Journal of Information Technology in Construction (ITcon)**, [s. l.], v. 21, n. November, p. 402–421, 2016. Available at: <http://www.itcon.org/2016/25>

PREIDEL, C.; DAUM, S.; BORRMANN, A. Data retrieval from building information models based on visual programming. **Visualization in Engineering**, [s. l.], v. 5, n. 1, 2017. Available at: <https://doi.org/10.1186/s40327-017-0055-0>

PRIMER, D. **The Dynamo Primer**. [S. l.], 2018. Available at: <http://primer.dynamobim.org/en/>. Acesso em: 12 out. 2018.

REINHART, C. F.; MARDALJEVIC, J.; ROGERS, Z. Dynamic daylight performance

metrics for sustainable building design. **LEUKOS - Journal of Illuminating Engineering Society of North America**, [s. l.], v. 3, n. 1, p. 7–31, 2006. Available at: <https://doi.org/10.1582/LEUKOS.2006.03.01.001>

RODRÍGUEZ, M.; HEINECK, L. A construtibilidade no processo de projeto de edificações. *In:* , 2003, São Carlos, SP. **II Sibragec - Simpósio Brasileiro de Gestão e Economia da Construção**. São Carlos, SP: [s. n.], 2003. Available at: <https://doi.org/10.1007/BF00048972>

SCHWABE, K.; TEIZER, J.; KÖNIG, M. Applying rule-based model-checking to construction site layout planning tasks. **Automation in Construction**, [s. l.], v. 97, n. October 2018, p. 205–219, 2019. Available at: <https://doi.org/10.1016/j.autcon.2018.10.012>

SINGER, D.; BORRMANN, A. A novel knowledge-based engineering approach for infrastructure design. **Civil-Comp Proceedings**, [s. l.], v. 109, p. 1–14, 2015.

SLAUGHTER, E. S. Design strategies to increase building flexibility. **Building Research and Information**, [s. l.], v. 29, n. 3, p. 208–217, 2001. Available at: <https://doi.org/10.1080/09613210010027693>

SOLIHIN, W.; EASTMAN, C. Classification of rules for automated BIM rule checking development. **Automation in Construction**, [s. l.], v. 53, p. 69–82, 2015. Available at: <https://doi.org/10.1016/j.autcon.2015.03.003>

SOLIHIN, Wawan. Lessons learned from experience of code-checking implementation in Singapore Success, Challenges, and Future Outlook. *In:* , 2004, Singapura. **BuildingSMART Conference**. Singapura: [s. n.], 2004.

SONEGO, M.; ECHEVESTE, M. S. Seleção de métodos para modularização no desenvolvimento de produtos: revisão sistemática. **Production**, [s. l.], v. 26, n. 2, p. 476–487, 2015. Available at: <https://doi.org/10.1590/0103-6513.138413>

SUCCAR, B. **Building Information Modelling : conceptual constructs and performance improvement tools**. 202 f. 2013. - Faculty of Engineering and Built Environment University of Newcastle, [s. l.], 2013. Available at: <https://doi.org/10.1016/j.autcon.2013.05.016>

UHM, M. *et al.* Requirements for computational rule checking of requests for proposals (RFPs) for building designs in South Korea. **Advanced Engineering Informatics**, [s. l.], v. 29, n. 3, p. 602–615, 2015. Available at: <https://doi.org/10.1016/j.aei.2015.05.006>

VAISHNAVI, V.; KUECHLER, B.; PETTER, S. **Design Science Research in Informaiton System**. [S. l.], 2004. Available at: <https://doi.org/10.1007/978-3-642-29863-9>.

VRIJHOEF, R.; KOSKELA, L. A Critical Review of Construction as a Project-based. Industry: Identifying Paths Towards a Project-independent Approach to Construction. *In:*  LEARNING FROM EXPERIENCE: NEW CHALLENGES, THEORIES AND PRACTICES IN CONSTRUCTION. Helsink (England): Technical Research Centre of Finland & Association of Finnish Civil Engineers, 2005. p. 12.

WCDE, W. C. on E. and D. **The Brundtland Report: “Our Common Future”**Oxford University Press. Oxford: [s. n.], 1987. Available at:

<https://doi.org/10.1080/07488008808408783>.

XIA, bo; CHAN, A. P. c. Measuring complexity for building projects: A Delphi study. **Engineering, Construction and Architectural Management**, [s. l.], v. 19, n. 1, p. 7–24, 2012. Available at: <https://doi.org/10.1108/09699981211192544>

YALCINKAYA, M.; SINGH, V. Patterns and trends in Building Information Modeling (BIM) research: A Latent Semantic Analysis. **Automation in Construction**, [s. l.], v. 59, p. 68–80, 2015. Available at: <https://doi.org/10.1016/j.autcon.2015.07.012>

ZARDO, P.; DA SILVA, J. L.; MUSSI, A. Q. Dynamo e Grasshopper: difusão científica e popularidade entre usuários. *In:* , 2017, Passo Fundo. **X Mostra de Pesquisa de Pós-Graduação - IMED**. Passo Fundo: [s. n.], 2017. p. 45–53.

ZHANG, C. *et al.* Quantitative Assessment of Building Constructability Using BIM and 4D Simulation. **Open Journal of Civil Engineering**, [s. l.], v. 06, n. 03, p. 442–461, 2016. Available at: <https://doi.org/10.4236/ojce.2016.63037>

ZHANG, S. *et al.* BIM-based fall hazard identification and prevention in construction safety planning. **Safety Science**, [s. l.], v. 72, p. 31–45, 2015. Available at: <https://doi.org/10.1016/j.ssci.2014.08.001>

ZHANG, S. *et al.* Building Information Modeling (BIM) and Safety: Automatic Safety Checking of Construction Models and Schedules. **Automation in Construction**, [s. l.], v. 29, p. 183–195, 2013. Available at: <https://doi.org/10.1016/j.autcon.2012.05.006>

## APÊNDICE

Esta seção mostra o resumo dos algoritmos desenvolvidos, mostrando entradas, saídas e etapas do processo. Explica a finalidade do critérios e apresenta a devida classificação. São divididos nas categorias, conforme apresentados no trabalho.

### Construtibilidade

Categoria	N	Autor	Nome	Equação	Parâmetros	
Construtibilidade	<b>C1</b>	Lantelme (1994) Mascaró (2010)	Índice de Compacidade	$IC=2*\sqrt{(\pi*Apav)/Pp}$	$Apav = \text{Área do pavimento}$ $Pp = \text{Perímetro do pavimento}$	
<b>Descrição</b>	Representa inversamente a complexidade geométrica do perímetro do pavimento. Quanto mais distante de um quadrado (0,84), menor a construtibilidade.					
Etapa	Entrada			Processo	Saída	
	Tipo	Fonte	Entrada	Etapas	Resultado	Und
Algoritmo	Categoria	Elementos do Modelo	Área Pavimento	Coletar perímetro do pavimento (Apav)	$2*\sqrt{[\pi*(B+C+D)]/Apav}$	
	Categoria	Elementos do Modelo	Paredes	Coletar parâmetro da área das paredes (B)		
<b>Classificação</b>	Categoria	Elementos do Modelo	Pilares estruturais	Coletar parâmetro da área da base dos pilares (C)		
II	Categoria	Elementos do Modelo	Pisos	Coletar parâmetro área dos pisos (D)		

Categoria	N	Autor	Nome	Equação	Parâmetros	
Construtibilidade	<b>C2</b>	Oliveira, Lantelme e Formoso (1995)	Índice de Área Molhada	$Iam = Am/Apav$	$Apav = \text{Área do pavimento}$ ; $Am = \text{Área molhada no pavimento}$	
<b>Descrição</b>	Áreas molhadas exigem maior quantidade de serviços devido a impermeabilização, testes e uso de cerâmicas nas alvenarias.					
Etapa	Entrada			Processo	Saída	
	Tipo	Fonte	Entrada	Etapas	Resultado	Und
Algoritmo	Categoria	Elementos do Modelo	Pisos	Coletar área de todos os pisos (A)	Calcular relação (B/A) %	
<b>Classificação</b>						
II	Booleano	Usuário	Indicação de Piso molhado	Filtrar por piso molhado e coletar área molhada(B)		

<b>Categoria</b>	<b>N</b>	<b>Autor</b>	<b>Nome</b>	<b>Equação</b>	<b>Parâmetros</b>	
Construtibilidade	<b>C3</b>	Oliveira, Lantelme e Formoso (1995)	Densidade de Paredes	$DP = \text{Apar} / \text{Apavt}$	<i>Apavt = Área do pavimento</i> <i>Apar = Área de paredes</i>	
<b>Descrição</b>	Este índice tem o objetivo de verificar o grau de otimização da compartimentação do pavimento tipo					
<b>Etapa</b>	<b>Entrada</b>			<b>Processo</b>	<b>Saída</b>	
	<i>Tipo</i>	<i>Fonte</i>	<i>Entrada</i>	<i>Etapas</i>	<i>Resultado</i>	<i>Und</i>
Algoritmo	Categoria	Elementos do Modelo	Paredes	Calcular área da projeção das paredes (A)	Calcular relação $A / (A+B+C)$	%
<b>Classificação</b>	Categoria	Elementos do Modelo	Pilares	Calcular área da base dos pilares (B)		
II	Categoria	Elementos do Modelo	Pisos	Coletar parâmetro área dos pisos (C)		

<b>Categoria</b>	<b>N</b>	<b>Autor</b>	<b>Nome</b>	<b>Equação</b>	<b>Parâmetros</b>	
Construtibilidade	<b>C4</b>	Oliveira e Freitas (1996), Mascaró (2010) e Narloch (2015)	Índice de Fachadas	$IF = \text{Af} / \text{Apav}$	<i>Af = Área de fachadas do pavimento tipo</i> <i>Apav = Área total do pavimento</i>	
<b>Descrição</b>	Os planos verticais das fachadas são mais difíceis e custosos de serem construídos. O indicador revela a proporção de fachadas em relação ao pavimento tipo.					
<b>Etapa</b>	<b>Entrada</b>			<b>Processo</b>	<b>Saída</b>	
	<i>Tipo</i>	<i>Fonte</i>	<i>Entrada</i>	<i>Etapas</i>	<i>Resultado</i>	<i>Und</i>
Algoritmo	Categoria	Elementos do Modelo	Área	Coletar área de paredes externas do pavimento (A)	Calcular relação $A / (B+C+D)$	%
<b>Classificação</b>	Categoria	Elementos do Modelo	Paredes	Calcular área da projeção das paredes (B)		
II	Categoria	Elementos do Modelo	Pisos	Coletar parâmetro área dos pisos (D)		

<b>Categoria</b>	<b>N</b>	<b>Autor</b>	<b>Nome</b>	<b>Equação</b>	<b>Parâmetros</b>	
Construtibilidade	<b>C5</b>	Oliveira, Lantelme e Formoso (1995)	Densidade de Esquadrias	$DE = Aes/Apar$	<i>Aes = Área de esquadrias (portas e janelas) Apar = Área de paredes</i>	
<b>Descrição</b>	Esquadrias demandam maior número de serviços e aumentam a complexidade construtiva					
<b>Etapa</b>	<b>Entrada</b>			<b>Processo</b>	<b>Saída</b>	
	<i>Tipo</i>	<i>Fonte</i>	<i>Entrada</i>	<i>Etapas</i>	<i>Resultado</i>	<i>Und</i>
Algoritmo	Categoria	Elementos do Modelo	Paredes	Coletar área de todas as paredes (A)	Densidade de esquadrias ((B+D)/A) %	
<b>Classificação</b>	Categoria	Elementos do Modelo	Janelas	Coletar área de todas as janelas (B)		
II	Categoria	Elementos do Modelo	Portas	Coletar área de todas as portas (D)		

<b>Categoria</b>	<b>N</b>	<b>Autor</b>	<b>Nome</b>	<b>Equação</b>	<b>Parâmetros</b>	
Construtibilidade	<b>C6</b>	Oliveira, Lantelme e Formoso (1995)	Índice de Padronização de Esquadrias	$IE = Edif/Etot$	<i>Edif = Número de esquadrias diferentes; Etot = Número total de esquadrias</i>	
<b>Descrição</b>	A maior diversidade de esquadrias influi na complexidade do projeto, na compra, e no planejamento e operação de execução.					
<b>Etapa</b>	<b>Entrada</b>			<b>Processo</b>	<b>Saída</b>	
	<i>Tipo</i>	<i>Fonte</i>	<i>Entrada</i>	<i>Etapas</i>	<i>Resultado</i>	<i>Und</i>
Algoritmo	Categoria	Elementos do Modelo	Janelas	Coletar quantidade de todas as janelas (A)	Calcular relação 1- (C+D)/(B+A) %	
	Categoria	Elementos do Modelo	Portas	Coletar quantidade de todas as portas (B)		
<b>Classificação</b>	Categoria	Elementos do Modelo	Janelas	Coletar todos os tipos de janelas (C)		
II	Categoria	Elementos do Modelo	Portas	Coletar todos os tipos de portas (D)		

<b>Categoria</b>	<b>N</b>	<b>Autor</b>	<b>Nome</b>	<b>Equação</b>	<b>Parâmetros</b>	
Construtibilidade	<b>C7</b>	Jarkas (2010) Franco (2014)	Densidade de Pilares	$Dpi = Abp/Np$	<i>Abp = Área da base dos pilares; Np = Número de pilares ajustados</i>	
<b>Descrição</b>		Pilares restringem a movimentação na obra e aumentam pontos de fundações.				
<b>Etapa</b>	<b>Entrada</b>			<b>Processo</b>	<b>Saída</b>	
	<i>Tipo</i>	<i>Fonte</i>	<i>Entrada</i>	<i>Etapas</i>	<i>Resultado</i>	<i>Und</i>
Algoritmo	Categoria	Elementos do Modelo	Pilares estruturais	Coletar área da base dos pilares (A)	Calcular relação A/B	%
<b>Classificação</b>				Coletar número de pilares, maior peso para pilares de geometria irregular (B)		
II						

<b>Categoria</b>	<b>N</b>	<b>Autor</b>	<b>Nome</b>	<b>Equação</b>	<b>Parâmetros</b>	
Construtibilidade	<b>C8</b>	Jarkas (2010)	Densidade de Vigas	Se $Cv/ Apv \leq 0,45$ (1) se não (2). (1) $DPv = Cv/(0,45*Apv)$ (2) $DPv = 2 - Cv/(0,45*Apv)$	<i>Cv= Comprimento de vigas Apav = Área do pavimento</i>	
<b>Descrição</b>		Este índice representa a eficiência do projeto. Quanto menor este valor, menores serão as complexidades dos serviços de formas e concretagem, também reduzindo interferências.				
<b>Etapa</b>	<b>Entrada</b>			<b>Processo</b>	<b>Saída</b>	
	<i>Tipo</i>	<i>Fonte</i>	<i>Entrada</i>	<i>Etapas</i>	<i>Resultado</i>	<i>Und</i>
Algoritmo	Categoria	Elementos do Modelo	Quadros estruturais	Coletar comprimento das vigas (A)	Calcular relação $A/(B+C+D)$	%
	Categoria	Elementos do Modelo	Paredes	Calcular da área da projeção das paredes (B)		
<b>Classificação</b>	Categoria	Elementos do Modelo	Pilares estruturais	Calcular área da base dos pilares (C)		
II	Categoria	Elementos do Modelo	Pisos	Coletar parâmetro área dos pisos (D)		

<b>Categoria</b>	<b>N</b>	<b>Autor</b>	<b>Nome</b>	<b>Equação</b>	<b>Parâmetros</b>
Construtibilidade	<b>C9</b>	Jarkas (2010)	Índice de Padronização dos Pilares	$I_{pp} = P_{dif}/P_{tot}$	<i>P<sub>dif</sub> = Pilares diferentes</i> <i>P<sub>tot</sub> = Pilares totais</i>
<b>Descrição</b>					
Este índice considera a complexidade na individualidade de tipos estruturais, através da relação de quantidades de pilares diferentes em suas seções transversais e o número total de pilares.					
<b>Etapa</b>	<b>Entrada</b>			<b>Processo</b>	<b>Saída</b>
	<i>Tipo</i>	<i>Fonte</i>	<i>Entrada</i>	<i>Etapas</i>	<i>Resultado</i> <i>Und</i>
Algoritmo	Categoria	Elementos do Modelo	Pilares estruturais	Coletar número de pilares (A)	Calcular relação 1 - B/A   %
<b>Classificação</b>				Coletar tipos de pilares (B)	
II					

<b>Categoria</b>	<b>N</b>	<b>Autor</b>	<b>Nome</b>	<b>Equação</b>	<b>Parâmetros</b>
Construtibilidade	<b>C10</b>	Jarkas (2010)	Índice de Padronização das Vigas	$I_{pv} = V_{dif}/V_{tot}$	<i>V<sub>dif</sub> = Vigas diferentes</i> <i>V<sub>tot</sub> = Vigas totais</i>
<b>Descrição</b>					
Este índice considera a complexidade na individualidade de tipos estruturais, através da relação de quantidades de vigas diferentes em suas seções transversais e o número total de vigas.					
<b>Etapa</b>	<b>Entrada</b>			<b>Processo</b>	<b>Saída</b>
	<i>Tipo</i>	<i>Fonte</i>	<i>Entrada</i>	<i>Etapas</i>	<i>Resultado</i> <i>Und</i>
Algoritmo	Categoria	Elementos do Modelo	Quadros estruturais	Coletar número de vigas (A)	Calcular relação B/A   %
<b>Classificação</b>				Coletar tipos de vigas(B)	
II					



<b>Categoria</b>	<b>N</b>	<b>Autor</b>	<b>Nome</b>	<b>Equação</b>	<b>Parâmetros</b>	
Construtibilidade	<b>C11</b>	Jarkas (2010)	Índice de Padronização de Lajes	$I_{pl} = L_{dif}/L_{tot}$	$L_{dif} = \text{Lajes diferentes}$ $L_{tot} = \text{Lajes totais}$	
<b>Descrição</b>	Este índice considera a complexidade na individualidade de tipos estruturais, através da relação de quantidades de lajes diferentes em suas seções transversais e o número total de lajes.					
<b>Etapa</b>	<b>Entrada</b>			<b>Processo</b>	<b>Saída</b>	
	<i>Tipo</i>	<i>Fonte</i>	<i>Entrada</i>	<i>Etapas</i>	<i>Resultado</i>	<i>Und</i>
Algoritmo	Categoria	Elementos do Modelo	Pisos Estruturais	Coletar número de lajes (A)	Calcular relação B/A	%
<b>Classificação</b>				Coletar tipos de lajes (B)		
II						

<b>Categoria</b>	<b>N</b>	<b>Autor</b>	<b>Nome</b>	<b>Equação</b>	<b>Parâmetros</b>	
Construtibilidade	<b>C12</b>	Matos e Paliari (2010)	Índice de Tubulações	$I_{tub} = C_{tub}/A_{pav}$	$C_{tub} = \text{Comprimento de tubulações}$ $A_{pav} = \text{Área do pavimento}$	
<b>Descrição</b>	Este índice relaciona com a densidade de tubulações na área privativa do pavimento. De modo que quanto maior este índice, menores são as produtividades destes serviços.					
<b>Etapa</b>	<b>Entrada</b>			<b>Processo</b>	<b>Saída</b>	
	<i>Tipo</i>	<i>Fonte</i>	<i>Entrada</i>	<i>Etapas</i>	<i>Resultado</i>	<i>Und</i>
Algoritmo	Categoria	Elementos do Modelo	Tubulação	Coletar comprimento das tubulações (A)	Calcular relação A/(B+C+D)	%
	Categoria	Elementos do Modelo	Paredes	Coletar parâmetro da área das paredes (B)		
<b>Classificação</b>	Categoria	Elementos do Modelo	Pilares estruturais	Coletar parâmetro da área da base dos pilares (C)		
II	Categoria	Elementos do Modelo	Pisos	Coletar parâmetro área dos pisos (D)		

<b>Categoria</b>	<b>N</b>	<b>Autor</b>	<b>Nome</b>	<b>Equação</b>	<b>Parâmetros</b>	
Construtibilidade	<b>C13</b>	Tubino (2014), Oliveira, Lantel e Formoso (1995)	Índice de Conduites	$I_{cond} = C_{cond}/A_{pav}$	$C_{cond} =$ <i>Comprimento de conduites</i> $A_{pav} =$ <i>Área do pavimento</i>	
<b>Descrição</b>	Lantelme propõe o índice de pontos por comprimento de conduíte, devendo representar a performance do projeto e melhor aproveitamento dos eletrodutos.					
<b>Etapa</b>	<b>Entrada</b>			<b>Processo</b>	<b>Saida</b>	
	<i>Tipo</i>	<i>Fonte</i>	<i>Entrada</i>	<i>Etapas</i>	<i>Resultado</i>	<i>Und</i>
Algoritimo	Categoria	Elementos do Modelo	Conduítes	Coletar comprimento dos conduites (A)	Calcular relação $A/(B+C+D)$	%
	Categoria	Elementos do Modelo	Paredes	Coletar parâmetro da área das paredes (B)		
<b>Classificação</b>	Categoria	Elementos do Modelo	Pilares estruturais	Coletar parâmetro da área da base dos pilares (C)		
II	Categoria	Elementos do Modelo	Pisos	Coletar parâmetro área dos pisos (D)		

## Flexibilidade

Categoria	N	Autor	Nome	Equação	Parâmetros	
Flexibilidade	F1	Cavallieri (2019)	Modularidade da estrutura	$\text{Mod} = (N_{\text{modx}}/N_x) * Y_x + (N_{\text{mody}}/N_y) * Y_y + (N_{\text{modz}}/N_z) * Y_z; \quad Y_i = (N_i * c_{d_i}) / (N_x * c_{d_x} + N_y * c_{d_y} + N_z * c_{d_z})$	<i>N<sub>modi</sub></i> = Número de vãos que possuem um modulo em comum; <i>N<sub>i</sub></i> = Número de vãos; <i>Y<sub>i</sub></i> = Fator de peso; <i>c<sub>d<sub>i</sub></sub></i> = Fator de direção onde <i>c<sub>d<sub>x</sub></sub></i> =0,4, <i>c<sub>d<sub>y</sub></sub></i> =0,4 e <i>c<sub>d<sub>z</sub></sub></i> =0,2	
<b>Descrição</b>	Analisa a modularidade da estrutura, as distâncias entre os elementos de suporte de carga seguem padrões. Quanto mais a estrutura de um edifício é modular, mais ela é flexível.					
Etapa	Entrada			Processo	Saída	
	Tipo	Fonte	Entrada	Etapas	Resultado	Und
Algoritmo	Categoria	Elementos do Modelo	Pilares Estruturais	Coleta posição dos Pilares	Aplicar conjunto de informações a equação	%
<b>Classificação</b>				Ordenando pontos nas direções X, Y e Z		
				Agrupando pontos nas direções X, Y e Z		
				Calculando a distância entre um grupo e o próximo.		
III						

Categoria	N	Autor	Nome	Equação	Parâmetros	
Flexibilidade	F2	Cavallieri (2019)	Posição do Shaft	$OS = [1 - (Xc/X/2) + (Yc/Y/2)] / Nc$	<i>Xc</i> = Distância entre o baricentro do shaft e o baricentro do projeto em X; <i>Yc</i> = Distância entre o baricentro do shaft e o baricentro do projeto em Y; <i>X</i> e <i>Y</i> = Lados do retângulo que circunscreve o projeto; <i>Nc</i> = número de shafts	
<b>Descrição</b>	Quantifica a distribuição das prumadas de instalações em relação aos ambientes do projeto, se forem melhor distribuídos permitem maior flexibilização no uso					
Etapa	Entrada			Processo	Saída	
	Tipo	Fonte	Entrada	Etapas	Resultado	Und
Algoritmo	Categoria	Elementos do Modelo	Ambientes	Coleta posição dos Shafts	Aplicar conjunto de informações a equação	%
<b>Classificação</b>				Calcular envoltória geométrica sobre os pisos. Identificar geometria das arestas		
				Selecionar a maior Aresta em cada dimensão		
III	Categoria	Elementos do Modelo	Ambientes	Calcular baricentro dos shafts e o baricentro dos pisos		

Categoria	N	Autor	Nome	Equação	Parâmetros	
Flexibilidade	F3	Cavallieri (2019)	Elementos construtivos removíveis	$ER = P(Apr/Ap) + Pe(Aper/Ape) + Pi*(Apir/Api) + E(Cer/Ce);$ onde: Ip=0,4; Ipe = 0,3; Ipi = 0,2; Ie = 0,1	<i>I = Pesos por categorias; Apr = Área de partições removíveis; Ap = área de partições; Aper = Área de paredes externas removíveis; Ape = Área paredes externas; Apir = área pisos removíveis; Api = Área pisos; Cer = Comprimento de elementos estruturais removíveis; Ce = Comprimento de Elementos Estruturais</i>	
<b>Descrição</b>	Considera quatro categorias de elementos de construção removíveis, nomeadamente paredes divisórias, paredes externas, pisos e elementos de suporte de carga (vigas e pilares), avalia a percentagem de elementos de construção removíveis em comparação com o total					
Etapa	Entrada			Processo	Saída	
	Tipo	Fonte	Entrada	Etapas	Resultado	Und
Algoritmo	Categoria	Elementos do Modelo	Paredes/ Pisos/ Pilares/ Vigas/ Lajes	Identificar elementos de cada uma das categorias, utilização função da parede para diferenciar partições de paredes externas	Aplicar conjunto de informações a equação	%
<b>Classificação</b>	Booleano	Usuário	Identificar Elementos Removíveis			
III	Booleano	Parâmetro de Tipo	Função da Parede	Calcular áreas e comprimentos		

Categoria	N	Autor	Nome	Equação	Parâmetros	
Flexibilidade	F4	(CAVALLIERE et al., 2019)	Porcentagem e orientação das janelas	$\sum_j \frac{L_w + L_{cw}}{L_W} \times \frac{1}{4}$	<i>j = Orientação da construção Lw= Largura total das janelas Lcw = Largura total das paredes cortina LW = Largura total das paredes</i>	
<b>Descrição</b>	Estabelecer um indicador para a relação entre o número e a extensão das aberturas em cada orientação, apontando sua suficiência em ajudar na redistribuição dos espaços internos.					
Etapa	Entrada			Processo	Saída	
	Tipo	Fonte	Entrada	Etapas	Resultado	Und
Algoritmo	Categoria	Elementos do Modelo	Paredes	Calcular orientação das paredes externas (J)	Calcular indicador, aplicando parâmetros na equação	%
	Categoria	Elementos do Modelo	Janelas	Coletar parâmetro largura das janelas (Lw) por orientação		
<b>Classificação</b>	Categoria	Elementos do Modelo	Paredes Cortina	Coletar parâmetro largura das paredes cortina (Lcw) por orientação		
IV	Categoria	Elementos do Modelo	Função das Paredes	Coletar parâmetro comprimento das paredes (LW) por orientação		

<b>Categoria</b>	<b>N</b>	<b>Autor</b>	<b>Nome</b>	<b>Equação</b>	<b>Parâmetros</b>	
Flexibilidade	<b>F5</b>	Cavallieri (2019)	Partições internas móveis	$PMo = Pm/P$	<i>Pm = Área de paredes móveis; P = Área de paredes;</i>	
<b>Descrição</b>	Quantifica a distribuição das prumadas de instalações em relação aos ambientes do projeto, se forem melhor distribuídos permitem maior flexibilização no uso					
<b>Etapas</b>	<b>Entrada</b>			<b>Processo</b>	<b>Saida</b>	
	<i>Tipo</i>	<i>Fonte</i>	<i>Entrada</i>	<i>Etapas</i>	<i>Resultado</i>	<i>Und</i>
Algoritimo	Categoria	Elementos do Modelo	Paredes	Identificar paredes móveis	Aplicar conjunto de informações a equação %	
				Indetificar paredes internas		
<b>Classificação</b>	Booleano	Usuário	Identificar paredes removíveis	Calcular área de todas as paredes internas		
				III		

## Sustentabilidade

<b>Categoria</b>	<b>N</b>	<b>Autor</b>	<b>Nome</b>	<b>Equação</b>	<b>Parâmetros</b>
Sustentabilidade	S1	Cavallieri (2019)	Índice de Permeabilidade	$I_p = A_p/A$	$A_p = \text{Área total}; A = \text{Área do terreno}$
<b>Descrição</b>	Razão que representa a parte da área do terreno que está impermeabilizada devido ao empreendimento				
<b>Etapa</b>	<b>Entrada</b>			<b>Processo</b>	<b>Saída</b>
	<i>Tipo</i>	<i>Fonte</i>	<i>Entrada</i>	<i>Etapas</i>	<i>Resultado</i> <i>Und</i>
Algoritmo	Categoria	Elementos do Modelo	Paredes/ Pisos/ Pilares/ lajes/ Terreno	Calcular projeção do Subsolo utilizando elementos coletados Coletar projeção do Pilotis utilizando os elementos coletados	Soma a área remanescente do terreno com a área permeável dos pilotis e divide pela área do total do terreno
<b>Classificação</b>	Valor	Usuário	Nº de Subsolos	Coleta geometria do terreno, realizando a subtração da geometria das projeções do mesmo	%
III	Valor	Usuário	Permeabilidade Pisos	Soma áreas do pilotis fora da projeção dos subsolos com pisos permeáveis,	

<b>Categoria</b>	<b>N</b>	<b>Autor</b>	<b>Nome</b>	<b>Equação</b>	<b>Parâmetros</b>
Sustentabilidade	S2	Cavallieri (2019)	Uso de materiais com conteúdo reciclado	$Re = M_r/M_t$	$M_r = \text{Massa reciclável}; M_t = \text{Massa total}$
<b>Descrição</b>	Participação de materiais reciclados dentro da massa total dos materiais utilizados na edificação				
<b>Etapa</b>	<b>Entrada</b>			<b>Processo</b>	<b>Saída</b>
	<i>Tipo</i>	<i>Fonte</i>	<i>Entrada</i>	<i>Etapas</i>	<i>Resultado</i> <i>Und</i>
Algoritmo	Categoria	Elementos do Modelo	Todos	Coleta todos os elementos do modelo Filtra os elementos marcados como reciclável	Aplica na equação
<b>Classificação</b>	Valor	Usuário	Densidade	Calcula massa total	%
III	Booleano	Usuário	É reciclável?	Calcula massa reciclável	

<b>Categoria</b>	<b>N</b>	<b>Autor</b>	<b>Nome</b>	<b>Equação</b>	<b>Parâmetros</b>	
Sustentabilidade	S3	Cavallieri (2019)	Uso de materiais orgânicos	$Or = Mo/Mt$	$Mo = \text{Massa Orgânica}; Mt = \text{Massa total}$	
<b>Descrição</b>	Participação de materiais de origem orgânicas no empreendimento. Todos os produtos à base de madeira e orgânicos devem ter sua certificação identificada					
<b>Etapa</b>	<b>Entrada</b>			<b>Processo</b>	<b>Saída</b>	
	<i>Tipo</i>	<i>Fonte</i>	<i>Entrada</i>	<i>Etapas</i>	<i>Resultado</i>	<i>Und</i>
Algoritmo	Categoria	Elementos do Modelo	Todos	Coleta todos os elementos do modelo Filtrar os elementos marcados como reciclável	Aplica na equação	%
<b>Classificação</b>	Valor	Usuário	Densidade	Calcula massa total		
III	Booleano	Usuário	É orgânico?	Calcula massa orgânica		

<b>Categoria</b>	<b>N</b>	<b>Autor</b>	<b>Nome</b>	<b>Equação</b>	<b>Parâmetros</b>	
Sustentabilidade	S4	Cavallieri (2019)	Uso de substitutos de concreto	$SC = Msc/Mtc$	$Mr = \text{Massa de substitutos}; Mt = \text{Massa total de concreto}$	
<b>Descrição</b>	Participação de substitutos de cimento no concreto utilizado no empreendimento					
<b>Etapa</b>	<b>Entrada</b>			<b>Processo</b>	<b>Saída</b>	
	<i>Tipo</i>	<i>Fonte</i>	<i>Entrada</i>	<i>Etapas</i>	<i>Resultado</i>	<i>Und</i>
Algoritmo	Categoria	Elementos do Modelo	Pilares/ Vigas / Lajes	Coleta elementos estruturais Calcula massa de concreto	Aplica na equação	%
<b>Classificação</b>	Valor	Usuário	Densidade	Calcula massa de substituição		
III	Valor	Usuário	% de substituição	Calcula massa de substituição		

<b>Categoria</b>	<b>N</b>	<b>Autor</b>	<b>Nome</b>	<b>Equação</b>	<b>Parâmetros</b>
Sustentabilidade	S5	Cavallieri (2019)	Acessibilidade ao Transporte Público	$D = 2 * R * \text{asin}(\text{sqrt}(\text{hav}))$ ); onde: $d_{lon} = lon2 - lon1$ $d_{lat} = lat2 - lat1$ $\text{hav} = \sin(d_{lat}/2)^2 + \cos(lat1) * \cos(lat2) * \sin(d_{lon}/2)^2$	$R = \text{Raio}$ <i>Aproximado da terra; lat = latitudes do ponto 1 e 2; long = longitude do ponto 1 e 2</i>
<b>Descrição</b>		Avaliar as distâncias de viagem do prédio até as paradas de transporte público próximas			
<b>Etapa</b>	<b>Entrada</b>			<b>Processo</b>	<b>Saída</b>
	<i>Tipo</i>	<i>Fonte</i>	<i>Entrada</i>	<i>Etapas</i>	<i>Resultado</i> <i>Und</i>
Algoritmo	Parâmetro	Configuração de Projeto	Localização do projeto	Coletar coordenadas de localização do projeto	Aplicar função a todos os pontos e selecionar a menor distância    m
<b>Classificação</b>	Base	Externa	Lista de coordenada de pontos de transporte público	Conectar-se e ler banco de dados do transporte público	
IV					

<b>Categoria</b>	<b>N</b>	<b>Autor</b>	<b>Nome</b>	<b>Equação</b>	<b>Parâmetros</b>
Sustentabilidade	S6	Cavallieri (2019)	Veículos elétricos	$VE = V_e/V$	$V_e = \text{Número de vagas para veículos elétricos}; V = \text{Número Total de vagas}$
<b>Descrição</b>		Define quantidade de vagas para veículos elétricos em relação ao total de vagas			
<b>Etapa</b>	<b>Entrada</b>			<b>Processo</b>	<b>Saída</b>
	<i>Tipo</i>	<i>Fonte</i>	<i>Entrada</i>	<i>Etapas</i>	<i>Resultado</i> <i>Und</i>
Algoritmo	Categoria	Elementos do Modelo	Vagas	Calcular número de vagas no modelo	Aplicar função a todos os pontos e selecionar a menor distância    %
<b>Classificação</b>	Booleano	Usuário	É elétrico?	Calcular número de Vagas para veículos elétricos	
II					



<b>Categoria</b>	<b>N</b>	<b>Autor</b>	<b>Nome</b>	<b>Equação</b>	<b>Parâmetros</b>
Sustentabilidade	S7	Cavallieri (2019)	Aberturas para Iluminação	$Ilu = (Aj/Ar)*i$	<i>Aj = Abertura de Janelas; Ar = área do do ambiente; i = coeficiente de iluminação por ambiente</i>
<b>Descrição</b>					Projetar ambientes com maiores áreas de aberturas para iluminação natural
<b>Etapa</b>	<b>Entrada</b>			<b>Processo</b>	<b>Saída</b>
	<i>Tipo</i>	<i>Fonte</i>	<i>Entrada</i>	<i>Etapas</i>	<i>Resultado</i> <i>Und</i>
Algoritmo	Categoria	Elementos do Modelo	Ambiente	Agrupar Janelas por ambiente Calcular área dos ambientes	Aplicar função a todos os pontos e selecionar a menor distância
<b>Classificação</b>	Categoria	Elementos do Modelo	Janelas	Calcular abertura de janelas em cada ambiente	%
III					

<b>Categoria</b>	<b>N</b>	<b>Autor</b>	<b>Nome</b>	<b>Equação</b>	<b>Parâmetros</b>
Sustentabilidade	S8	Cavallieri (2019)	Aberturas para Ventilação	$Vent = (Aj/Ar)*v$	<i>Aj = Abertura de Janelas; Ar = área do do ambiente; v = coeficiente de ventilação por tipo de ambiente</i>
<b>Descrição</b>					Projetar ambientes com maiores áreas de aberturas para ventilação natural
<b>Etapa</b>	<b>Entrada</b>			<b>Processo</b>	<b>Saída</b>
	<i>Tipo</i>	<i>Fonte</i>	<i>Entrada</i>	<i>Etapas</i>	<i>Resultado</i> <i>Und</i>
Algoritmo	Categoria	Elementos do Modelo	Ambiente	Agrupar Janelas por ambiente Calcular área dos ambientes	Aplicar função a todos os pontos e selecionar a menor distância
<b>Classificação</b>	Categoria	Elementos do Modelo	Janelas	Calcular abertura de janelas em cada ambiente	%
III					

## Conformidade

<b>Categoria</b>	<b>N</b>	<b>Autor</b>	<b>Nome</b>	<b>Equação</b>	<b>Parâmetros</b>	
Conformidade	N1	Lei de uso e Ocupação de Fortaleza; Especialistas	Índice de Aproveitamento	$IA = AC/Aterr$	IA = Índice de Aproveitamento; AC = Área Construída; Aterr = Área do Terreno.	
<b>Descrição</b>	Número que multiplicado pela área do terreno, resulta na área de construção computável, estabelecendo as condições de utilização dos instrumentos urbanísticos, jurídicos e tributários					
<b>Etapa</b>	<b>Entrada</b>			<b>Processo</b>	<b>Saída</b>	
	<i>Tipo</i>	<i>Fonte</i>	<i>Entrada</i>	<i>Etapas</i>	<i>Resultado</i>	<i>Und</i>
Algoritmo	Categoria	Elementos do modelo	Pisos/Pilares/Paredes	Soma das áreas de pisos, base de paredes e seção de pilares do modelo (A)	Calcular relação A/B	%
<b>Classificação</b>	Valor	Usuário	Possíveis benefícios de legislação	Coleta a Área Total do Terreno a partir da topografia (B)	Compara com o Limite estabelecido para o índice	bool
II	Valor	Usuário	Índice de Aproveitamento do terreno	Verifica se Paredes foram modeladas em cima do piso		

<b>Categoria</b>	<b>N</b>	<b>Autor</b>	<b>Nome</b>	<b>Equação</b>	<b>Parâmetros</b>	
Conformidade	N2	Lei de uso e Ocupação de Fortaleza; Especialistas	Taxa de Ocupação	$IA = AC/Aterr$	IA = Índice de Aproveitamento; AC = Área Construída; Aterr = Área do Terreno.	
<b>Descrição</b>	Percentual utilizado pela edificação em relação a área total do terreno, considerando apenas sua projeção horizontal, não sendo computados nesta projeção os elementos componentes das fachadas					
<b>Etapa</b>	<b>Entrada</b>			<b>Processo</b>	<b>Saída</b>	
	<i>Tipo</i>	<i>Fonte</i>	<i>Entrada</i>	<i>Etapas</i>	<i>Resultado</i>	<i>Und</i>
'Algoritmo	Categoria	Elementos do Modelo	Pisos/ Pilares/ Paredes/ Telhados/ Terreno	Coleta geometria pisos, paredes, Pilares e Telhados	Calcular relação A/B	%
<b>Classificação</b>	Valor	Usuário	Possíveis benefícios de legislação	Translada toda a geometria para o nível 0 do terreno e une-as	Compara com o Limite estabelecido para o índice	bool
III	Valor	Usuário	Taxa Ocupação do Terreno	Calcula área a área da superfície formada pela geometria unida (A)		
	III	Valor	Usuário	Coleta a Área Total do Terreno a partir da topografia (B)		

<b>Categoria</b>	<b>N</b>	<b>Autor</b>	<b>Nome</b>	<b>Equação</b>	<b>Parâmetros</b>	
Conformidade	N3	Lei de uso e Ocupação de Fortaleza; Especialistas	Recuo edificação	$R_n \geq R_{nmin}$	$R_n$ = Distância da edificação a borda do terreno em ua direção $R_{nmin}$ = Menor R admtido na direção	
<b>Descrição</b>					É a distância medida entre o limite externo da projeção da edificação no plano horizontal, e a divisa do lote.	
<b>Etapa</b>	<b>Entrada</b>			<b>Processo</b>	<b>Saída</b>	
	<i>Tipo</i>	<i>Fonte</i>	<i>Entrada</i>	<i>Etapas</i>	<i>Resultado</i>	<i>Unid</i>
Algoritmo	Categori a	Elementos do Modelo	Pisos/Pilares/Paredes /Telhados/Terreno	Gerar Geometria da Edificação a partir dos elementos de Entrada (G)	Verificar intersecção entre G e S, apontando qual a direção	bo ol
				Coletar bordas do terreno		
				Separar bordas em lateral, frontal e recuo baseado no vetor frontal		
Realizar offset das bordas em direção ao gentro do terreno no valor dos limites						
<b>Classificaçã o</b>	Valor	Usuário	Limites de recuo: Fontral, Lateral e Fundos	Gerar sólido entre o limite e o offset(S)		
III	Valor	Usuário	Indicar frente do terreno			

<b>Categoria</b>	<b>N</b>	<b>Autor</b>	<b>Nome</b>	<b>Equação</b>	<b>Parâmetros</b>	
Conformidade	N4	Lei de uso e Ocupação de Fortaleza; Especialistas	Recuo Subsolo	$R_n \geq R_{nmin}$	$R_n$ = Distância da edificação a borda do terreno em ua direção; $R_{nmin}$ = Menor R admtido na direção	
<b>Descrição</b>					É a distância medida entre o limite externo da projeção do subsolo da edificação no plano horizontal, e a divisa do lote.	
<b>Etapa</b>	<b>Entrada</b>			<b>Processo</b>	<b>Saída</b>	
	<i>Tipo</i>	<i>Fonte</i>	<i>Entrada</i>	<i>Etapas</i>	<i>Resultado</i>	<i>Unid</i>
Algoritimo	Categori a	Elementos do Modelo	Pisos/Pilares/Paredes/T elhados/Terreno	Gerar Geometria do Subsolo a partir dos elementos de Entrada (G)	Verificar intersecção entre G e S, apontando qual a direção	bo ol
				Coletar bordas do terreno		
				Separar bordas em lateral, frontal e recuo baseado no vetor frontal		
Realizar offset das bordas em direção ao centro do terreno no valor dos limites						
<b>Classificaçã o</b>	Valor	Usuário	Limites de recuo: Frontal, Lateral e Fundos	Gerar sólido entre o limite e o offset(S)		
III	Valor	Usuário	Indicar frente do terreno			

<b>Categoria</b>	<b>N</b>	<b>Autor</b>	<b>Nome</b>	<b>Equação</b>	<b>Parâmetros</b>	
Conformidade	N5	Lei de uso e Ocupação de Fortaleza; Especialistas	Recuo Entre Torres	$R_{torre} \geq R_{torremin}$	$R_{torre}$ = Distância entre construções no mesmo lote $R_{subnmin}$ = Menor $R_{subn}$ admtdto na direção	
<b>Descrição</b> É a distância medida entre as projeções das edificações em um mesmo terreno no plano horizontal						
<b>Etapa</b>	<b>Entrada</b>			<b>Processo</b>	<b>Saída</b>	
	<i>Fonte</i>	<i>Entrada</i>	<i>Etapas</i>	<i>Etapas</i>	<i>Resultado</i>	<i>Und</i>
Algoritmo	Categoria	Elementos do Modelo	Pisos/Pilares/Paredes	Calculando ponto médio entre todas as geometrias	Comparar $R_{torre}$ com o mínimo	bool
				Separando geometria das edificações pelo vetor em relação ao ponto médio		
<b>Classificação</b>				Identificar vértices e calcular distância entre eles		
III	Valor	Usuário	Indicar frente do terreno	Menor distância entre vértices de duas edificações ( $R_{torre}$ )		

<b>Categoria</b>	<b>N</b>	<b>Autor</b>	<b>Nome</b>	<b>Equação</b>	<b>Parâmetros</b>	
Conformidade	N6	Lei de uso e Ocupação de Fortaleza; Especialistas	Gabarito	$H < H_{lim}$	$H$ = Altura da Edificação; $H_{lim}$ = Limite de altura	
<b>Descrição</b> É o limite máximo estabelecido pela legislação urbanística para a altura das edificações de uma determinada área						
<b>Etapa</b>	<b>Entrada</b>			<b>Processo</b>	<b>Saída</b>	
	<i>Tipo</i>	<i>Fonte</i>	<i>Entrada</i>	<i>Etapas</i>	<i>Resultado</i>	<i>Und</i>
Algoritmo	Categoria	Elementos do Modelo	Pisos/Pilares/Paredes/Níveis	Identificando Nível do Térreo	Compara $H$ calculado com $H_{lim}$	bool
				Coletando Geometria dos Elementos acima do térreo		
<b>Classificação</b>				Criando envoltória geométrica, BoundingBox baseado na geometria coletada		
III	Valor	Usuário	Limite de Altura	Calculando altura da BoundingBox ( $H$ )		

<b>Categoria</b>	<b>N</b>	<b>Autor</b>	<b>Nome</b>	<b>Equação</b>	<b>Parâmetros</b>		
Conformidade	N7	Lei de uso e Ocupação de Fortaleza; Especialistas	Número de vagas	$N \geq N_{lim}$	N = Número de vagas; Nlim = Limite mínimo de vagas		
<b>Descrição</b>		Quantidade de vagas de estacionamento mínimas para o tipo de edificação					
<b>Etapa</b>	<b>Entrada</b>			<b>Processo</b>	<b>Saída</b>		
	<i>Tipo</i>	<i>Fonte</i>	<i>Entrada</i>	<i>Etapas</i>	<i>Resultado</i>	<i>Und</i>	
Algoritmo	Categoria	Elementos do Modelo	Estacionamento	Identificando elementos de Estacionamento	Compara N calculado com Nlim	bool	
<b>Classificação</b>				III			Valor

<b>Categoria</b>	<b>N</b>	<b>Autor</b>	<b>Nome</b>	<b>Equação</b>	<b>Parâmetros</b>	
Conformidade	N8	Lei de uso e Ocupação de Fortaleza; Especialistas	Tamanho de vagas	$H \geq H_{lim}; L \geq L_{lim}; C \geq C_{lim}$	H = Altura livre da vaga; Hlim = Limite mínimo de altura; L = Largura da vaga; Llim = Largura mínima admitida; C = Comprimento da vaga; Clim = Comprimento limite	
<b>Descrição</b>		Largura, comprimento e altura necessários para as vagas de estacionamento				
<b>Etapa</b>	<b>Entrada</b>			<b>Processo</b>	<b>Saída</b>	
	<i>Tipo</i>	<i>Fonte</i>	<i>Entrada</i>	<i>Etapas</i>	<i>Resultado</i>	<i>Und</i>
Algoritmo	Categoria	Elementos do Modelo	Estacionamento/Nível	Coletando geometria dos elementos de estacionamento	Compara L e C com o limite informado e aponta vagas sem altura livre necessária	bool
				Coletando largura (L) e comprimento (C) para cada elemento de estacionamento		
				Criando envoltória geométrica das vagas, extrudando até a altura mínima		
<b>Classificação</b>	Valor	Usuário	Limite de Altura, Comprimento e Largura	Coletando Geometria dos elementos do nível das vagas e do nível acima		
III				Verificando intersecção dos elementos com as vagas para checar a altura livre		

<b>Categoria</b>	<b>N</b>	<b>Autor</b>	<b>Nome</b>	<b>Equação</b>	<b>Parâmetros</b>	
Conformidade	N9	NBR 15.575; NT N° 005; Especialistas	Rota de Fuga/ Saída de emergência	$L < L_{max}$	L = Distância do ponto mais distante a saída de emergência; $L_{max}$ = Distância máxima admitida	
<b>Descrição</b>					Identificar a rota de fuga em um momento de emergência, verificar se está adequada a edificação. Atendimento a norma de desempenho	
<b>Etapas</b>	<b>Entrada</b>			<b>Processo</b>	<b>Saída</b>	
	<i>Tipo</i>	<i>Fonte</i>	<i>Entrada</i>	<i>Etapas</i>	<i>Resultado</i>	<i>Und</i>
Algoritmo	Categoria	Elementos do Modelo	Paredes /Ambientes	Coleta geometria de Paredes e aberturas	Compara-se L com a maior distância Admissível	bool
				Identifica-se pontos extremos de todos os ambientes		
<b>Classificação</b>	Valor	Usuário	Maior distância admissível ( $L_{max}$ )	A geometria é utilizada para gerar todas as rotas possíveis entre os pontos nos ambientes e o a saída.		
V	Escolha	Usuário	Identificar elemento que representa a saída	Identifica-se a maior distância entre todas as rotas (L)		

<b>Categoria</b>	<b>N</b>	<b>Autor</b>	<b>Nome</b>	<b>Equação</b>	<b>Parâmetros</b>	
Conformidade	N10	NBR 15.575; Especialistas	Pé direito	$H > H_{min}$	H = Altura livre do ambiente; $H_{min}$ = Altura mínima necessária	
<b>Descrição</b>					Altura livre necessária em ambientes da edificação	
<b>Etapas</b>	<b>Entrada</b>			<b>Processo</b>	<b>saída</b>	
	<i>Tipo</i>	<i>Fonte</i>	<i>Entrada</i>	<i>Etapas</i>	<i>Resultado</i>	<i>Und</i>
Algoritmo	Categoria	Elementos do Modelo	Ambientes / Níveis / Pisos	Gera um sólido que representa o volume livre da edificação, a partir dos ambientes e dos limites de altura	Compara-se H com a $H_{min}$	bool
<b>Classificação</b>				Coleta toda a geometria do nível analisado e do superior a este		
III	Valor	Usuário	Limites de Altura por ambiente ( $H_{min}$ )	Realiza a intersecção geométrica entre os volumes e a geometria dos elementos		

<b>Categoria</b>	<b>N</b>	<b>Autor</b>	<b>Nome</b>	<b>Equação</b>	<b>Parâmetros</b>	
Conformidade	N11	NT N° 001/2008; Especialistas	Separação entre edificações	$L_{edf} > L_{min}$	L <sub>edf</sub> = Distância entre duas edificações; L <sub>min</sub> = Distância mínima admissível	
<b>Descrição</b> Identificar a distancia entre as edificações vizinhas é maior que a mínima						
<b>Etapa</b>	<b>Entrada</b>			<b>Processo</b>	<b>saída</b>	
	<i>Tipo</i>	<i>Fonte</i>	<i>Entrada</i>	<i>Etapas</i>	<i>Resultado</i>	<i>Und</i>
Algoritmo	Categoria	Elementos do Modelo	Pisos	Coletar geometria dos pisos e calcular ponto médio entre todos os pisos	Compara-se H com a H <sub>min</sub>	bool
<b>Classificação</b>				Separa pisos pelo o ângulo do vetor do centro do piso ao ponto médio, formando grupos de pisos das edificações		
III	Valor	Usuário	Limites de distância entre edificações (L <sub>min</sub> )	Calcular distância entre os vértices de todos os pisos		

<b>Categoria</b>	<b>N</b>	<b>Autor</b>	<b>Nome</b>	<b>Equação</b>	<b>Parâmetros</b>	
Conformidade	N12	NBR 9050; Especialistas	Inclinação de Rampas	$I < I_{max}$	I = inclinação da rampa; I <sub>max</sub> = inclinação máxima permitida	
<b>Descrição</b> As características de rampas de acesso destinadas a cadeirantes garantem a funcionalidade destas, verificar principalmente inclinação da rampa						
<b>Etapa</b>	<b>Entrada</b>			<b>Processo</b>	<b>saída</b>	
	<i>Tipo</i>	<i>Fonte</i>	<i>Entrada</i>	<i>Etapas</i>	<i>Resultado</i>	<i>Und</i>
Algoritmo	Categoria	Elementos do Modelo	Rampas	Coletar inclinação das rampas	Comparar Inclinação com o limite	bool
<b>Classificação</b>						
III	Valor	Usuário	Limite de inclinação	Verificando se é antiderrapante		

<b>Categoria</b>	<b>N</b>	<b>Autor</b>	<b>Nome</b>	<b>Equação</b>	<b>Parâmetros</b>	
Conformidade	N13	NBR 9050; Especialistas	Vagas Acessíveis	$V_{ace} > V_{min}$	I = inclinação da rampa; I <sub>max</sub> = inclinação máxima permitida	
<b>Descrição</b> Reserva de vagas para veículos que transportam pessoa com deficiência com comprometimento de mobilidade						
<b>Etapa</b>	<b>Entrada</b>			<b>Processo</b>	<b>Saída</b>	
	<i>Tipo</i>	<i>Fonte</i>	<i>Entrada</i>	<i>Etapas</i>	<i>Resultado</i>	<i>Und</i>
Algoritmo	Categoria	Elementos do Modelo	Vagas	Identificar Vagas no modelo	Comparar Número de Vagas acessíveis com o limite	bool
<b>Classificação</b>				Escolha		
III	Valor	Usuário	Número mínimo de vagas	Quantificar vagas acessíveis		

<b>Categoria</b>	<b>N</b>	<b>Autor</b>	<b>Nome</b>	<b>Equação</b>	<b>Parâmetros</b>	
Conformidade	N14	NBR 9050; Especialistas	Dimensão mínima de portas	$P_l > P_{lmin}$	I = inclinação da rampa; I <sub>max</sub> = inclinação máxima permitida	
<b>Descrição</b>						
Verificar dimensões mínimas de aberturas destinadas a ambientes e edificações acessíveis						
<b>Etapa</b>	<b>Entrada</b>			<b>Processo</b>	<b>saída</b>	
	<i>Tipo</i>	<i>Fonte</i>	<i>Entrada</i>	<i>Etapas</i>	<i>Resultado</i>	<i>Und</i>
Algoritmo	Categoria	Elementos do Modelo	Portas	Identificar Portas no modelo	Comparar largura com limite	bool
<b>Classificação</b>						
III	Valor	Usuário	Largura mínima de portas	Calcular largura das portas		