

DESENVOLVIMENTO DE ADD-IN PARA CONTROLE DA INFORMAÇÃO EM MODELOS BIM

Add-in development to control information on BIM models

Tiago Pinto de Cerqueira¹, Gustavo Martins da Silva Gusmão², Maria Carolina Gomes de Oliveira Brandstetter³



PALAVRAS CHAVE:

BIM;
Nomenclatura BIM;
Add-in para Revit®

KEYWORDS:

BIM;
BIM Nomenclature;
Add-in for Revit®

RESUMO: O uso do BIM tem crescido e consolidado no meio da Construção Civil e muitos países já têm exigido seu uso para conceder licitações em obras públicas. O crescimento do BIM vem acompanhado de tentativas de padronizações por parte de organizações internacionais, órgãos públicos e universidades. A padronização dos projetos tem ajudado a tornar os mesmos acessíveis a quem o leia, independentemente da localização geográfica ou disciplina (por exemplo: ar condicionado, estrutura, eletricidade, entre outros). Entre os vários itens a serem padronizados, buscou-se desenvolver uma ferramenta que permitisse a padronização da nomenclatura, que é o único parâmetro comum e obrigatório a todos os modelos BIM. O objetivo do trabalho é apresentar um processo de modelagem paramétrica por meio da solução de um Add-In para o software Revit®, capaz de controlar este fluxo de informações. Para atingir esse objetivo, utilizou-se programação e banco de dados em conjunto com a plataforma BIM da Autodesk, o Revit®. Dentro da ferramenta desenvolvida, o nome foi dividido em partes menores que possuem um valor definido de acordo com seus parâmetros, tornando o nome dos elementos também a sua identidade. Essa padronização tornará os modelos não só legíveis para os usuários como também a outros softwares, por meio da vinculação com as várias ferramentas do mercado da construção, como ERPs e ferramentas de planejamento.

ABSTRACT: *The use of BIM has grown and consolidated in the AEC industry. Many countries have already required its use for public work bids. The BIM growth has been accompanied by attempts at standardization by international organizations, governmental entities and universities. The projects standardization has helped to make them accessible to anyone who reads them, regardless of geographic location or discipline (HVAC, structural, electrical among others). Among the various items to be standardized, in this work a tool was developed to allow the standardization of the nomenclature, which is the unique common and mandatory parameter for all BIM models. The objective of the work is to present a parametric modeling process through the solution of an Add-In for the Revit® software, capable of controlling this flow of information. To achieve this objective, programming and database were used in conjunction with Autodesk's BIM platform, Revit®. Within the developed tool, the name was divided into smaller parts that have a defined value according to their parameters, making the name of the elements also their identity. This standardization will make the models not only readable for users but also for other software, through links with various tools in the construction market, such as ERPs and planning tools.*

* Contato com os autores:

Publicado em 29 de dezembro de 2023

¹ e-mail: cerqueira.tiago15@gmail.com (T. P. de Cerqueira)

Engenheiro Civil pela Escola de Engenharia Civil da Universidade Federal de Goiás (EECA/UFG)

² e-mail: gustavogusmao18@gmail.com (G. M. da S. Gusmão)

Engenheiro Civil pela Escola de Engenharia Civil da Universidade Federal de Goiás (EECA/UFG)

³ e-mail: maria_carolina_brandstetter@ufg.br (M. C. G. O. Brandstetter)

Engenheira Civil, Doutora, Docente do Curso de Engenharia Civil e Ambiental (EECA) e do curso de Pós-Graduação em Geotecnia, Estruturas e Construção Civil da Escola de Engenharia Civil da Universidade Federal de Goiás (PPG-GECON/UFG)

1. INTRODUÇÃO

A Modelagem da Informação na Construção, em inglês *Building Information Modeling*, é um conjunto de processos, ferramentas e tecnologias capaz de produzir, utilizar e atualizar um modelo virtual de informações representativo da edificação contemplando todo o ciclo de vida da mesma e servindo a diferentes finalidades (PENTTILA, 2006; SUCCAR, 2009; SANTOS, 2012; SHIROWZHANA *et. al*, 2020). Com isso, a Modelagem da Informação na Construção Civil pode ser apontada como uma expressão atual que melhor sintetiza as inovações inerentes a esta indústria (SUCCAR e KASSEM, 2015).

Como esse modelo virtual possui diferentes finalidades, em um ambiente controlado como o ambiente virtual, pode-se simular cenários que ocorrerão no processo de construção da edificação, antecipando a detecção de incoerências e, como afirma Eastman *et al.* (2014), que quanto antes detecta-se esses erros, seja dos processos ou projetos, menor será o impacto no custo do empreendimento e maior será a capacidade de influenciar no seu custo em todo o ciclo de vida da edificação. Essa relação é ilustrada na Figura 1.

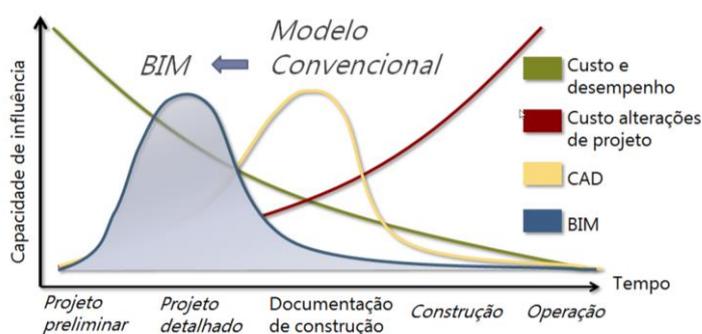


FIGURA 1: Curva de MacLeamy

FONTE: Adaptado de CURT (2004)

Pelo gráfico da Figura 1 pode-se analisar que, com a utilização de processos BIM na fase de projeto detalhado, tem-se um alto impacto na edificação, antes mesmo do processo de documentação, o que implica em mais tempo para poder solicitar alterações nas concepções do projeto, amplificando assim o poder de tomada de decisões sobre processos construtivos, materiais, soluções de engenharia, etc. Somado a isso pode-se agregar também a facilidade de alterações de soluções construtivas representadas nos *softwares* BIM, como por exemplo o *Revit*[®], que é uma ferramenta de modelagem paramétrica 3D.

Segundo Eastman *et al.* (2014), a ideia básica de modelagem paramétrica é que as instâncias de forma e propriedades podem ser definidas e controladas de acordo com uma hierarquia de parâmetros nos níveis de conjunto e subconjunto, assim como no nível de objeto individual. Alguns dos parâmetros dependem de valores definidos pelo usuário; outros dependem de valores fixos e outros são obtidos de outras formas ou são relativos a elas. As formas podem ser 2D ou 3D.

No projeto paramétrico o projetista pode definir as condições as quais a edificação será exposta e parametrizá-la de forma que o modelo virtual comporte às variações de parâmetros, sejam estes relacionados à forma ou características próprias da instância do objeto e devido à interrelação, existente ou não, entre os parâmetros. Sendo assim, o modelo virtual expressará diferentes condições, formas e dados, que auxiliam os envolvidos no empreendimento a tomar melhores decisões em curto prazo.

Em relação às ferramentas de modelagem da informação na construção, segundo Eastman *et al.* (2014), são modelos paramétricos baseados em objetos com um conjunto predefinido de famílias; cada uma tem um comportamento programado, e podem ser customizadas pelo usuário ou já programadas pelo fornecedor do *software* ou ainda pelo fabricante do objeto.

A ferramenta mais usada no mercado, segundo Eastman *et al.* (2014), é o Revit® compartilhada em versão estudantil com completo acesso a quase todos os recursos da mesma, sendo a única restrição a de comercialização do produto gerado.

O software Revit® possui uma estrutura de elementos baseada em objetos que são próprios do sistema, que são caracterizados como Famílias de Sistema, e outra classe de objetos que são caracterizados como Famílias carregáveis, Famílias locais. Uma família é um grupo de elementos com um conjunto comum de parâmetros (propriedades) e uma representação gráfica relacionada (Central de Ajuda Revit®, 2018).

Os diferentes elementos pertencentes a uma família podem ter diferentes valores para alguns ou todos os parâmetros, mas o conjunto de parâmetros (seus nomes e significados) é o mesmo. Essas variações dentro da família são denominadas de tipos de família ou tipos (Central de Ajuda Autodesk Revit®, 2018).

Quando um elemento é criado em um projeto com uma família e tipo de família específicos, é possível criar uma instância do elemento. Cada instância do elemento tem um conjunto de propriedades, no qual é possível alterar alguns parâmetros do elemento independente aos parâmetros de tipo de família. Estas alterações se aplicam somente à instância do elemento, o elemento único no projeto. Se quaisquer alterações para os parâmetros de tipo de família forem efetuadas, as alterações aplicam-se a todas as instâncias do elemento que foram criadas com aquele tipo (Central de Ajuda Autodesk Revit®, 2018).

Definidos esses conceitos sobre a ferramenta e a modelagem paramétrica, é importante destacar que a produção de um modelo para certos tipos de análises, seja encontrar interferências, análises financeiras, de planejamento e até mesmo análises dimensionais e desempenho, pode não atender todos os tipos de modelagem ao mesmo tempo. Segundo Eastman *et al.* (2014), nenhuma aplicação pode cumprir sozinha todas as tarefas associadas ao projeto e à produção de uma construção. Sendo assim, surge o conceito de Interoperabilidade, que representa a necessidade de passar dados entre as aplicações, permitindo que múltiplos tipos de especialistas e aplicações contribuam para o trabalho em questão.

Um dos principais modelos de dados de produtos de construção é o *Industry Foundation Class* (IFC). Para Eastman *et al.* (2014) o IFC foi desenvolvido a fim de criar um grande conjunto de dados consistentes de informação da construção para intercâmbio entre as aplicações relacionadas às indústrias de Engenharias, Arquitetura e Construção (AEC).

Com base nesses conceitos definidos, o presente trabalho justifica-se na proposta do desenvolvimento de um processo para o controle de dados e fará o uso de programação Orientada a Objetos (através da linguagem C#) para o desenvolvimento de um programa para controle de informação relacionada a nomenclatura dos objetos em um modelo paramétrico.

Com o crescente uso de BIM no Brasil nos últimos anos a fim de tentar atingir seus benefícios finais, sendo um dos mais importante, o positivo impacto financeiro no empreendimento, é importante destacar a importância da organização dos dados que serão inseridos na produção de um modelo virtual representativo da edificação.

Nenhuma aplicação pode cumprir sozinha todas as tarefas associadas ao projeto e à produção de uma construção, sendo assim os dados presentes no modelo precisam ter condições de serem manipulados por outras aplicações a fim de atingir os objetivos para o qual o modelo foi adquirido.

Segundo Catelani e Santos (2016), mesmo a realização de atividades simples como, por exemplo de dar nome aos componentes, funções e processos, pode gerar mal-entendidos e retrabalhos. Sendo assim é necessária uma padronização que facilite o trabalho colaborativo já que não estamos em um mundo ideal em que sempre será possível aproveitar o esforço já feito, partindo de um mesmo ponto.

Sendo assim, este trabalho propõe o desenvolvimento de um *software* que auxiliará o processo de padronização de nomenclaturas dos objetos dentro de um modelo paramétrico facilitando a gestão de informações do mesmo. O *software* é um *Add-in*¹ que atuará no *Revit*® 2018 facilitando a gestão de dados referentes à identificação dos elementos em um modelo.

2. OBJETIVO

Apresentar um processo de modelagem paramétrica, por meio da solução de um *Add-in*¹ para o *software Revit*® 2018, para o controle do fluxo de informações referentes à identificação dos componentes do modelo paramétrico seguindo uma padronização customizada pré-definida.

3. REVISÃO

3.1 QUALIDADE DE PROJETO

Existem algumas definições para o conceito de qualidade de projeto, porém entre todas as definições, o foco dos conceitos engloba o cliente e/ou consumidor final e suas necessidades. Sendo assim para esse trabalho define-se como cliente as empresas de orçamentação e departamentos de controle de obras de construtoras as quais fariam utilização do modelo parametrizado finalizado para análises relacionadas a orçamentação. E o público-alvo deste trabalho são empresas que utilizam dos processos da Modelagem da Informação na Construção Civil para a construção de modelos paramétricos cujo objetivo final é a utilização em orçamentos e controle de gastos em obra.

Segundo Novaes (2001) o projeto possui dois conceitos distintos, sendo um estático, referente a projeto como produto, constituído por elementos gráficos e descritivos, ordenados e elaborados de acordo com linguagem apropriada, destinado a atender às necessidades da etapa de produção. E outro, dinâmico, que confere ao projeto um sentido de processo, através do qual as soluções são elaboradas e necessitam ser compatibilizadas.

Instrumentos destinados a controles da qualidade devem incluir controles de produção e controles de recebimento. Os primeiros, realizados por meio de autocontrole interno, acrescidos de controle externo independente; e os segundos, realizados por meio de controles exercidos pelo cliente contratante, tendo por finalidade a verificação da conformidade dos resultados do processo às exigências previamente estabelecidas (NOVAES, 2001).

Sendo assim, entende-se que no âmbito interno, grande parte do controle seria feito pelo próprio projetista, ao respeitar os parâmetros intrínsecos à própria disciplina de seu projeto específico e os dados contidos nas informações transmitidas pelos demais participantes de processo de projeto. Adicionalmente, este controle deve ser exercido no âmbito da coordenação de projetos, no cumprimento de suas atribuições (MESEGUER, 1991; MELHADO *et. al*, 2006).

Para tanto, são importantes os instrumentos da garantia e controle da qualidade de projetos de edifícios: qualificação de profissionais de projeto e de novos projetos; coordenação e análise crítica de projetos; elaboração de projetos para produção; controle da qualidade de projetos; controle de modificações durante a produção; elaboração de projetos com emprego de recursos computacionais; e parâmetros de projeto relacionados com o tempo (PICCHI, 1993; SACKS *et. al*, 2018).

Como instrumentos de controle, podem ser citados o controle de interfaces, compatibilização de projetos, controle dos dados de entrada e controle de revisões e pendências (PICCHI, 1993; SUCCAR, 2009; SACKS *et. al*, 2018). O processo proposto por este trabalho englobará cada um destes itens e dando

¹ *Add-in*: Programa computacional que executa funções adicionais às funções nativas do *software Revit*®.

um destaque maior para o desenvolvimento de uma ferramenta para controle dos dados de entrada para a identificação dos componentes de um modelo paramétrico.

3.2 BIM

3.2.1 Definições de BIM

Para o nome BIM existem duas definições diferentes, sendo uma se referindo apenas ao modelo e outra se referindo ao processo de modelagem. Segundo o AIA (2008), o *Building Information Model* é o modelo digital do edifício, que representa suas características geométricas, a relação entre seus componentes e variados parâmetros relevantes para os envolvidos na elaboração e leitura do modelo. A segunda definição é o *Building Information Modeling*, que significa todo o processo e tecnologia utilizado na criação do modelo, sendo essa a mais usual ao se referir aos processos e não ao produto gerado. Para esse trabalho será utilizada a segunda definição já que neste trabalho será desenvolvido um processo de modelagem para obtenção de informações utilizadas em análise financeira.

Para permitir uma implementação integrada, a simulação pelo *Building Information Modeling* deve exibir seis características principais:

- Digital;
- Espacial (3D);
- Mensurável (quantificável, dimensionável, consultável);
- Abrangente (incorporando e comunicando as interações do projeto, o desempenho da construção, a construtibilidade e incluir aspectos financeiros e sequenciais de meios métodos);
- Acessível (a toda a equipe do empreendimento e ao proprietário por meio de uma interface interoperável e intuitiva);
- Durável (utilizável ao longo de todas as fases da vida de uma edificação).

Segundo Eastmann *et al.* (2014) e Sacks *et al.* (2018), ainda não há processos, *softwares* e *hardwares* capazes de suportar toda a tecnologia BIM, e acredita-se que isso mudará ao longo dos próximos anos. Porém o que ainda se tem atualmente são partes de um processo que auxiliam bem em seus objetivos.

É importante tanto a definição do que é BIM quanto mostrar o que não é BIM a fim de entender ao longo dos processos os rumos aos quais o modelo deverá seguir. Sendo assim, para Eastmann *et al.* (2014), existem 4 tipos de modelos:

- Modelos que só contém dados 3D, sem atributos (parâmetros) relacionados ao objeto;
- Modelos sem suporte para comportamentos por meio de alterações paramétricas;
- Modelos que possuem múltiplas referências a arquivos CAD 2D;
- Modelos que ao sofrer modificações relacionadas às dimensões em uma vista de documentação, não atualizam em outras automaticamente.

3.2.2 Uso do Revit® para Modelagem da Informação

Segundo Eastman *et al.* (2014) e Sacks *et al.* (2018) o Revit® é a ferramenta mais conhecida e atualmente líder de mercado para o uso do BIM em projetos de arquitetura, instalações e estruturas. Ele foi introduzido pela Autodesk em 2002, depois da aquisição do programa de uma empresa iniciante. O Revit® é uma plataforma completamente separada do AutoCAD®, com código base e estrutura de arquivos diferente.

O modelo necessita ser parametrizado, ou seja, os objetos necessitam possuir características em relação a comportamento, geometria, dados referentes as suas composições de materiais e dados

intrínsecos ao processo de construção do empreendimento e de seus métodos construtivos. Em relação a esse critério o *software Revit®* possui elementos nativos, como paredes, pisos, escadas, tubulações etc, dos quais também possui parâmetros básicos nativos que indicam seus materiais, como visualizado na Figura 2, que é totalmente possível extrair quantitativos através de comandos próprios do *software*, ilustrado na Figura 3, além de indicar dados referentes a sua geometria.

Os elementos do *Revit®* são categorizados segundo suas funções mais comuns na construção civil, seja essa função apenas conceitual como uma Grade que está na categoria de Grades, ou até mesmo elementos mais concretos como paredes que possuem a categoria que leva o mesmo nome.

Todos os elementos do *Revit®* estão associados a uma família. Segundo a Central de Ajuda do Autodesk *Revit®* (2018), uma família é um grupo de elementos com um conjunto comum de propriedades chamado de parâmetros e uma representação gráfica relacionada.

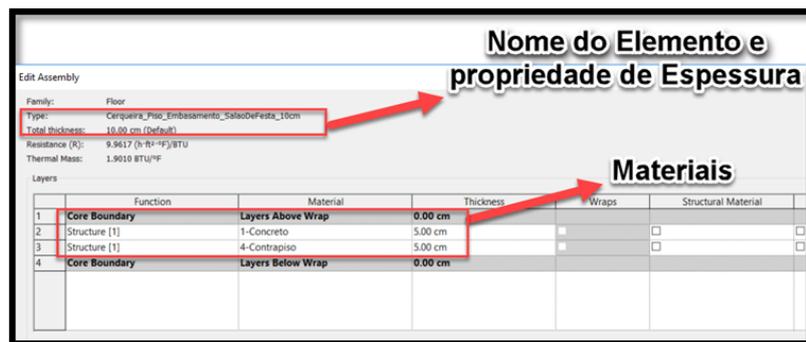


FIGURA 2: Propriedade de tipo de um piso no *software Revit®*

FONTE: Autoria própria

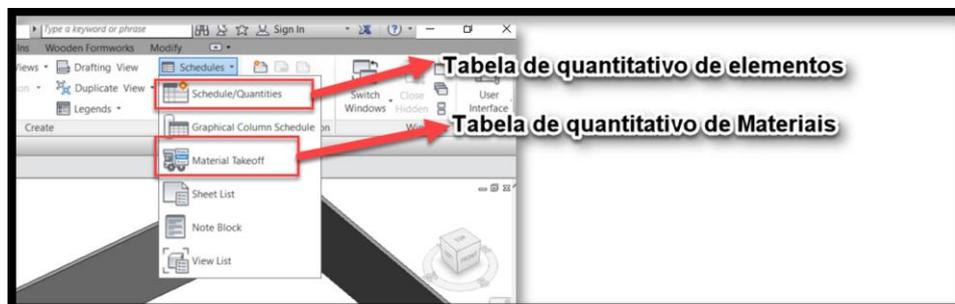


FIGURA 3: Comandos de criação de tabela de quantitativos do *Software Revit®*

FONTE: Autoria própria

Uma poderosa ferramenta do *software* é a liberdade de criação de parâmetros para um específico projeto, onde se vê os *Project Parameters*, para o português, Parâmetros de Projeto. O mesmo pode até ser compartilhado a outros projetos através de um arquivo de formato “.txt”, esses são chamados de *Shared Parameter*, ou em português, Parâmetro Compartilhado.

Segundo a Central de Ajuda do Autodesk *Revit®* (2018), os Parâmetros de Projeto são armazenadores de informações que podem ser definidas e, a seguir, adiciona-se em múltiplas categorias de elementos em um projeto, como visualizado na Figura 4.

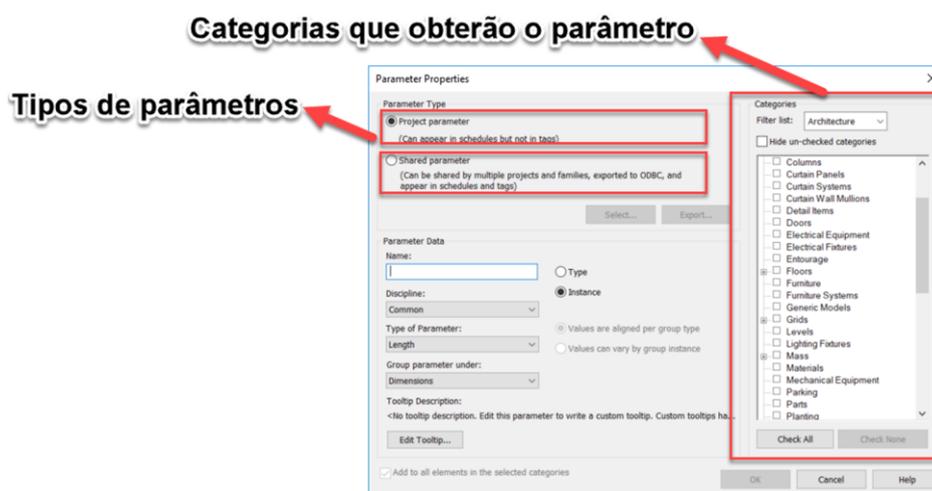


FIGURA 4: Criação de parâmetros do projeto do *software Revit®*
FONTES: Autoria própria

Os Parâmetros Compartilhados, segundo a Central de Ajuda do Autodesk *Revit®* (2018), são definições de parâmetros que podem ser adicionadas nos projetos. As definições dos parâmetros compartilhados são armazenadas em um arquivo “.txt”, independentemente de qualquer arquivo do projeto do *Revit®*; e isso permite acessar o arquivo de diferentes famílias ou projetos. O parâmetro compartilhado é uma definição de um contêiner para as informações que podem ser utilizados em múltiplas famílias e projetos.

As famílias podem ser classificadas como de Sistema, Carregáveis e locais. Cada família possui um conjunto de tipos que são elementos que, ao possuir a mesma parametrização, se diferem em relação aos valores dos parâmetros relacionados.

Ainda segundo a Central de Ajuda do Autodesk *Revit®* (2018), todos os elementos que estão em um projeto são instâncias de tipos de famílias e essas instâncias possuem dois conjuntos de propriedades ou parametrizações que controlam seu comportamento e aparência, que são as propriedades comuns ao tipo e as propriedades relacionadas à instância.

Sendo tipos e famílias elementos que possuem propriedades em comum e considerando que cada elemento dentro de um modelo é uma instância que pertence a um tipo de uma família, é necessário que ele seja bem classificado e, além disso, bem identificados para que suas propriedades sejam extraídas com mais agilidade e confiabilidade.

Sendo assim, com a possibilidade de inserir informações e extraí-las associadas aos objetos aos quais elas pertencem, faz com que o *Revit®* seja uma ferramenta poderosa para o desenvolvimento de processos BIM. Porém um cuidado importante é a identificação correta dos elementos para que as informações sejam encontradas com mais facilidade, precisão, sem duplicidades e sem ambiguidades.

Com a padronização da identificação dos elementos e o controle automatizado, o processo desenvolvido nesse trabalho referente à correta nomenclatura dos objetos em um modelo paramétrico facilitará o controle de informações em um modelo paramétrico.

3.2.3 Controle da Informação

Segundo Meseguer (1991) é necessário, para manter a qualidade de um projeto, o autocontrole do profissional responsável pelo projeto. Nos processos de modelagem, o projetista trabalhará com dezenas de informações ao mesmo tempo associadas a representações 3D dos objetos, acrescentando mais itens aos quais ele necessitará ter controle. Sendo assim, sem uma padronização definida para o controle das informações, a qualidade do projeto ficará comprometida.

O *software Revit*® não possui ferramentas próprias para o controle das informações que o modelo receberá do projetista, e com isso o projeto e o fim ao qual ele se destina ficam comprometidos em qualidade e veracidade respectivamente. Oportunamente, a Autodesk disponibiliza aos usuários documentações para desenvolvimento de *softwares* que podem manipular, criar e controlar as informações dentro de um modelo, são os chamados plugins ou *Add-in*.

3.3 PADRONIZAÇÃO DA NOMENCLATURA

Segundo Chen *et al.* (2017), a nomenclatura de um objeto dentro do modelo é o primeiro parâmetro e informação do mesmo e é frequentemente usado para fazer a conexão com informações com as informações que realmente são relevantes do objeto como o comprimento de uma parede de bloco cerâmico de 9cm.

Ainda segundo Chen *et al.* (2017), ao redor do mundo, vários gerentes de projeto e instituições tentaram fazer convenções e padronizações de nomenclatura para objetos BIM, porém o trabalho manual para a implantação e manutenção dos mesmos fizeram com que esse processo se tornasse utópico.

Segundo Catelani e Santos (2016), a norma Brasileira de BIM trará alguns padrões e códigos para a melhor identificação dos elementos dentro de um modelo BIM, independente de qual plataforma esteja usando a fim de facilitar a comunicação entre projetistas e seus projetos. E com um processo automatizado, a implantação da mesma ficará muito mais fácil.

Segundo Chen *et al.* (2017), a automatização é uma necessidade para esse processo afim de garantir consistência e falta de redundância em relação a informação em um projeto. Sendo assim o trabalho apresentará um *Add-in* capaz de controlar a correta identificação dos componentes de um modelo paramétrico a fim de possibilitar uma parametrização da nomenclatura associando dados fornecidos pelo contratante do modelo que serão importados para um banco de dados e os próprios parâmetros do objeto que uma vez atualizados repercutirá em uma atualização da identificação do elemento ao qual os mesmos pertencem.

4. METODOLOGIA

Neste item será caracterizada a metodologia utilizada no desenvolvimento do *Add-in* para *Revit*® destinado à definição e controle de nomenclaturas dentro de modelos BIM. Para isso, serão analisados os recursos envolvidos para o desenvolvimento, os processos envolvidos no desenvolvimento, as melhorias implementadas durante o desenvolvimento e finalmente os resultados esperados com sua utilização.

4.1 ANÁLISE DO CENÁRIO EM QUE O *ADD-IN* SERÁ INSERIDO

Para analisar o cenário em que o *Add-in* será inserido, é necessário entender melhor o modo como um *software Revit*® trabalha com os elementos de seu modelo, mais especificamente, como o *software* lida com a identidade dos elementos dentro de um modelo.

Dentro de um modelo *Revit*® os elementos não são (ou não deveriam ser) meras geometrias 3D que podem ser vistas de diferentes ângulos. O que enquadra um modelo *Revit*® no âmbito BIM são as informações que estão vinculadas aos seus elementos. Especificamente para o *Revit*®, essa informação é armazenada em parâmetros.

Os parâmetros podem ser enquadrados em duas possíveis naturezas. A primeira delas está ligado aos parâmetros que garantem a identidade e a outra ligada aos parâmetros que permitem a quantificação do elemento (ou conjunto de elementos).

Os parâmetros identificadores, como o próprio nome sugere, busca trazer uma identidade ao elemento de forma a transformar um conjunto de parâmetros em sua identidade. Esses parâmetros estão

relacionados a geometrias (altura e largura como exemplo), propriedades físico-químicas (peso específico, pressão, voltagem entre outros) e informações não físicas (tempo para ser construído, fabricante e vencimento da garantia). A combinação destes parâmetros irá garantir a identidade do elemento dentro do modelo como exemplificado para o caso de paredes a seguir:

- Alvenaria em bloco cerâmico de 14cm;
- Alvenaria em bloco cerâmico de 9cm;
- Alvenaria em bloco de concreto de 14cm.

É possível observar a existência de três parâmetros indiretamente ligados as paredes em questão: “tipo”, “material” e “espessura”. Uma rápida análise evidencia o mesmo valor de “tipo” para as três paredes (Alvenaria), propriedade de “material” semelhante entre as duas primeiras (Bloco Cerâmico) mas diferente na última (Bloco de concreto) e finalmente “espessura” semelhante entre a primeira e última (14 cm) mas diferente para a segunda (9cm).

Conclui-se então que o nome não deve ser entendido como um conjunto de caracteres alfanuméricos, mas sim como um elemento nome formado por partes menores, neste caso, “tipo + material + espessura”. Diferentes elementos podem ter um, dois ou mesmos as três partes citadas anteriormente alteradas, gerando assim, novas combinações de valores que irão gerar um novo elemento com uma identidade única no modelo.

Considerando um elemento já devidamente identificado, os parâmetros de natureza quantitativa, aqui identificados como parâmetros quantitativos, estarão disponíveis para a obtenção de valores quantitativos para orçamentação ou planejamento por exemplo. Estes podem contemplar volume, área, comprimento linear, quantidade unitária, entre outros. Através destes parâmetros, é possível obter, para as paredes exemplificadas acima, informações como:

- 5 m² de alvenaria em bloco cerâmico de 9 cm;
- 10 metros lineares de alvenaria em bloco de concreto de 14 cm.

Informações como a exemplificada podem alimentar tabelas e outros *softwares* ligados às várias fases de uma construção. Porém, esses valores somente poderão ser encontrados e utilizados caso estejam corretamente identificados.

4.2 IDENTIFICAÇÃO DA ÁREA DE ATUAÇÃO DO ADD-IN

Os parâmetros identificadores de um elemento podem ou não estar presentes no nome de um elemento. Considerando o exemplo de parede visto no tópico anterior, a permeabilidade da parede pode ou não ser algo necessário no nome da família caso o uso do modelo seja orçamentação por exemplo. Porém, ele pode ser necessário caso o modelo seja utilizado para uma análise hidráulica em um reservatório.

Conclui-se então que o nome dos elementos deve conter os parâmetros identificadores que o tornam tão único quanto o necessário para seu uso, estando este aspecto diretamente vinculado ao plano de execução BIM dos interessados no projeto. Isso se deve ao fato de que o *software* irá aceitar dois elementos com os parâmetros identificadores semelhantes, porém não irá aceitar elementos diferentes com o mesmo nome.

Independente da estratégia desenvolvida para a execução do modelo BIM, uma nomenclatura correta dos elementos dentro do *Revit*® pode ser usada como uma metodologia para o mapeamento das informações e logo, é necessário ser cuidadosamente planejamento e controlado. Deve se levar em conta que o nome pode ser o único parâmetro obrigatório e comum para todas as disciplinas envolvidas de um projeto (estrutura, arquitetura, elétrico entre outras), pois como o parâmetro “voltagem”, que é útil em um modelo elétrico por exemplo, pode não ser necessário em modelo de estrutura.

Dada a importância do nome dos elementos para o modelo BIM, confirma-se que o nome não pode, como já foi dito, ser pensado somente como um conjunto de caracteres, pois uma parede de alvenaria em bloco cerâmico com 9 cm de espessura pode ser nomeada por alguém como “Alvenaria em bloco cerâmico de 9 cm”, outro usuário pode escrever “Alvenaria – Cerâmico – 9 cm” e outro usuário, muitas vezes por um limite máximo de caracteres permitidos, nomear a mesma parede como “Alv. Bl Cerâmico – 9”.

A atuação do *Add-in* foi pensada então como atuante diretamente na criação de padronização da nomenclatura diante dos possíveis usos do modelo. Já a escolha de um *software* para controlar os nomes e não um controle manual, foi influenciada pela possibilidade de se converter os mais variados tipos de informação em objetos menores para comporem posteriormente um nome, retirando do processo a possibilidade de erro por parte de quem está escrevendo esse nome.

Uma vez que a nomenclatura esteja bem definida, encontrar qualquer elemento para ser vinculado a um orçamento, planejamento ou outro fim, se tornará uma tarefa mais simples e menos suscetível a erro, pois os elementos poderão ser filtrados por qualquer uma das propriedades embutidas neste nome independentemente da disciplina envolvida. Essa capacidade de ser filtrada se mostra útil atualmente após a chegada da Indústria 4.0. Nela, a informação virtual precisa ser encontrada, depois analisada/aprimorada e finalmente devolvida à realidade com um maior valor agregado. Se uma informação não pode ser encontrada no meio virtual, todo o restante do ciclo é comprometido.

4.3 ESTRUTURA DO ADD-IN

Para o desenvolvimento do *Add-in* foi utilizada a API (*Application Programming Interface*) disponibilizada pela Autodesk® para o seu *software Revit*. Dentre as linguagens de programação disponíveis para o desenvolvimento (C#, Visual Basic e C++), foi escolhida a linguagem C# pela experiência de um dos autores no desenvolvimento de *Add-in* para o *Revit*® nesta linguagem.

Para armazenar os dados requeridos pelo *Add-in* em desenvolvimento, cogitou-se inicialmente o uso do próprio *Revit*® e sua API. Porém, encontrou-se uma limitação no armazenamento de informações no *Revit*®, não sendo possível a criação customizada requerida para este trabalho. Era necessário um local onde as diretrizes para a nomenclatura pudessem ser armazenadas e a partir deste ponto, os elementos do *Revit*® vinculados à essa diretriz teriam seus nomes editados segunda essa diretriz.

Para armazenar esses dados, fez o uso de um banco de dados relacional, que segundo o site da Amazon AWS, consiste em uma coleção de dados com relacionamentos predefinidos entre si.

A ferramenta escolhida para gerenciar estes dados foi o *SQL Server* da Microsoft, fazendo uso da linguagem T-SQL, também criado pela mesma empresa. Para os dados a serem armazenados pelo *Add-in* foi escolhido o banco de dados *Amazon Relational Database Service* (RDS), um banco de dados online mantido pela Amazon®. O serviço oferecido pela empresa é pago e o custo proporcionalmente ligado a demanda, porém, por se tratar de um trabalho acadêmico, a versão de testes gratuita foi utilizada.

Para realizar as devidas configurações e estruturações do banco de dados, foi utilizado o *software SQL Server Management Studio* (SSMS), desenvolvido também pela Microsoft®.

A escolha das linguagens de programação, banco de dados e locais para armazenamento especificadas acima levou em conta consultas a colegas da área de computação para checagem dos meios mais eficientes para alcançar os objetivos propostos neste trabalho.

4.4 FUNCIONAMENTO DA FERRAMENTA AUXILIAR

Para o funcionamento do *Add-in* proposto, as regras de nomenclatura serão definidas pelo usuário dentro de uma interface própria, contudo, o controle e acompanhamento dessas regras

independem do usuário. Ou seja, o usuário fica responsável de atribuir corretamente as diretrizes para a nomenclatura e através destes, o nome é automaticamente criado por meio de programação.

Pretende-se com essa solução retirar do usuário a responsabilidade de verificação contínua de nomes e focar o usuário nas partes que demandam uma maior planejamento e processos escolhas, além de permitir mais tempo para o processo de modelagem.

Buscou-se durante o desenvolvimento trazer para a nomenclatura a ideia descrita nos tópicos anteriores de que um nome é nada mais do que um conjunto de propriedades (dentro do âmbito da computação, objetos) que, seguindo uma estrutura predefinida, servirão como moldes para a nomenclatura dos elementos.

Dadas os vários elementos que irão compor um modelo dentro da construção civil, como paredes, pisos, portas, janelas, escadas, entre muitos outros, as regras de nomenclatura precisam ser subdivididas para atender essa variedade de elementos. Para isso, utilizou de um sistema em que o uso dos elementos (ou função) definirá a semelhança de nomenclatura entre os elementos. Um esquema da organização hierárquica do *Add-in* pode ser observado na Figura 5.

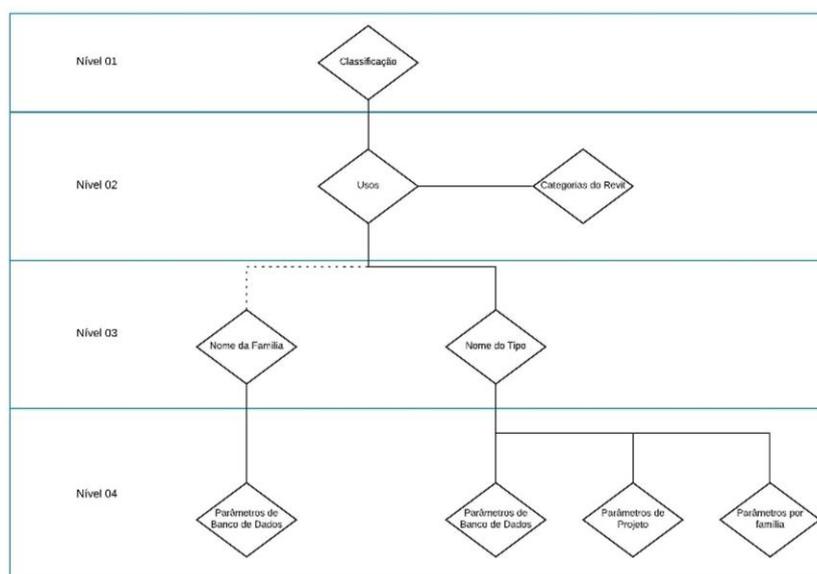


FIGURA 5: Níveis hierárquicos do *add-in*

FONTE: Autoria própria (Gráfico gerado pelo site www.lucidchart.com)

4.4.1 Primeira Aba (*Classification*)

Para que a definição de nomenclatura pudesse ser realizada no modelo, foi necessário pensar em uma subdivisão que, para elementos com finalidades semelhantes, houvesse também regras de nomenclatura semelhantes.

Foi pensado inicialmente em subdividir os elementos por categorias do próprio *Revit*[®] (paredes, pisos, forros, vigas entre outros). Cada categoria do *Revit*[®] possui características muito bem definidas quanto ao modo de ser inserida, como por exemplo, paredes serão sempre inseridas linearmente e em superfícies verticais, já pisos serão inseridos por área e em superfícies horizontais.

A ideia de subdivisão por categorias teve de ser repensada pois foi identificado logo no início uma necessidade de elementos de categorias de *Revit*[®] diferentes terem finalidades idênticas. O exemplo mais evidente foi o revestimento em argamassa. Neste caso, o revestimento seria feito usando categoria “Parede” por serem inseridos sobre uma superfície vertical, porém, em casos como o topo de um muro ou um reenquadramento de uma janela, parte desse revestimento deve ser inserida sobre uma superfície horizontal e logo, ser usado uma outra categoria do *Revit*[®] como o “Piso” assim como representado na Figura 6.

Foi então estabelecido que a divisão das nomenclaturas seria agora definida por um conjunto de Categorias do *Revit*[®] que possuem uma finalidade em comum. Para isso, foi criada a subdivisão “Uso” no *Add-in*. Essa subdivisão, representada no nível dois na Figura 5, seria então composta por quantas categorias do *Revit*[®] fossem necessárias para atender ao uso em questão. Essa subdivisão mostrou-se eficiente para subdividir o modelo em por exemplo, Revestimento Externo, Estrutura ou Fundação, porém, dois cuidados tiveram que ser tomados.

O primeiro deles leva em conta os dois tipos de elementos do *Revit*[®] descritos na referência bibliográfica: Famílias Carregáveis e Famílias de Sistemas, que possuem modos de inserção no modelo distintos. Cada categoria do *Revit*[®] se encontra em um destes dois tipos e por isso foi definido que cada uso poderia conter ou somente categorias de famílias carregáveis ou somente categorias de famílias de sistema.

O segundo fator a ser levado em conta era que a definição de uma única hierarquia na subdivisão dos elementos tornaria a organização visualmente difícil, dada a quantidade de usos existentes em uma obra. Foi então criada uma nova subdivisão com hierarquia superior ao “Uso”, denominado “Classificação” (representada pelo nível 1 na Figura 5). Nessa nova divisão, por exemplo, uma classificação como “Paredes e Painéis” contém os usos “Revestimento Interno” e “Revestimento Externo”, não ocorrendo o mesmo para o uso “Contrapiso”, que estará em uma classificação mais adequada às suas características.

Portanto, é importante destacar que a subdivisão classificação não delimita elementos com parâmetros identificadores semelhantes dentro do modelo, mas sim elementos com características indiretas semelhantes como mostrado na Figura 6.

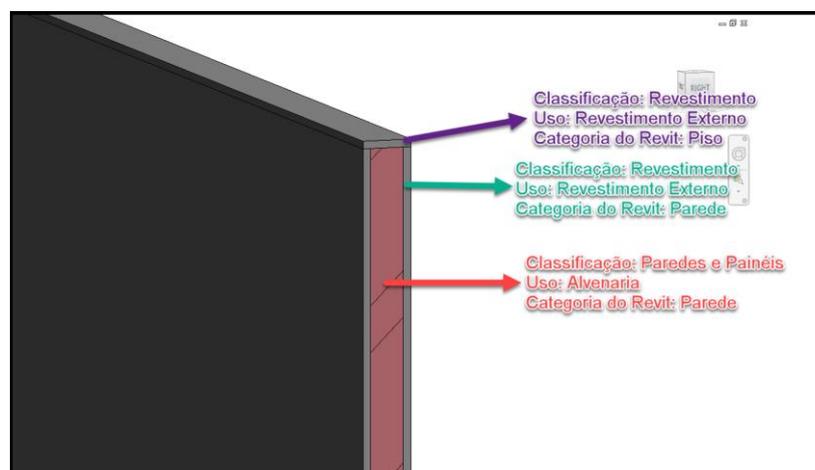


FIGURA 6: Exemplo de aplicação de revestimento em argamassa para parede em bloco de alvenaria

FONTE: Autoria própria

O resultado dos processos relatados pode ser visto na Figura 7 e nela, as três subdivisões descritas: Classificação (“*Classification*”), Uso (“*Use*”) e Categoria do *Revit*[®] (“*Category*”). Para cada um desses campos, foram criados botões para que o usuário adicione, edite ou remova itens de cada local.

O uso aqui deve ser entendido como o objeto central deste trabalho, pois é ele quem armazenará as configurações de nomenclatura e é nele que serão vinculadas as famílias no modelo. Apesar de estar no nível dois, o uso é a subdivisão importante deste projeto, estando a classificação acima dele meramente para organizar os usos de uma maneira que possam ser encontrados como uma maior facilidade.

É importante evidenciar que há um vínculo criado internamente entre os três campos da primeira aba. Isso significa que uma categoria do *Revit*[®] somente poderá ser inserida caso um uso tenha sido

criado, que por sua vez somente poderá ser criado, caso uma classificação tenha sido criada. Na Figura 7, é possível observar como exemplo que a classificação “Estrutura” contém o uso “Vigas”, que por sua vez contém a categoria do Revit® “Vigas Estruturais” (“Structural Framing”).

A lógica por trás da primeira aba consiste em um conjunto de dados que estão vinculados entre si de forma ordenada, como em uma tabela. Para melhor entendimento dessa tabela, essa lógica foi resumida em um formato de planilha que contém alguns exemplos do que está sendo proposto na primeira aba (Figura 8).

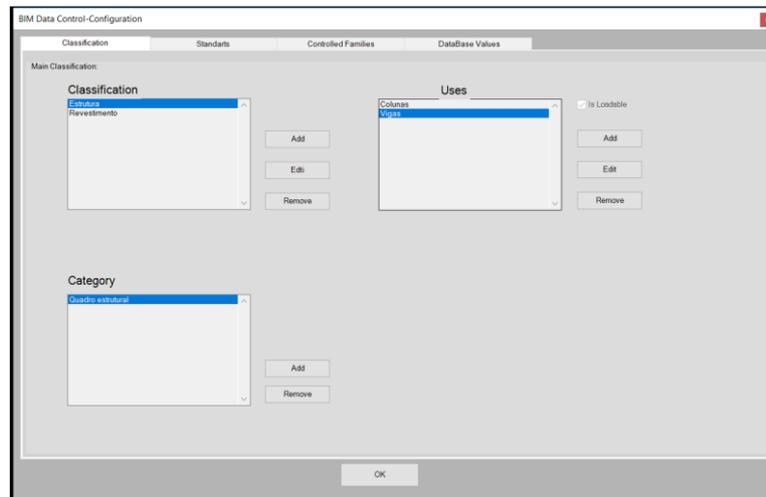


FIGURA 7: Exemplo da primeira aba do *add-in*

FONTE: Autoria própria

Classificação	Uso	Tipo	Categorias do Revit
Paredes e Painéis	Alvenaria	Família de Sistema	Paredes
Revestimento	Revestimento Interno	Família de Sistema	Paredes
			Pisos
	Pedras	Família de Sistema	Forros
			Paredes
Estrutura	Vigas	Família Carregável	Vigas Estruturais
	Colunas	Família Carregável	Colunas
			Colunas Estruturais

FIGURA 8: Tabela relacional da primeira aba da primeira aba

FONTE: Autoria própria

4.4.2 Quarta Aba (Database Values)

Antes de iniciar a segunda aba, é necessário que seja introduzido primeiramente a quarta aba nesta metodologia. Dentre os três tipos parâmetros que serão falados na segunda aba, dois tipos estão diretamente ligados ao Revit® e existem independentemente do uso ou não do *Add-in*. Já o terceiro, os parâmetros de banco de dados (*DataBase Parameters*), se encontram fora do modelo e são aplicados por meio do *Add-in*.

Para este trabalho, os parâmetros de banco e seus valores devem ser inseridos manualmente, como é o caso de um parâmetro “Fornecedor”, que poderia receber valores como “Fornecedor A” e “Fornecedor B” como exemplificado na Figura 9.

Os parâmetros de banco representados podem passar a ideia equivocada de que os mesmos podem ser substituídos por parâmetros do próprio Revit® e logo, ser controlado diretamente pelo Revit®

sem o auxílio de um *software* adicional. Porém, o parâmetro de banco foi inserido nesta aba como uma inserção manual por uma simplificação para o desenvolvimento do *Add-in*. Isso ocorre, pois, o parâmetro de banco foi pensado com um potencial vínculo à *softwares* externos ao *Revit*. Como exemplo de *softwares* externos, pode-se citar um parâmetro ligado à EAP (Estrutura analítica do projeto) e que esteja presente em *software* de planejamento como o Microsoft Project® ou um parâmetro de “Código de Orçamento” e que esteja vinculado à uma solução ERP (*Enterprise Resource Planning*).

Os dados externos ao *Revit*® podem então, por meio dessa aba, ser vinculados ao *Revit*® para permitir que a nomenclatura dos elementos esteja conectada diretamente com a fonte de base à que o usuário julgue necessário de acordo com os fins do modelo.

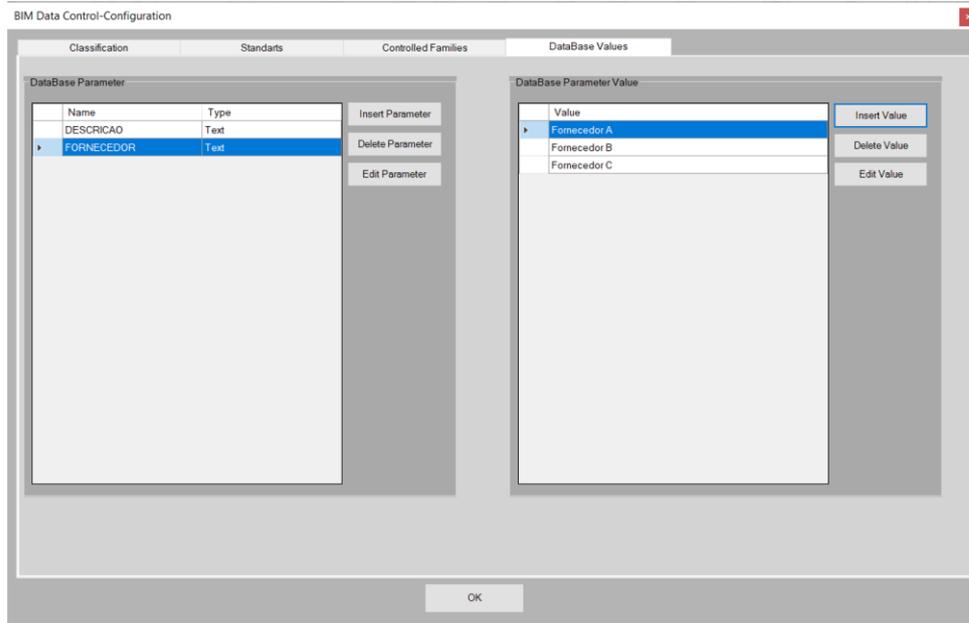


FIGURA 9: Exemplo da quarta aba do *Add-in*

FONTE: Autoria própria

4.4.3 Segunda Aba (*Standards*)

Na segunda aba do *Add-in* se encontram as regras de nomenclaturas definidas para cada um dos usos criados na primeira aba. Nela se observa um resumo de cada um destes usos por meio das colunas de classificação, uso, tipo da família (carregável ou de sistema), status atual da regra de nomenclatura, regra de nome da família (caso seja do tipo carregável) e uma última coluna para a regra de nome do tipo (para famílias carregáveis e de sistema). A Figura 10 mostra um exemplo com alguns usos já cadastrados.

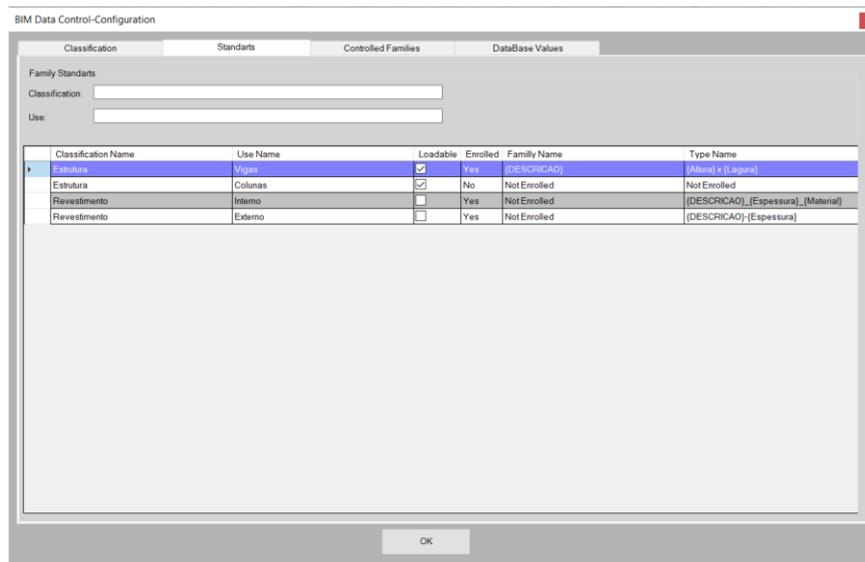


FIGURA 10: Exemplo da segunda aba do *Add-in*
FONTE: Autoria própria

Essa aba funciona apenas como um visualizador dos usos com sua respectiva padronização de nome (ou a falta dela). Nesta aba ainda há dois campos iniciais que podem ser utilizados para filtragem por uso e classificação para facilitar ao usuário encontrar o item que deseja mais rapidamente, pois foi considerando que em um projeto muito grande, encontrar determinados usos se tornaria uma tarefa difícil.

Ao clicar em qualquer um destes usos, uma nova janela aparecerá permitindo adicionar ou editar as regras de nomenclatura. Nesta nova janela, duas interfaces podem ser aparecer: Nomes de Família e Nomes de Tipo. Como já dito anteriormente, o *Revit*[®] subdivide seus elementos em carregáveis e de sistemas. Os elementos de sistemas não possuem nomes de família editáveis pelo usuário, somente o nome do tipo pode ser alterado, enquanto os elementos carregáveis possuem nome de família e nome de tipo editáveis.

A Figura 11 mostra a interface de edição da nomenclatura de uma família do uso “Vigas”, que neste caso, é um uso cadastrado para elementos carregáveis e por isso há duas abas, uma para nomenclatura da família e outra para tipo. Quanto à nomenclatura, o usuário poderá inserir os parâmetros (cujos nomes devem estar entre chaves para ser identificado como um parâmetro) que irão compor o nome das famílias. Entre os parâmetros, caracteres especiais poderão ser inseridos para facilitar a organização, assim como nomes auxiliares para aquele uso. A seguir estão representadas algumas alternativas à regra definida na Figura 11:

- “{DESCRICA}_ {FORNECEDOR}”;
- “{DESCRICA} - {FORNECEDOR}”;
- “Viga do tipo {DESCRICA} / {FORNECEDOR}”;
- “{FORNECEDOR}_ {DESCRICA}”;
- “{FORNECEDOR} - {DESCRICA}”.

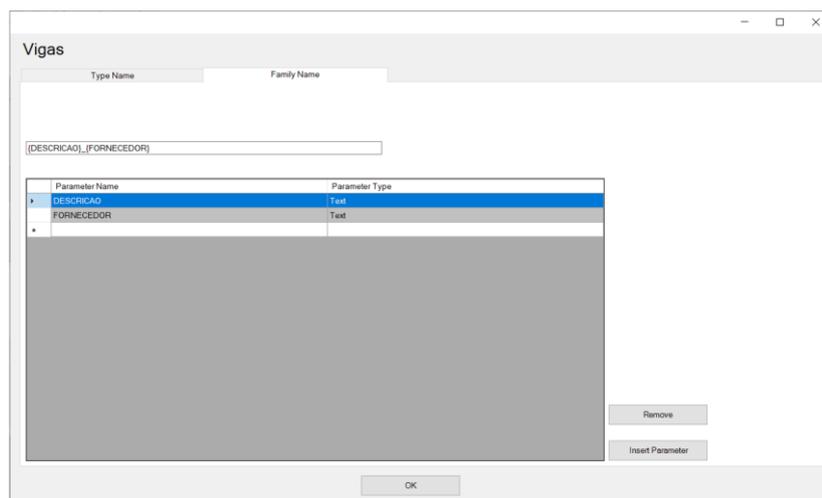


FIGURA 11: Exemplo de cadastro de nomenclatura para um uso

FONTE: Autoria própria

Para o nome da família, é importante ressaltar que somente parâmetros de identificação do tipo banco de dados (os mesmos vistos na quarta aba) podem ser ali vinculados pois se tratam de parâmetros externos ao Revit®.

Enquanto a customização da nomenclatura de família está limitada a elementos carregáveis, a customização da nomenclatura de tipo pode ser realizada tanto para famílias carregáveis quanto para famílias de sistema.

Na Figura 12 é descrito um exemplo de definição da nomenclatura para tipo em famílias carregáveis. Para a composição do nome do tipo, as regras de digitação e caracteres obrigatórios aqui são as mesmas utilizadas anteriormente para o nome de família, porém, aqui tem-se a diferença de que há mais dois tipos de parâmetros de identificação disponíveis para inserção no nome:

- Parâmetro de Projeto (*Revit® Parameter*): Parâmetro nativo do Revit® e que pode ser criado pelo usuário dentro do próprio Revit® de acordo com as informações que necessitam ser armazenadas. Este parâmetro, como discutido na referência bibliográfica, é atribuído a categorias do Revit® escolhidas pelo usuário.
- Parâmetro de Execução (*Runtime Parameter*): Parâmetro não nativo do Revit® e criado pelos autores para permitir que diferentes famílias tivesse a nomenclatura de seus tipos padronizada. O parâmetro de execução é a representação de parâmetros internos de uma família. Para ajudar na compreensão, pode-se citar o exemplo de duas famílias de vigas com os seguintes parâmetros internos:
 - Família de Viga Retangular: “Largura” e “Altura”.
 - Família de Viga Quadrada: “Largura”.

Considerando uma regra de nomenclatura “{Largura} x {Altura}”, identificou-se um problema, pois a viga retangular teria esses parâmetros disponíveis, porém a viga quadrada tem somente um parâmetro para os dois valores. Por meio então de dois parâmetros de execução (“Dimensão 01” e “Dimensão 02”) e a nova regra de nomenclatura como sendo “{Dimensão 01} x {Dimensão 02}”, tem-se que:

- Família de Viga Retangular: “Dimensão 01” será vinculado à “Largura” e “Dimensão 02” à “Altura”.
- Família de Viga Quadrada: “Dimensão 01” e “Dimensão 02” serão vinculadas à “Largura”.

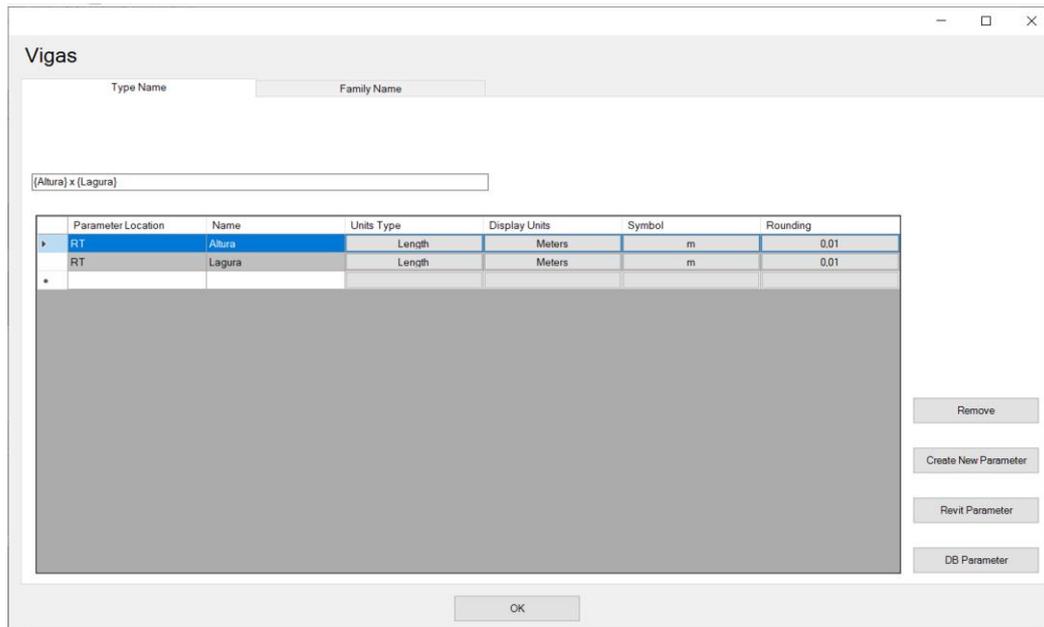


FIGURA 12: Janela para cadastramento de nomenclatura para um tipo
FONTE: Autoria própria

Uma vez inseridos os parâmetros de identificação desejados à regra de nomenclatura, é oferecido ao usuário a opção de o usuário especificar, caso o parâmetro seja número, a maneira como o valor deve ser exibido. Para isso, o usuário deve indicar a unidade que deve ir ao nome, o símbolo e a aproximação desejada. Essa funcionalidade permite que diferentes unidades possam estar presentes no nome, independentemente da unidade oficial usada no modelo, logo, um tipo de viga retangular poderá ter o nome: “Viga 15cm x 30cm”, tornando o nome customizável para atender elementos de qualquer grandeza e gerando um nome final mais próximo do que seria uma nomenclatura usual.

Depois de definida a nomenclatura desejada, ao clicar em “OK” a regra é salva para o uso em questão e a tela inicial da segunda aba reaparece como na Figura 11, porém agora mostrando as novas regras cadastradas.

Ao longo de todo o *Add-in* foram utilizadas telas de confirmação tanto para comandos concluídos com sucesso como também para comandos que falharam. Para este último, o aviso contempla o problema em questão, como por exemplo a inserção na regra de nomenclatura de um parâmetro inexistente, permitindo ao usuário tomar providências sobre o problema de forma mais rápida. Em um processo manual, o erro poderia ser visto somente quando o modelo já estivesse abastecendo algum uso como orçamentação ou planejamento por exemplo. A Figura 13 mostra um caso de um uso que teve sua regra de nomenclatura editada com sucesso.

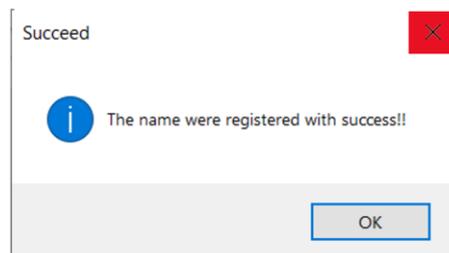


FIGURA 13: Aviso de comando realizado com sucesso
FONTE: Autoria própria

4.4.4 Terceira Aba (*Controlled Families*)

A terceira aba do *Add-in* é responsável por vincular as famílias e tipos do *Revit*® aos usos criados na primeira aba assim como fazer a alteração dos nomes segundo o que foi definido na segunda aba. A Figura 14 mostra o layout desta aba.

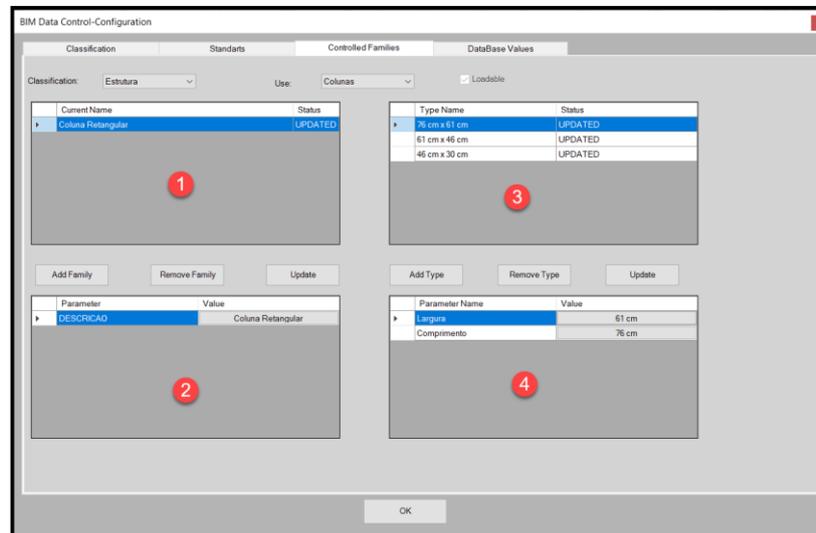


FIGURA 14: Exemplo da terceira aba
FONTE: Autoria própria

Primeiramente, na parte superior, deve-se selecionar a classificação e respectivo uso a serem trabalhados. A partir daqui há dois possíveis cenários, um com usos vinculados a famílias de sistemas e outro para famílias carregáveis.

Para famílias carregáveis, a seleção do uso resultará em mostrar as famílias vinculadas a este uso no espaço marcado com “1” na Figura 14, juntamente com o status atual da família, que pode estar definido como:

- *Updated*: O nome da família já está atualizado segundo a configuração de nomenclatura definida.
- *Ready to Update*: A regra de nomenclatura está definida, porém os valores dos parâmetros foram atualizados e o nome da família precisa ser atualizado.
- *Incompleted*: Os parâmetros que compõem a família não tiveram seus valores preenchidos para aquela família ou as regras de nomenclatura ainda não foram completamente definidas na segunda aba.

A partir deste ponto, novas famílias podem ser adicionadas ou removidas por meio dos botões disponíveis abaixo deste espaço. Ao ser selecionado uma família, o campo 2 irá mostrar os parâmetros de banco de dados (lembrando que são os únicos permitidos para nomenclatura de famílias) que precisam ser preenchidos para que o nome da família seja alterado.

Ao ser selecionado uma família no campo 2, o campo 3 também será automaticamente preenchido com os tipos daquela família cadastrados. O status do tipo poderá se enquadrar nos três vistos para família seguindo as mesmas regras. Ainda no campo 3 poderão ser adicionados ou removidos novos tipos por meio dos botões imediatamente abaixo deste campo.

O campo 4 possui o mesmo modo de funcionamento do campo 2 visto anteriormente, dada a ressalva que aqui também devem ser preenchidos os valores dos parâmetros de execução e projeto.

Com as famílias/Tipos vinculados nos campos 1 e 3 e os parâmetros devidamente preenchidos nos campos 2 e 4. O status das famílias e tipos estará como “*Ready to Update*”. A partir disso, o *Add-in*

estará apto para realizar as atualizações nos nomes dos elementos e clicando no botão “Update” a atualização será efetivada.

5. APLICAÇÃO DO PROGRAMA

Para demonstrar o funcionamento do programa, um modelo exemplo, nativo de instalação do programa *Revit*[®], foi utilizado afim de demonstrar a facilidade de utilização do *software* bem como sua consistência. Para simplificar o exemplo, o sistema de controle foi aplicado somente a categoria de paredes.

As paredes, pertencentes à família *Basic Wall*, que já haviam materiais cadastrados em suas camadas, tiveram o parâmetro *Description* preenchido com informações genéricas descrevendo uma numeração única para cada uma, como mostra a Figura 15 uma tabela retirada do próprio *software* descrevendo a nomenclatura sem controle do Tipo, o valor atribuído do parâmetro *Description* e também o nome da família ao qual pertence, e como explicado anteriormente as paredes são famílias de Sistema e portanto não são modificáveis em nomenclatura e alteração de qualquer propriedade nativa do mesmo, logo sua nomenclatura não poderá ser controlada.

<Wall Schedule>		
A	B	C
Family	Type	Description
Basic Wall	CL_W1	Parede 7
Basic Wall	Foundation - 300mm Concrete	Parede 6
Basic Wall	Interior - 165 Partition (1-hr)	Parede 5
Basic Wall	Interior - Partition	Parede 4
Basic Wall	Retaining - 300mm Concrete	Parede 3
Basic Wall	SIP 202mm Wall - conc clad	Parede 2
Basic Wall	Wall - Timber Clad	Parede 1

FIGURA 15: Tabela de paredes do *Revit*[®]

FONTE: Autoria própria

A partir de então, foi definido que as paredes deveriam estar na Classificação Alvenaria e no Uso Vedação. Sendo assim, a nomenclatura foi definida que para as paredes do modelo que estavam cadastradas ao uso Vedação tivessem o nome com a seguinte estrutura de formação: Vedação_Material_Descrição_Espessura, como mostra na Figura 16, porém como a versão do *Revit*[®] estava em inglês, a parametrização estava nesta língua (Vedacao_Structural Material_Description_Thickness).

Vedação						
Type Name						
Vedacao_(Structural Material)_(Description)_(Thickness)						
Parameter Location	Name	Units Type	Display Units	Symbol	Rounding	
RT	Thickness	Length	Centimeters	cm	0.01	
RV	Structural Material	Number				
RV	Description	Number				
*						

FIGURA 16: Exemplo de regra de nomenclatura para um tipo

FONTE: Autoria própria

Uma vez definido a nomenclatura, basta cadastrar quais tipos deverão ser controlados para que a parametrização seja preenchida automaticamente. Sendo assim, na Figura 17 a seguir são mostrados os tipos cadastrados e a definição, que acontece apenas uma vez, do parâmetro de *RunTime*. Essa definição

acontece apenas uma vez porque uma vez aplicado a um tipo, os outros tipos do mesmo Uso terão o parâmetro de *RunTime* definido com a mesma regra.

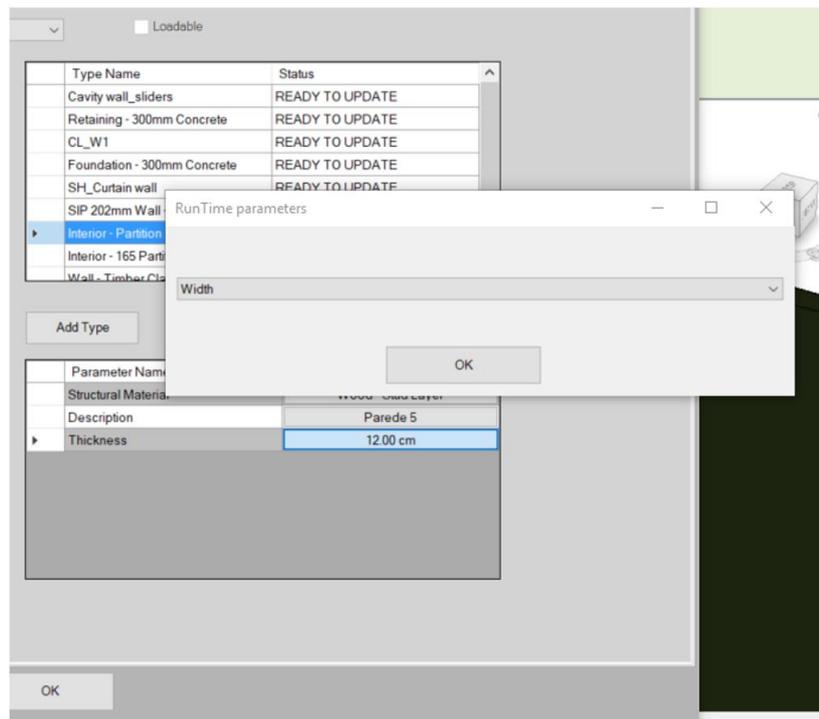


FIGURA 17: Janela de Atualização de Parâmetros RunTime

FONTE: Autoria própria

Uma vez definidos o valor de *RunTime*, automaticamente o Status de Atualização das famílias serão modificados para “*Ready To Update*” que significa que todos os parâmetros foram já contemplados com valores para ser parte da composição da nomenclatura dos objetos, como mostra a Figura 18.

Type Name	Status
Cavity wall_sliders	READY TO UPDATE
Retaining - 300mm Concrete	READY TO UPDATE
CL_W1	READY TO UPDATE
Foundation - 300mm Concrete	READY TO UPDATE

FIGURA 18: Status de tipos para receberem uma atualização de nome

FONTE: Autoria própria

Apertando o botão para fazer a atualização das nomenclaturas, o botão “*Update*”, os nomes dos elementos que estavam com todos os valores definidos mudaram automaticamente sua estrutura de nome se adaptando à customização imposta. A Figura 19 mostra como ficou a mesma tabela da Figura 15 após a atualização dos elementos.

<Wall Schedule>		
A	B	C
Family	Type	Description
Basic Wall	Vedacao_Concreto Tipo 1_Parede 7_28.00 cm	Parede 7
Basic Wall	Vedacao_Concreto Tipo 2_Parede 6_30.00 cm	Parede 6
Basic Wall	Vedacao_Concreto Tipo 3_Parede 3_30.00 cm	Parede 3
Basic Wall	Vedacao_Madeira_Parede 4_12.00 cm	Parede 4
Basic Wall	Vedacao_Madeira_Parede 5_16.51 cm	Parede 5
Basic Wall	Vedacao_Painel de Madeira_Parede 1_20.20 cm	Parede 1
Basic Wall	Vedacao_Painel de Madeira_Parede 2_20.20 cm	Parede 2

FIGURA 19: Lista de paredes depois da atualização pelo *Add-in*

FONTE: Autoria própria

6. CONSIDERAÇÕES FINAIS

A ideia deste trabalho surgiu com a necessidade de gerenciar modelagens paramétricas de projetos de edifícios multipavimentos. A dificuldade percebida por inúmeros projetistas e consultores de escritórios de projeto no âmbito da construção civil se relaciona diretamente com a gerência das necessidades de organização de nomenclatura e informação no projeto.

Então ao pesquisar por ferramentas que poderiam facilitar o emprego de padrões de nomenclatura customizados, observou-se que não havia ou ao menos não foram encontradas. Sendo assim, surgiu uma inquietação por parte dos autores em tentar buscar melhorias para a comunidade que aplica BIM em seus projetos, a fim de impulsionar a padronização e a busca de qualidade para obtenção de informação em um modelo paramétrico.

O *software* apresenta bom desempenho quando aplicado a modelos paramétricos de grande volume de informações, pois mostra a facilidade proporcionada por definir regras parametrizadas e customizadas de nomenclatura que são facilmente gerenciadas.

Acredita-se que o *software* ainda não está em sua versão final, pois existem potenciais que ainda não foram explorados como, por exemplo, a interface de parâmetros de banco de dados e seus valores, citados na quarta aba do *Add-in*, que seriam melhor empregados a uma interface *web*, onde o dono do modelo paramétrico, que é o contratante, forneceria as informações necessárias para a correta associação com a nomenclatura. Essa interface também poderá ser bem aplicada com a associação a bancos de dados de ERP's (*Enterprise Resources Planning*), em português, Gerenciador de Recursos da Empresa relacionando seus setores com propriedades do modelo paramétrico BIM.

Outro potencial importante que será explorado nas próximas versões será a associação com a plataforma *web* da Autodesk, chamada de *Forge*, que permitirá a consulta de dados do modelo paramétrico, em um navegador *web*, possibilitando que o contratante defina as regras de nomenclatura assim como sua parametrização via interface *web*.

Os resultados obtidos e os potenciais a serem desenvolvidos a partir do que foi proposto neste trabalho permitem contribuir para o aprimoramento das produções de modelos virtuais das edificações por meio da melhoria da organização e controle dos dados durante os processos.

Cabe ainda destacar como sugestão de pesquisas, a partir deste trabalho, a análise da importância da aplicação de padrões de nomenclatura, utilizando essa ferramenta desenvolvida, em um modelo BIM com objetivos específicos de análise financeira, planejamento e controle de obra ou para a coordenação de projetos.

7. REFERÊNCIAS BIBLIOGRÁFICAS

AIA – THE AMERICAN INSTITUTE OF ARCHITECTS. **National Documents Committee. Document E202 – Building Information Modeling Protocol Exhibit.** Washington, DC, 2008.

CATELANI, W. S.; SANTOS, E. T. Normas brasileiras sobre BIM. **Concreto & Construções**, v. 44, n. 84, p. 54-59, 2016.

Central de ajuda Autodesk Revit 2019. Disponível em:< <http://help.autodesk.com/view/RVT/2019/PTB/>> . Acesso em: 06 de maio de 2023.

CHEN, K.; LU, W.; WANG, H. D.; NIU, Y.; HUANG, G. G. Naming Objects in BIM: A convention and a semiautomatic Approach. **Journal of Construction Engineering and Management**. v. 143, n. 7, 2017.

CURT. **Collaboration, Integrated Information and the Project Lifecycle in Building Design, Construction and Operation.** In: Committee, A. E. P. (Ed.) WP-1202. Construction Users Roundtable, 2004, 48 p.

EASTMAN, C.; TEICHOLZ, P.; SACKS, R.; LISTON, K. **Manual de BIM: Um Guia de Modelagem da Informação da Construção para Arquitetos, Engenheiros, Gerentes, Construtores e Incorporadores.** São Paulo: Bookman, 2014.

MELHADO, S. B.; ADESSE, E.; BUNEMER, R.; LEVY, M. C.; LUONGO, M.; MANSO, M. A. A gestão de projetos de edificações e o escopo de serviços para coordenação de projetos. In: LARES - Conferência Internacional Latin American Real Estate Society, 6., São Paulo, 2006. **Anais...**São Paulo: LARES/USP, 2006.

MESEGUER, A. G. **Controle e garantia da qualidade na construção**. São Paulo, Sinduscon-SP/Projeto/PW, 1991.

NOVAES, C. C. Ações para controle e garantia da qualidade de projetos na construção de edifícios. In: Workshop Nacional Gestão do Processo de Projeto na Construção de Edifícios, 1., 2001, São Carlos, SP. **Anais...** São Carlos: EESC/USP, 2001.

PENTTILÄ, H. Describing the Changes in Architectural Information Technology to Understand Design Complexity and Free-Form Architectural Expression. **Journal of Information Technology in Construction**, v. 11, n. 29, p. 395-408, 2006.

PICCHI, F. A. **Sistemas da qualidade: uso em empresas de construção de edifícios**. 1993. Tese (Doutorado) – Escola Politécnica - Universidade de São Paulo, São Paulo, 1993.

SACKS, R.; EASTMAN, C.; LEE, G.; TEICHOLZ, P. **BIM Handbook: a Guide to Building Information Modeling for Owners, Managers, Designers, Engineers, and Contractors**. Third ed. New Jersey: Wiley, 2018.

SHIROWZHAN, S.; SEPASGOZAR, S. M., EDWARDS, D. J., LI, H., WANG, C. BIM compatibility and its differentiation with interoperability challenges as an innovation factor. **Automation in Construction**, v. 112, 2020.

SUCCAR, B. Building Information Modelling Framework: a research and delivery foundation for industry stakeholders. **Automation in Construction**, v. 18, n. 3, p. 357-375, 2009.

SUCCAR, B.; KASSEM, M. Macro-BIM Adoption: conceptual structures. **Automation in Construction**, v. 57, p. 64-79, 2015.