

API: GERAÇÃO AUTOMÁTICA DE DESENHOS DE PORMENOR DE ARMADURAS EM ESTRUTURAS DE BETÃO

Ruben Pereira ⁽¹⁾, **Rodrigo Falcão Moreira** ^{(1) (2)}

(1) GROUND MOTION – Earthquake & Structural Engineering International, Lda, Porto

(2) CONSTRUCT-LESE, Faculty of Engineering, University of Porto

Resumo

O avanço significativo da construção digital, concretizado na metodologia BIM (*Building Information Modelling*), veio abrir portas a projetos e empreendimentos cada vez mais complexos. Essa complexidade é já uma realidade e tem influência significativa nas tarefas de análise estrutural e definição de processos construtivos. Em complemento, existe a necessidade premente de automatizar aquelas que são as tarefas mais repetitivas no desenvolvimento de um projeto. Essa automatização, desde que devidamente validada, permite a redução de erros e o controlo rigoroso e imediato de todas as tarefas.

Neste enquadramento, o presente artigo aborda a temática na perspetiva do projeto de estruturas de betão, em que a tarefa de representação de armaduras é morosa e pode revelar-se demasiado “*time consuming*”. A utilização de modelos BIM na sua pormenorização detalhada permite a verificação de incompatibilidades na modelação, criação de desenhos de pormenor, extração de quantidades e elaboração de mapas de ferros (*shop drawings*).

A abordagem aqui apresentada baseia-se no desenvolvimento e aplicação de uma ferramenta BIM focada na importação e tratamento da informação integrante do modelo digital. Esta ferramenta, desenvolvida em linguagem de programação C#, funciona como um *Add-in* ao software primário Autodesk Revit®. Incorpora uma interface gráfica e atua fundamentalmente no domínio da geração automática de desenhos de armaduras. O algoritmo desenvolvido vem colmatar algumas das lacunas de automatização do software primário, permitindo assim implementar um procedimento para pormenorização de armaduras de acordo com *templates* empresariais padronizados.

1. Introdução

A indústria de Arquitetura, Engenharia e Construção (AEC) há muito tempo que procura técnicas tanto para reduzir o custo e tempo de entrega, como para aumentar a produtividade e a qualidade dos projetos. A metodologia BIM tem o potencial para atingir esses objetivos [1].

A utilização de ferramentas que operam em ambiente BIM para a modelação de armaduras em elementos de betão armado permite uma melhor quantificação de um dos aspetos mais importantes num projeto de estruturas [2]. No entanto, a tarefa de modelar consome muito tempo e exige esforço assinalável, por vezes com o único objetivo de satisfazer um determinado nível de detalhe (LOD).

A modelação e pormenorização de armaduras deve ser ponderada para cada projeto mediante a exigência do cliente. Caso se pretenda uma rigorosa orçamentação das armaduras e a produção de *shop drawings* (mapas de ferros), por exemplo para projetos internacionais, a utilização de ferramentas BIM agiliza esse processo. Existem atualmente projetos que apenas recorrem à informação do modelo digital sem a necessidade de desenhos de pormenor. Em projetos em que o prazo é consideravelmente curto, a modelação completa de armaduras dificulta o cumprimento desse objetivo.

No contexto nacional existe ainda a necessidade de fornecer desenhos de pormenor de armaduras, sendo muitas vezes suficiente apresentar cortes tipo das secções dos elementos consoante a fase do projeto. A inclusão de alçados e cortes de armaduras para cada elemento em fase de execução justifica-se caso a jusante essa informação seja utilizada para fabricação/construção de forma a economizar tempo e recursos.

O presente artigo tem como objetivo apresentar um aplicativo informático que agiliza a geração de armaduras assim como a pormenorização das mesmas. O aplicativo cria armaduras com base em inputs mínimos de forma a tornar rápida a sua utilização, produz automaticamente desenhos de pormenor e apresenta-os de acordo com os templates padronizados da empresa que o desenvolveu (GROUNDMOTION – Earthquake & Structural Engineering International). Esta metodologia possibilita a rápida criação de *shop drawings* e *bar bending schedules* (tabelas onde são especificados os comprimentos de dobragem das armaduras) de acordo com a norma inglesa BS 8666:2005 [3].

A grande maioria dos atuais softwares BIM dispõem de API's que permitem a programação de algoritmos. Estes podem funcionar como *add-ins* tendo como propósito a implementação e automatização de tarefas de modelação, potenciando a adição das mais diversas competências ao software base. O software utilizado foi o *Autodesk Revit*®, que na sua API suporta a framework .NET 4.7 [4], e a linguagem de programação escolhida foi o C#.

O aplicativo funciona a partir da consideração de um modelo BIM, realizado no software *Autodesk Revit*®, cujos objetos / famílias se encontram organizados por categorias. No caso presente, os elementos adotados foram pilares em betão armado, sendo que se pretende que o aplicativo funcione futuramente com outros elementos.

2. Objetivos

O nível de maturação e implementação que a metodologia BIM apresenta no contexto nacional da indústria AEC obriga ainda à conversão de toda a informação digital em documentação 2D materializada em formatos passíveis de serem compreendidos, analisados e verificados por todos os intervenientes do processo [5]. Ora, estes formatos são aqueles que ainda hoje fazem parte do processo tradicional, nomeadamente *.xls*, *.pdf*, *.doc*, *.dwg* ou *dwf*.

Consequentemente, e tendo em conta que a nível interno o procedimento implementado na GROUNDMOTION assenta fundamentalmente na metodologia BIM, surgiu a necessidade de automatizar algumas destas tarefas de “conversão de informação” evitando qualquer perda de

legibilidade e representatividade da informação. Nesta perspetiva sinalizaram-se aquelas que, no âmbito do projeto de estruturas, seriam as mais sensíveis e repetitivas para que de uma forma estruturada se conseguisse conceber métodos e estratégias conducentes com uma eficaz modelação e gestão dessa informação. Entre as tarefas mais sensíveis destacam-se, claramente, a geração automática de peças desenhadas.

A modelação de armaduras pode ser efetuada recorrendo às ferramentas nativas do software ou agilizada através de *add-ins*. Qualquer das opções auxilia na geração de armaduras. No entanto, na pormenorização das armaduras (alçados ou cortes) será necessário complementar manualmente a informação com diferentes anotações (dimensões, textos ou *tags*). Estas limitações obrigam a um conjunto de passos incompatível com a automatização pretendida para o efeito. Portanto, sobra o recurso ao API do software como única alternativa viável para alcançar os objetivos traçados.

Doravante, o artigo focar-se-á na forma como foi materializada a automatização do processo de geração automática de armaduras e elaboração do Quadro de Pilares. Com efeito, pretende-se dar maior ênfase às potencialidades associadas à exploração da API do software e cujos fins / usos podem ser os mais diversos.

3. Desafios

O desenvolvimento de *add-ins* ou outras ferramentas informáticas para complemento, automatização e extensão das competências dos softwares BIM disponibilizados no mercado é um desafio atual e que julgamos será crescente nos tempos que se aproximam. A necessidade de otimizar a gestão da quantidade, cada vez maior, de informação gerada em cada processo, o aumento de produtividade exigido pela competitividade do mercado e a complexificação das geometrias associadas aos novos empreendimentos obrigam a um crescente domínio das competências associadas às ciências e tecnologias de programação.

A maioria dos softwares existentes no mercado promovem a possibilidade da criação dessas ferramentas através da utilização das API's. Contudo, estas API's requerem conhecimentos sólidos de programação e encontram-se escritas em linguagens de programação complexas como C# ou C++.

A modelação geométrica dos elementos, num projeto de estruturas, revela-se como uma tarefa simples quando comparada com a modelação de armaduras ou a criação de plantas, cortes e detalhes com a informação necessária para a sua execução. A correta quantificação e elaboração de *shop drawings* com *bar bending schedules* depende da modelação de todas as armaduras, o que implica um acréscimo no tempo de modelação.

Na tarefa de filtrar os pilares no mesmo alinhamento vertical, com o objetivo de atribuir a mesma designação e comparar com outros pilares do modelo, não é possível automatizar com recurso às opções nativas do software. A opção *Element Positioning*, disponibilizada em *Autodesk Revit® Extensions* nas versões anteriores ao *Autodesk Revit® 2019*, permitia atribuir a designação dos elementos automaticamente, no entanto, distinguia os pilares não só por tipo como também por piso. A necessidade de ter os pilares do mesmo alinhamento agrupados e comparar esses conjuntos com base em determinadas propriedades é apenas possível de alcançar através da API do software.

No fim pretendia-se que o aplicativo colocasse automaticamente o Quadro de Pilares referente ao modelo numa nova folha (*Sheet*) devidamente preenchida com vistas de pormenor dos diferentes pilares.

4. Aplicativo

O aplicativo foi desenvolvido com recurso ao *Microsoft® Visual Studio 2015* permitindo, em ambiente *Autodesk Revit®*, a execução e a validação do código. A utilização desta ferramenta prendeu-se com a necessidade de criar um aplicativo cuja interface fosse de fácil e intuitiva utilização. O aplicativo está dividido em três tarefas: (i) a geração de armaduras; (ii) a designação dos pilares e (iii) a elaboração do quadro de pilares. A geração de armaduras e a designação dos pilares podem ser realizadas sem qualquer ordem específica. O desenvolvimento do aplicativo implicou o acesso aos parâmetros (de instância ou tipo), à geometria dos pilares de betão armado e a métodos para criação de novos elementos, com a dificuldade inicial de compreender como a API do software disponibiliza essa informação. Essa informação serve de base para a geração de armaduras, assim como, para complementar a informação apresentada no quadro de pilares.

4.1. Interface gráfica

A interface gráfica utilizada é do tipo *Windows Presentation Foundation (WPF)* que exhibe como principal vantagem a separação entre o design e o comportamento, promovendo a criação de um código melhor organizado. O design no *WPF* é criado geralmente no *XAML (Extensible Application Markup Language)* e o comportamento/código é implementado numa linguagem de programação [6], neste caso o C#. Isto possibilita que a interface seja criada por um designer e o código por um programador, de maneira independente.

4.2. Algoritmo

A organização do código foi materializada à custa da criação de classes que agrupam propriedades e métodos para agilizar a interação com os diferentes propósitos do aplicativo. O aplicativo é dividido em três funções principais: geração de armaduras, designação dos pilares e criação de quadro de pilares com vistas de pormenor.

Na geração de armaduras existem diversas classes com o objetivo de recolher informações relativas à geometria do elemento, proceder à criação das armaduras horizontais e verticais e auxiliar na definição de métodos de cálculo. Previamente à criação de armaduras, o aplicativo realiza uma análise para que, no caso de o pilar ser intersetado por paredes e/ou lajes, seja executado automaticamente o comando *Switch Join Order* de modo a garantir que a sua geometria não seja alterada.

O aplicativo permite gerar armaduras para os pilares com a mesma geometria simultaneamente ou separadamente. O diâmetro e quantidade das armaduras verticais deve ser especificado pelo utilizador, assim como, o diâmetro e espaçamento do estribo exterior do pilar. Dependendo da quantidade de varões verticais, são gerados automaticamente estribos adicionais interiores. A geometria dos estribos interiores e a duplicação destes na zona de base e topo são criados com base em regras definidas no código do aplicativo.

Na tarefa de designação dos pilares houve a necessidade de filtrar os pilares e definir as propriedades a extrair. Esta tarefa pode ser realizada antes da geração de armaduras sendo que,

se existirem objetos sem armadura, o aplicativo emite um alerta, prosseguindo se ter esse parâmetro em conta na diferenciação dos pilares. A classe *ColumnsUtils* foi criada especificamente para agrupar todas as operações relativas à designação dos pilares.

Inicialmente o algoritmo filtra os níveis presentes no modelo, ordenando-os por elevação de forma ascendente (ex. Nível -1, Nível 0, etc.). De seguida, para cada nível é criada uma listagem com os pilares em betão que estejam associados ao mesmo. Posteriormente, para cada pilar dessa listagem é criada uma “caixa delimitadora” (*BoundingBox*) que permite selecionar os pilares que se encontrem acima. A informação necessária para a diferenciação de cada pilar, que se encontre dentro da “caixa delimitadora”, é registada na classe *ColumnsSet* sendo cada *ColumnsSet* associado a um conjunto.

A classe *ColumnsSet* possui três listas que armazenam informação relativa aos pilares. A *ListElement* e a *ListElementID* contêm dados que são utilizados para acelerar as diferentes iterações no código. A *ListElementInfo* reúne informação como, o nível a que o pilar está associado, a sua geometria, a distância ao eixo e quantidade e diâmetro das armaduras. Esta informação permite comparar os diferentes conjuntos, com base na *ListElementInfo*, para encontrar os que são idênticos, atribuindo assim a mesma designação (Figura 1).

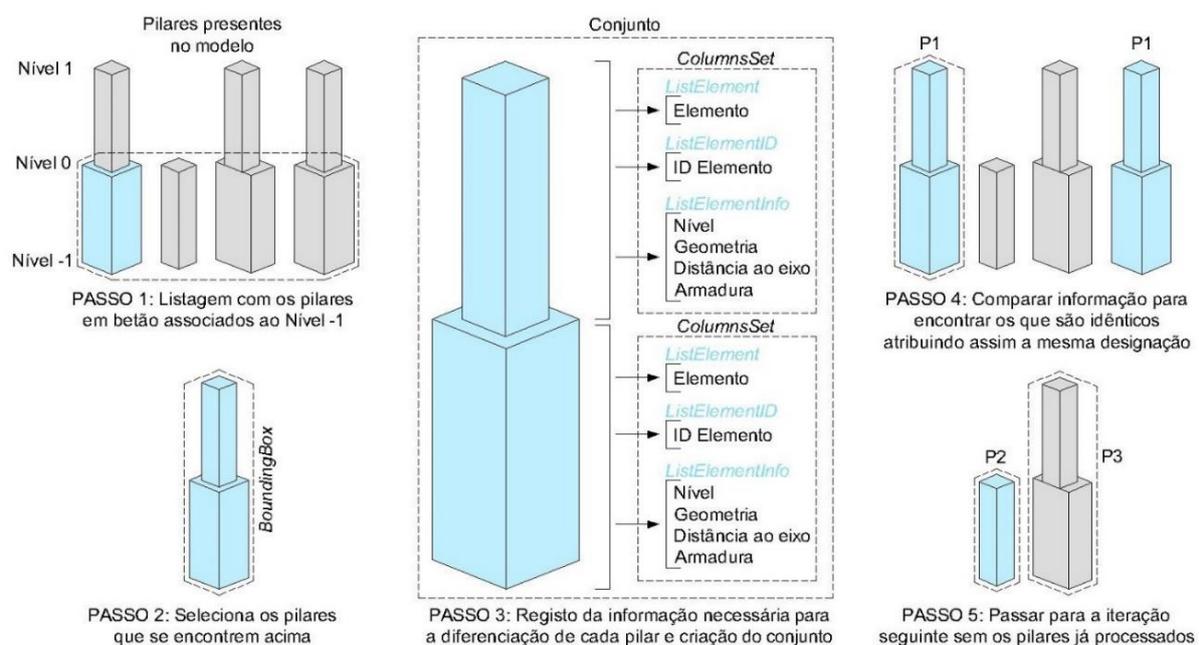


Figura 1: Estrutura do processo.

Em paralelo com a recolha de informação, foram criados métodos para determinar, caso a designação tenha sido introduzida / modificada manualmente, a existência de objetos sem armaduras e/ou sem identificação, conjuntos idênticos com identificação diferente ou se a numeração não é sequencial. Não obstante, o aplicativo reúne a informação relativa a cada situação e emite um alerta ao utilizador (Figura 2). Este processo funciona como um mecanismo de controlo do modelo. No entanto, se existirem pilares sem designação e/ou com designação incorreta, o aplicativo não avança se a situação identificada não for corrigida.

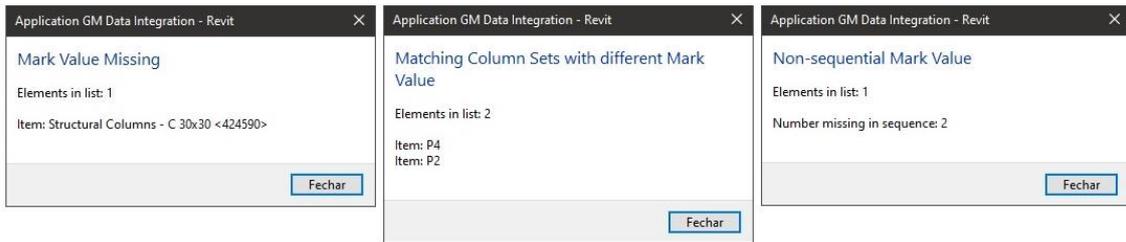


Figura 2: Mecanismo de controlo dos objetos.

Após validar a correta designação dos pilares o aplicativo cria, para cada pilar do conjunto, vistas de pormenor. As vistas incluem dimensões da geometria do pilar e a distância ao eixo, assim como informação relativa à armadura vertical e horizontal.

A esquadria do Quadro de Pilares é gerada recorrendo inteiramente à API do software, criando uma nova listagem (*Schedules/ Quantities*), opção nativa do *Autodesk Revit®*, editando o cabeçalho de modo a poder inserir o número de linhas correspondente ao número de níveis presentes no modelo e colunas correspondentes ao número de designações de pilares diferentes. O Quadro de Pilares é inserido numa nova folha (*Sheet*), assim como as vistas de pormenor dos pilares.

4.3. Utilização

O aplicativo inicia apresentando uma listagem na qual é possível escolher de que forma os elementos são apresentados (Figura 3). O utilizador pode optar pela opção *By Type* que apresenta os diferentes tipos de pilares existentes no modelo. Na eventualidade de o utilizador pretender definir armadura para um pilar ou conjunto específico, pode seleccionar a opção *Selected* ou *By Set* respetivamente. A opção *Show elements with Reinforcement* permite visualizar apenas os elementos que não possuem armaduras.

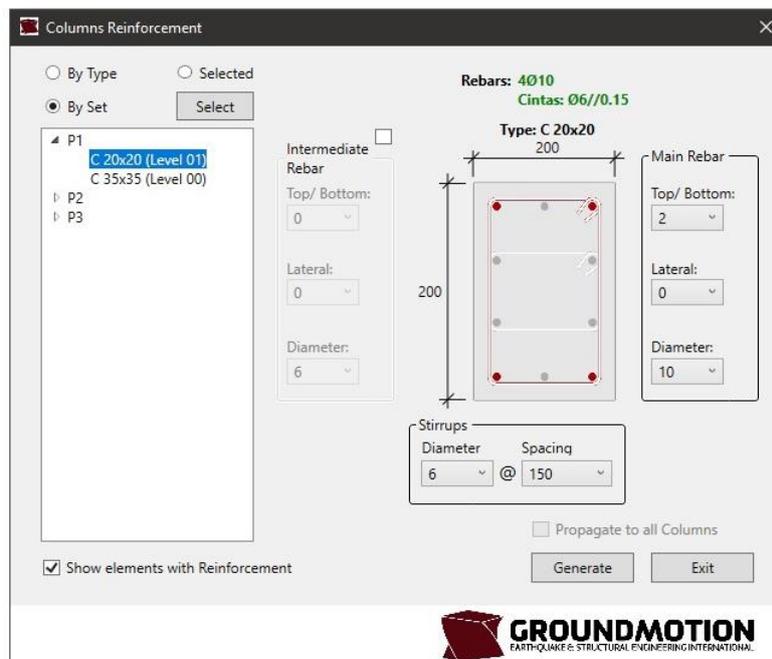


Figura 3: Interface do Aplicativo.

Após a seleção do tipo de pilar, o utilizador deve especificar a armadura vertical e horizontal. Na armadura vertical é possível definir diâmetros diferentes. A interface contém campos de preenchimento para a armadura com o diâmetro maior à direita da seção do pilar (*Main Rebar*), onde se define o diâmetro e o número de varões para cada face do pilar. Caso exista, pode-se especificar, de igual forma, armadura com diâmetro menor do lado esquerdo (*Intermediate Rebar*). O diâmetro e espaçamento do estribo exterior deve ser especificado no campo *Stirrups* situado abaixo da seção do pilar. Os estribos interiores são gerados automaticamente com o mesmo diâmetro do exterior.

A interface disponibiliza a opção *Propagate to all Columns* que permite propagar a armadura definida para os pilares com a mesma geometria, independentemente da sua designação ou conjunto, sendo apenas possível para a opção *By Type*. Após a definição da armadura de cada tipo de pilar devemos selecionar a opção *Generate* de modo a gerar armadura especificada para todos os pilares/ conjuntos idênticos. Posteriormente a informação relativa à armadura será inserida em parâmetros de instância do pilar, de modo a agilizar a sua alteração e/ou replicar para outros elementos.

Uma outra interface foi desenvolvida para proceder à designação dos pilares. Esta interface disponibiliza ao utilizador a possibilidade de especificar o prefixo, o número inicial, o sufixo e o sentido XY no qual pretende iniciar a numeração.

A elaboração do quadro de pilares aparece como passo final do processo. O utilizador deve apenas especificar que níveis pretende incluir no quadro. As vistas de pormenor são geradas automaticamente e inseridas numa nova folha (*Sheet*), juntamente com o quadro, tendo como base a correspondência entre o nível e a designação do pilar. Na eventualidade de, para o nível especificado, não exista correspondência o aplicativo coloca uma linha no espaço vazio (Figura 4).

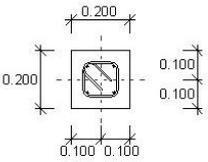
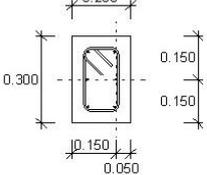
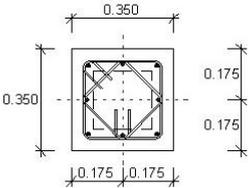
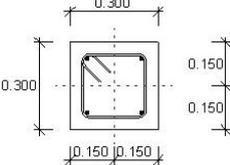
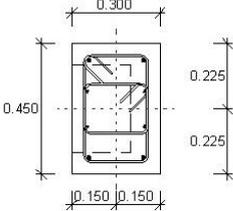
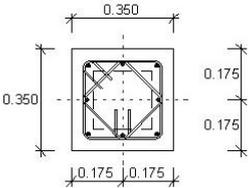
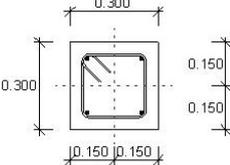
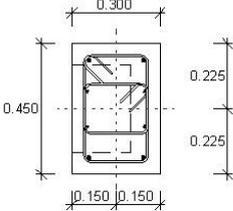
QUADRO DE PILARES			
	P1	P2	P3
Level 02	 <p>4Ø10 Cintas: Ø6//0.15</p>		 <p>6Ø10 Cintas: Ø6//0.15</p>
Level 01	 <p>8Ø16 Cintas: Ø6//0.15</p>	 <p>4Ø16 Cintas: Ø6//0.15</p>	 <p>4Ø16+4Ø12 Cintas: Ø6//0.15</p>
Level 00	 <p>8Ø16 Cintas: Ø6//0.15</p>	 <p>4Ø16 Cintas: Ø6//0.15</p>	 <p>4Ø16+4Ø12 Cintas: Ø6//0.15</p>

Figura 4: Quadro de pilares gerado automaticamente.

5. Limitações Atuais

O aplicativo foi, nesta primeira fase, desenvolvido para agilizar a geração de armaduras, sendo que algumas das regras criadas para evitar alguns passos ao longo do processo possam resultar apenas para a maioria dos casos.

A pormenorização de armaduras, de acordo com *templates* empresariais padronizados, implica que a representação gráfica, como as anotações, tabelas ou propriedades incluídas nos modelos de vista (*View Templates*) sejam definidas manualmente. No entanto, estas definições podem ficar guardadas no *template* do software para utilização posterior em diferentes processos e modelos.

A determinação da informação necessária está indexada à designação dos elementos do modelo digital. Logo, o aplicativo recolhe informação apenas dos elementos devidamente designados. O aplicativo requer um nível mínimo de detalhe e de informação a ler nos objetos, i.e., é necessário que todos os pilares tenham designação e armaduras para a correta elaboração do Quadro de Pilares.

6. Desenvolvimentos Futuros

A lógica metodológica em que o aplicativo se baseia permite uma integração com diferentes softwares. Nessa perspetiva, poderão as competências do aplicativo ser alargadas para que as propriedades para a geração de armaduras possam ser especificadas em ambiente *Microsoft® Excel* ou *Autodesk Robot®* importando essa informação para o software *Autodesk Revit®*.

O conhecimento adquirido da API do software na geração automatizada de armaduras servirá, no futuro, para extrapolar o procedimento para outros elementos de betão armado, possibilitando assim, um maior rigor na quantificação das armaduras e a criação automática de *Shop Drawings* com *Bar Bending Schedules*.

Referências

- [1] Salman Azhar, "Building Information Modeling (BIM): Trends, Benefits, Risks, and Challenges for the AEC Industry" in *Leadership and Management in Engineering* (2011), Vol. 11, Issue 3, pp. 241–252.
- [2] Marco Mellacqua, Elisa Bacchi e Davide Decarolis, "To Rebar or Not to Rebar: That Is the BIM Question", em *Autodesk University*, Las Vegas, 2017.
- [3] BSI, "Scheduling, dimensioning, bending and cutting of steel reinforcement for concrete. Specification" (2005).
- [4] Autodesk Knowledge Network – Development Requirements. URL: <https://knowledge.autodesk.com/search-result/caas/CloudHelp/cloudhelp/2018/ENU/Revit-API/files/GUID-FEF0ED40-8658-4C69-934D-7F83FB5D5B63-htm.html> (consultado em novembro de 2017)
- [5] P. Carvalho, "Avaliação do estado de implementação da tecnologia BIM no setor da construção em Portugal", Tese de Mestrado, Porto, FEUP, 2016.
- [6] Visual Studio Documentation - Introdução ao WPF (Windows Presentation Foundation), 2016, URL: <https://docs.microsoft.com/pt-br/visualstudio/designers/introduction-to-wpf> (consultado em novembro 2017)