

## **Design-by-Analogy: proposta para um modelo de ferramenta computacional de auxílio ao processo de design**

**Pablo E. Corrêa,** [pablo.ermida@gmail.com](mailto:pablo.ermida@gmail.com) – Programa de Pós-Graduação em Design, Universidade Federal do Rio Grande do Sul, Porto Alegre, Brasil

**Fábio G. Teixeira,** [fabioigt@ufrgs.br](mailto:fabioigt@ufrgs.br) – Programa de Pós-Graduação em Design, Departamento de Design e Exp. Gráfica, Universidade Federal do Rio Grande do Sul, Porto Alegre, Brasil

**Paulo J. Maldonado,** [paulomaldonado@inspaedia.com](mailto:paulomaldonado@inspaedia.com) – Departamento de Artes Visuais e Design, Escola de Artes, Universidade de Évora

### **Resumo**

*O desenvolvimento de novos produtos é uma atividade altamente complexa, onde a aplicação de metodologias e ferramentas para apoiar o processo de design é fundamental na condução das inúmeras atividades entre as etapas envolvidas. A fase conceitual no design é um momento crítico para o sucesso do produto, onde o acesso ao conhecimento de design gerado e a experiência prévia do designer são as bases para a construção do processo de criação. Diante disso, o campo Design-by-Analogy (DbA) é uma abordagem no design que utiliza os processos de analogias para estimular a geração de ideias na busca por soluções inovadoras através de técnicas e ferramentas computacionais. Neste contexto, este artigo tem como objetivo propor um modelo para o desenvolvimento de um software baseado nas técnicas em DbA para estimular e auxiliar o designer durante o processo projetual. Para atingir este objetivo, a revisão da literatura e o levantamento do estado da arte dos softwares existentes em DbA estabeleceram as bases de conhecimento para este estudo. A partir das informações obtidas destas bases, foram definidos critérios de análise para a avaliação das características de qualidade dos softwares existentes. Isto, reuniu um conjunto de características desejáveis para estabelecer os requisitos que guiarão a elaboração do modelo proposto, especificando seus elementos e estrutura. O resultado deste trabalho é a proposta deste modelo, onde a combinação de técnicas já existentes somada a novos elementos que exploram caminhos alternativos e promovem avanços para o campo do DbA. Este modelo será a base para o desenvolvimento de um software para auxílio e estímulo à fase conceitual de projeto, permitindo o registro dos conhecimentos gerados durante o desenvolvimento de projetos para que estes possam ser reutilizados em projetos futuros, acrescentando novas possibilidades que darão continuidade aos estudos em ferramentas digitais para apoiar o processo de design.*

**Palavras-chave:** *Design-by-Analogy, Processo de Design, Software.*

## **Design-by-Analogy: Proposal for a computational tool model to aid the engineering design process**

### **Abstract**

*New products development is a highly complex activity, where applying methodologies and tools to support the engineering design process is fundamental in conducting the numerous activities between the phases involved. The conceptual design phase is a critical moment for product success, where access to the existing design knowledge and the prior experience of the designer are the basis for building the creative process. Thus, the Design-by-Analogy (DbA) field is a design approach that uses analogous processes to stimulate ideas generation seeking innovative solutions through computational techniques and tools. In this context, this article aims to propose a model for a software development based on DbA techniques to stimulate and assist designers during project process. To fulfill this objective, literature review and existing state of the art DbA software survey establish the knowledge bases for this study. Given extracted information from these sources, analysis criteria were defined for the existing software quality characteristics evaluation. This, combined a set of desirable characteristics to establish the requirements that guided the elaboration of the proposed model, specifying its components and structure. The result of this paper is the model proposal, where combining existing techniques and new components that explore alternative paths will promote advances in DbA field. This model will guide the development of a software to aid and stimulate the conceptual phase of the project, allowing the registration of the knowledge generated during the development of projects so that they can be reused in future projects, adds new possibilities giving continuity to the studies in digital tools to support the engineering design process.*

**Keywords:** *Design-by-Analogy, Engineering Design Process, Software.*

## 1. INTRODUÇÃO

O processo de design de novos produtos está diretamente relacionado ao crescimento econômico de qualquer organização e é fundamental para o seu sucesso. A pressão imposta pela alta competitividade do mercado e a exigência cada vez maior dos seus consumidores obriga as organizações a responderem às necessidades impostas com maior agilidade e alta produtividade através da inovação dos seus produtos. Além disso, o gerenciamento de equipes multidisciplinares, os avanços tecnológicos, a diminuição do ciclo de vida dos produtos e a responsabilidade ambiental são fatores que levam as empresas a rever e a aprimorar constantemente os seus processos de desenvolvimento de produto – PDP (LÖBACH, 2001; ROMEIRO FILHO, 2010).

Todas estas características configuram um ambiente extremamente complexo de ser gerenciado pelas organizações. Desta forma, a utilização de métodos intuitivos ou não estruturados de projeto não são mais tolerados pelo mercado, que exige o emprego de uma metodologia que suporte novos e elaborados conjuntos de procedimentos para o desenvolvimento de produtos, essencial para alavancar os processos de inovação (ROMEIRO FILHO, 2010). As metodologias organizam e sistematizam as inúmeras fases e atividades que ocorrem durante o processo de design de produto. A evolução de abordagens sistemáticas no PDP pode ser observada através das diversas propostas existentes na literatura, que, a partir do modelo de Pahl e Beitz (1988), deu origem a diversos estudos (HUNDAL, 1990; ROSENFELD et al., 2006; ULLMAN, 2010; ULRICH; EPPINGER, 2015), entre outros, a partir da segunda metade do século XX até os dias de hoje.

Desde a introdução do uso de computadores nas organizações, diferentes tipos de *softwares* começaram a ser desenvolvidos para agilizar algumas etapas do PDP. Estes sistemas geralmente auxiliam a equipe de projeto na execução de inúmeras tarefas, como por exemplo: simulações, análises, otimizações, armazenamento, combinação e recuperação de informações de projeto, entre outras. Porém, considerando todo este universo, o número de aplicações que apoiam a etapa de projeto conceitual, durante a geração de alternativas para o problema de projeto, é relativamente baixo (BRYANT, 2007; MORENO et al., 2014). Uma das razões para isso está no fato de que as informações disponíveis em estágios iniciais do ciclo de vida do projeto, muitas vezes, são insuficientes, o que gera obstáculos para estabelecer os parâmetros necessários para o processamento das informações em um sistema computacional (WANG et al., 2002).

O projeto conceitual é uma etapa que envolve o intenso uso da criatividade da equipe de projeto, o que aumenta a complexidade de desenvolver ferramentas que contribuam de forma efetiva para os processos mentais dos indivíduos. É um momento crítico para o sucesso do produto, onde o acesso ao conhecimento de design gerado e a experiência prévia dos projetistas são as bases para a construção do processo criativo no contexto de cada projeto. A busca por soluções inovadoras para os problemas de projeto exige a correlação entre ideias, conhecimentos e habilidades técnicas. O resultado deste processo tem consequências diretas nos aspectos econômicos, sociais e ecológicos do produto desenvolvido (PAHL; BEITZ, 1988).

Neste contexto, os processos cognitivos dos indivíduos podem ser bloqueados por um fenômeno inconsciente conhecido como fixação de ideias (*design fixation*) (JANSSON; SMITH, 1991). O sucesso de experiências anteriores e a adaptação de soluções antigas para o mesmo problema, fortalecem os efeitos da fixação de ideias e, como resultado, as soluções nascem de conceitos pré-concebidos, com

características de forma e função de projetos já existentes (LUCERO, 2014).

Diante disso, a utilização de analogias no design deu origem ao campo *Design-by-Analogy* (DbA) (LINSEY, 2007), que surge como uma abordagem alternativa para a geração de ideias. As pesquisas nesta área apresentam diversas técnicas e ferramentas para estimular a equipe de projeto a explorar as analogias sob vários aspectos na resolução de problemas de design e gerar inovação em suas soluções. Este é um possível caminho para combater e minimizar os efeitos do *design fixation* (MORENO et al., 2014).

Considerando o contexto apresentado, este artigo tem como objetivo formular uma proposta de modelo que será a base para o desenvolvimento de um novo instrumento que auxilie e estimule o processo criativo dos projetistas para a tomada de decisões de projeto e, conseqüentemente, contribua para os estudos relacionados na área de DbA. No caso desta pesquisa, este instrumento corresponde a um *software*, aplicado à etapa de geração de alternativas, onde o reuso do conhecimento em design permitirá a aplicação de soluções já desenvolvidas para resolver novos problemas de projeto, em diferentes contextos, potencializando a busca por combinações improváveis e o alcance de soluções inovadoras.

Para isso, o conhecimento científico do que já foi produzido na área de DbA e o levantamento dos *softwares* relacionados existentes formam o corpo teórico necessário para este artigo. Após esta revisão, é apresentada a metodologia desenvolvida para a condução deste trabalho. A seguir, a análise entre os *softwares* e os resultados gerados pelos seus estudos, a partir dos critérios estabelecidos na metodologia adotada, são compilados e apresentados, a fim de esclarecer os fundamentos utilizados para a construção do modelo proposto. Logo após esta análise, é descrito o modelo e seus elementos, bem como as considerações finais a respeito do seu processo de elaboração e avaliação.

## 2. DESIGN-BY-ANALOGY

Exemplos de produtos inovadores, baseados em analogias, podem ser observados na literatura científica e no mercado global. As analogias podem ocorrer a partir de artefatos já existentes, onde as ligações entre domínios próximos podem ser efetivas e mais óbvias, mas em geral, produzem soluções menos inovadoras. Por outro lado, ao estabelecer analogias entre domínios distantes, o projetista reusa o conhecimento existente para a resolução do problema em um novo contexto, ampliando as oportunidades de inovação. As analogias com a natureza são uma fonte inesgotável e inspiradora para a geração de soluções inovadoras. O sistema de ar-condicionado de grandes edificações, por exemplo, é baseado no modo de refrigeração dos ninhos de cupins. Já os estudos sobre o inseto vaga-lume deram origem às lâmpadas de LED, que brilham mais e consomem menos energia que as incandescentes (ASKNATURE.ORG, 2017; LINSEY, 2007).

A utilização das técnicas em DbA, no processo de design, apresenta uma abordagem efetiva para a geração de alternativas inovadoras. Porém, estudos realizados nesta área (LUCERO, 2014; MORENO et al., 2014; MURPHY et al., 2014; ORIAKHI; LINSEY; PENG, 2011) apontam a necessidade de sua aplicação em conjunto com ferramentas computacionais e repositórios de conhecimento em design. Esta necessidade se justifica devido à quantidade de combinações que devem ser realizadas para gerar os resultados de uma busca em uma extensa base de dados. Tal complexidade, somada a restrição de tempo no projeto, limita a realização desta atividade de forma manual. Neste sentido, a partir das necessidades de pesquisa dos projetistas, o desenvolvimento de *softwares* integrados a bases de conhecimento tem o potencial para

agilizar a execução deste trabalho, permitindo a recuperação de informações de projeto em diferentes contextos, potencializando o processo de busca por soluções inovadoras.

A sistematização das metodologias de design no PDP e, a partir disso, a origem de diversas técnicas estruturadas, facilita a manipulação das informações envolvidas em sistemas computacionais. Esta vantagem possibilita o desenvolvimento de sistemas baseados em conhecimento, onde as informações podem ser armazenadas e relacionadas permitindo a representação do conhecimento em bases de dados, para que este possa ser recuperado pelo usuário através de um *software* (PAHL; BEITZ, 1988; SZYKMAN; SRIRAM, 2006).

Durante o processo de design, especificamente no projeto conceitual, as técnicas relacionadas aos aspectos funcionais do produto auxiliam os projetistas na busca de soluções para o projeto. A modelagem funcional, por exemplo, é uma importante técnica utilizada na decomposição do problema de projeto e, ao mesmo tempo, um recurso para reduzir os efeitos do *design fixation*. Em DbA sua aplicação permite representar o conhecimento em design através da estrutura funcional do produto, permitindo que esta informação seja resgatada através de analogias das funcionalidades existentes entre os produtos. Porém, para que isto seja possível, o *software* deverá ser capaz de representar o conhecimento gerado, através da utilização de um vocabulário controlado que permita estabelecer relacionamentos coerentes entre as informações armazenadas (FU et al., 2014; PAHL; BEITZ, 1988).

Apesar do potencial do uso de analogias no design e das diferentes possibilidades de utilização de técnicas e métodos, parecem haver barreiras quanto a adoção durante o processo criativo (LINSEY; TEXAS; MARKMAN, 2008). Alguns dos desafios evidenciados são: identificar a etapa ideal do processo criativo para estimular as analogias, oferecer *softwares* e banco de dados adequados às necessidades do processo, adaptar sua aplicação ao tempo e ao orçamento do projeto e, ainda, satisfazer as expectativas dos profissionais envolvidos (TSENG et al., 2008).

Contudo, o DbA permite a pesquisa sistemática das informações de projeto disponíveis em diferentes fontes de dados, buscando conceitos análogos que podem ser reaproveitados para complementar e enriquecer o processo de geração de alternativas através da introdução de analogias não óbvias (TOMKO et al., 2015). O desenvolvimento de ferramentas digitais amparadas por algoritmos que explorem as técnicas de DbA, somado aos repositórios de conhecimento de soluções de projeto, deve considerar tanto os aspectos cognitivos do indivíduo quanto aspectos computacionais. Deste modo, esta prática é considerada um esforço combinado envolvendo o ser humano e o computador. Diante disso, a produção de *softwares* inteligentes para apoiar o processo de design deve proporcionar uma interação equilibrada entre sistema e usuário. Assim, uma implementação deste tipo deve buscar o aumento da produtividade e, ao mesmo tempo, estimular e potencializar a criatividade dos utilizadores (DEMIAN, 2004).

### 3. METODOLOGIA

Para atingir o objetivo deste trabalho foi realizado o levantamento do estado da arte das ferramentas computacionais existentes, relacionadas ao processo de design, na etapa de geração de alternativas, e que tenham no seu enfoque o DbA. O estudo de trabalhos relacionados serve como referência para identificar de que modo estas ferramentas contribuem para o processo projetual, quais suas principais limitações e as necessidades apontadas por estes estudos e que ainda não foram contempladas. Em seguida, a

partir deste levantamento e da análise dos resultados destes estudos, é proposto o modelo para o desenvolvimento de um *software* baseado em DbA. (Critérios de qualidade)

O presente estudo foi desenvolvido a partir da revisão da literatura científica, no primeiro semestre de 2017, a partir da busca realizada nas seguintes bases científicas digitais relevantes no campo do design: Periódicos Capes, SciELO, ScienceDirect e ASME Digital Collection. De acordo com os objetivos do trabalho, a busca teve como fim identificar iniciativas de ferramentas digitais em DbA associadas a repositórios de conhecimento aplicados ao contexto delimitado. Para isto, a expressão de busca definida e utilizada foi a palavra-chave "*software*" acompanhada dos seguintes termos: *Design by Analogy*, *Concept Generation*, *Design Knowledge Repositories* e *Design Synthesis*. Os mesmos termos foram utilizados no idioma português para os repositórios nacionais, porém não foram encontrados *softwares* que se enquadrem nas características avaliadas.

Os critérios de seleção dos estudos foram definidos para resgatar toda a publicação científica produzida que apresentasse, de fato, uma ferramenta computacional em DbA. Foram considerados, portanto, os resultados que produziram sistemas que utilizavam a analogia como estratégia para a etapa conceitual de design, ligadas a exploração de soluções para os problemas de projeto, derivando, assim, o conjunto de estudos analisados neste levantamento.

De acordo com a análise destes estudos e dos resultados evidenciados por seus experimentos, foram obtidas as características relacionadas de cada *software*. Para fins de comparação, estas características foram classificadas segundo objetivos das ferramentas e técnicas no contexto do DbA, e os atributos de qualidade de um *software*.

A NBR ISO/IEC 9126-1 é uma norma que estabelece critérios para avaliação da qualidade de *softwares* sob diferentes aspectos do artefato examinado: funcionalidade, confiabilidade, usabilidade, eficiência, manutenibilidade e portabilidade (ABNT, 2003). Logo, para este trabalho, foram utilizados alguns critérios desta norma para identificar as características positivas e negativas dos *softwares* analisados, relacionadas aos usuários e a satisfação das suas necessidades (funcionalidade) e a facilidade de interação (usabilidade) durante o uso do sistema.

Os critérios selecionados para a análise dos *softwares* revisados neste artigo foram: (i) o objetivo da ferramenta de DbA, (ii) a(s) técnica(s) de representação do conhecimento adotada(s), e (iii) as características de qualidade do *software* (NBR ISO/IEC 9126): funcionalidade e usabilidade. Os dados coletados para o item (iii) correspondem às informações das avaliações realizadas pelos autores dos trabalhos revisados, a partir do *feedback* dos utilizadores.

Após a revisão dos sistemas computacionais, o resultado da análise destes artefatos é apresentado através de um quadro comparativo, formulado a partir dos critérios de análise adotados, evidenciando as questões relevantes para a elaboração do modelo proposto. A análise e reflexão destas informações permitiu a definição de quais características deveriam ser consideradas para o desenvolvimento de um novo *software* em DbA.

A proposta de modelo para o *software* foi elaborada a partir do mapeamento destas características em requisitos que o sistema deve atender. Para isso, um esquema gráfico para o modelo foi utilizado para representar os seus elementos a partir de uma estrutura comum identificada na análise comparativa entre os *softwares* existentes em DbA, a saber: (i) interface gráfica: ponto de interação do usuário com o sistema, corresponde às entradas e saídas dos dados

funcionalidades e dados; (ii) funcionalidade: composta pelos algoritmos em DbA que estabelecem as regras de negócio, responsáveis por manipular e relacionar os dados estruturados; (iii) taxonomia: que determina como os dados são classificados para representar o conhecimento; e (iv) a base de dados: onde estes são estruturados e armazenados.

#### 4. REVISÃO DOS SOFTWARES EM DBA

A revisão da literatura identificou um conjunto de ferramentas digitais no contexto estudado (DbA) que, em sua maioria, correspondem a *softwares* associados à determinadas bases de conhecimento que, através de princípios analógicos, auxiliam na resolução de problemas de projeto a partir do conhecimento já gerado. Portanto, a revisão dos trabalhos relacionados consiste em listar as ferramentas digitais que se enquadram neste universo e que podem ser aplicadas na etapa de geração de alternativas e, desta forma, verificar quais as contribuições e limitações de cada *software* para o processo de projeto.

Os estudos de Chakrabarti et al (2011) e Verhaegen et al (2011) também realizaram um levantamento de sistemas com estas características. No contexto de DbA, Chakrabarti et al (2011) ressalta a relevância de sistemas baseados em analogia para as atividades de criação no design, e que este campo ainda carece de apoio computacional, devido à complexidade de recuperar as analogias em memória digital, área estudada pela Inteligência Artificial (IA). De maneira geral, os autores classificam os *softwares* voltados ao DbA a partir de três abordagens: (i) ferramentas de raciocínio baseado em casos, (ii) ferramentas baseadas em análise semântica, através do processamento da linguagem natural, e (iii) ferramentas de analogia baseadas na natureza, apoiadas pela biomimética e biônica.

##### 4.1 Softwares de raciocínio baseado em casos

As ferramentas que produzem analogias no design, compartilham a finalidade de identificar e transferir o conhecimento gerado por projetos anteriores, de um mesmo domínio ou diferentes, para solucionar problemas em novos projetos. Este exercício de abstração consiste em definir a estrutura de relações entre os elementos que compõe um problema de projeto, solução, domínio ou estratégia, e de que forma a transferência do conhecimento ocorre para atender uma determinada necessidade em um novo contexto. A tradução de determinadas técnicas de DbA para o ambiente digital resultam em modelos computacionais capazes de simular os processos humanos na tentativa de interpretar as informações manipuladas durante o processo de projeto (CHAKRABARTI et al., 2011; VERHAEGEN et al., 2011). Entre os *softwares* verificados que se enquadram nesta categoria estão: o MEMIC (BRYANT; STONE; MCADAMS, 2008), o *Design Repository* (BOHM; STONE; SZYKMAN, 2005) e o DRACULA (LUCERO, 2014). Além destas ocorrências, foram considerados neste levantamento os estudos IDEAL (BHATTA; GOEL, 1996) e Kritik2 (GOEL; BHATTA; STROULIA, 1997) que, apesar de não apresentarem uma implementação de *software* para estas propostas, são modelos computacionais que estabeleceram as primeiras definições para o desenvolvimento de *softwares* no campo do DbA.

Os modelos computacionais IDEAL (BHATTA; GOEL, 1996) e Kritik2 (GOEL; BHATTA; STROULIA, 1997) definiram padrões para explorar as analogias em uma base de produtos existentes para gerar automaticamente soluções em projetos para dispositivos físicos. A técnica estruturada de modelagem *Structure-Behavior-Function* (SBF) (BHATTA; GOEL, 1996) é utilizada para representar a decomposição funcional, estrutural e comportamental do produto, adotada pelos

autores como meio de armazenar e recuperar as informações de projeto em *softwares*. Desta forma, os sistemas recuperam soluções anteriores com base na SBF, permitindo a verificação dos projetos propostos e sugerindo modificações através da comparação de soluções semelhantes preexistentes em um banco de dados. No entanto, a codificação dos modelos SBF a partir de projetos existentes exige um esforço técnico e intelectual considerável por parte dos projetistas, aspecto que, muitas vezes, dificulta a execução da tarefa (VERHAEGEN et al., 2011).

O estudo de Bryant et al (2008) deu origem à ferramenta MEMIC (*Morphological Evaluation Machine and Interactive Conceptualizer*), um gerador de conceitos automatizado capaz de produzir alternativas de soluções de design para um projeto de produto a partir de subfunções de produtos existentes, obtidas do conhecimento armazenado do projeto *Design Repository* (BOHM; STONE; SZYKMAN, 2005), um repositório de design disponível na web. Para isto, o MEMIC exige que o usuário informe dois arquivos em um determinado formato – baseados em técnicas de modelagem funcional – descrevendo as funcionalidades e estrutura do produto em um alto nível de abstração através do vocabulário controlado oferecido pela *Functional Basis* (FB) – linguagem de design controlada criada para representar a estrutura funcional de um produto e facilitar a transmissão deste conhecimento (HIRTZ; STONE; MCADAMS, 2002). O algoritmo utiliza as relações função-componente contidas em uma matriz que correspondem a estrutura do repositório de produtos existentes e, conseqüentemente, compara e sugere alternativas para o problema de design.

Apesar de produzir resultados promissores para a geração de conceitos, as observações realizadas durante o experimento apontaram que uma parcela dos participantes se sentiram desconfortáveis com a informação gerada pelo *software*. Isto ocorreu devido à complexidade em compreender os resultados textuais gerados pela ferramenta. Este desconforto em interpretar o retorno do *software* demonstrou a necessidade de melhorias na apresentação dos resultados, como por exemplo, adicionando recursos de imagens ou esquemas visuais (BRYANT; STONE; MCADAMS, 2008).

O projeto *Design Repository* (BOHM; STONE; SZYKMAN, 2005), apesar de ser uma iniciativa que somente possui uma base de conhecimento em design, é uma das poucas ferramentas que oferece uma interface web que permite explorar o repositório. O acesso a esta interface é público e permite o utilizador explorar e extrair as informações da base de dados de produtos indexados pelos seus criadores. Até o momento desta pesquisa (junho de 2017), o repositório armazena 175 sistemas (*systems*) e 6906 componentes (*artifacts*) – que relacionados compõe um sistema – indexados pelas suas características e funcionalidades (FB) permitindo a navegação entre as estruturas dos produtos existentes.

Porém, Bohm (2009) destaca a necessidade de desenvolver uma interface de integração para que outros sistemas possam acessar diretamente os dados do repositório e, assim, proporcionar um repositório de design centralizado para novas ferramentas digitais de design. Outro ponto observado, para o utilizador, foi a necessidade de um editor gráfico que permita a manipulação dos modelos funcionais do produto durante a indexação e, deste modo, facilitar a interação e visualização das informações de projeto.

Reconhecendo as potencialidades do DbA para a construção de ferramentas digitais, a pesquisa de Lucero (2014) desenvolveu o *software* chamado DRACULA (*Design Repository & Analogy Computation via Unit-Language Analysis*) que uniu o repositório de conhecimento em design

existente (*Design Repository*) de Bohm (2009) associado a um sistema com algoritmos que mapeiam as relações analógicas a partir de métricas de desempenho de engenharia e funcionalidades do produto. Através de uma interface, o usuário seleciona um único parâmetro de desempenho de engenharia associado a um conjunto sequencial de até três funções críticas do produto.

A seguir, o *software* recebe os parâmetros, executa a busca e retorna ao usuário somente as analogias encontradas em seu repositório, sem informações ou detalhes adicionais, cabendo ao projetista interpretar o resultado e compreender em que contexto o mesmo pode ser aplicado ao seu problema de projeto.

Este estudo apresentou um grande potencial em resgatar analogias relevantes aos problemas de projeto através de parâmetros de desempenho de engenharia e funcionalidades do produto, porém as pesquisas com experimentos mais detalhados estão em andamento e ainda devem ser publicadas pelos autores (TOMKO et al., 2015).

#### 4.2 Softwares de análise semântica

Para melhorar a eficiência de sistemas que utilizam a modelagem funcional ao estabelecer as relações entre os produtos, a análise semântica – através de algoritmos para o processamento de linguagem natural – é utilizada em *softwares* com o objetivo de aproximar o processamento da máquina da compreensão da língua humana, de acordo com um conjunto de regras e associações (MURPHY et al., 2014). Entre os *softwares* analisados que utilizam esta técnica estão o sistema de busca de analogias em patentes produzido por Murphy et al (2014) e a ferramenta *WordTree Express* (WTE) (ORIAKHI; LINSEY; PENG, 2011).

O sistema de busca de analogias em patentes (MURPHY et al., 2014), por exemplo, visa facilitar a descoberta de analogias no design através da extensa base de patentes americana (*United States Patent and Trademark Office*). A estratégia adotada pelo estudo, utiliza uma taxonomia que relaciona os verbos de funções existentes nas patentes com as funções mapeadas na *Functional Basis*, promovendo uma extensão da linguagem. A construção destes relacionamentos de linguagem proposta na taxonomia é realizado com o auxílio da base de dados lexical *WordNet* (MILLER, 1995), que agrupa as palavras em conjuntos de sinônimos chamados *synsets*, interligados por meio de relações conceituais-semânticas e lexicais auxiliando no processamento da linguagem natural.

Segundo o autor, o resultado do estudo mostrou que as patentes foram consideradas fontes de analogias e conceitos que contribuem positivamente para o processo de desenvolvimento de soluções inovadoras.

Nesta mesma linha de abordagem semântica, a ferramenta *WordTree Express* (WTE) (ORIAKHI; LINSEY; PENG, 2011) foi desenvolvida para agilizar e automatizar as etapas do *WordTree* (LINSEY; TEXAS; MARKMAN, 2008), método utilizado para estimular o uso de analogias no processo de design. Os diagramas hierárquicos em árvore gerados pela WTE são criados a partir da combinação das bases de dados *WordNet* e WTE, em poucos segundos, agilizando o processo manual que antes despendia um esforço de cerca de trinta minutos do usuário. Ao utilizar a ferramenta *WordTree Express*, o usuário insere uma função como palavra-chave (um verbo) e aciona o botão de pesquisa para verificar no repositório os diferentes sentidos do termo inserido. O usuário então seleciona o sentido da função, entre os registros retornados da base de dados, que melhor corresponde ao problema de projeto. Em seguida, o *software* gera um arquivo com o diagrama *WordTree* – aberto em um programa externo – que apresenta a árvore de funções e suas relações

hierárquicas permitindo ao utilizador a análise e descoberta de analogias.

A utilização do método com a ferramenta demonstrou uma melhora significativa na descoberta de analogias. As melhorias propostas por este estudo incluem: a integração de outros bancos de dados que possam enriquecer a base de funções e produtos; a melhoria da interface do usuário para implementar todas as funcionalidades em um único ambiente, eliminando a necessidade de *softwares* terceiros; e a combinação de palavras com imagens para estimular visualmente o processo de recuperação de analogias (ORIAKHI; LINSEY; PENG, 2011).

#### 4.3 Softwares de inspiração na natureza

O desenvolvimento de ferramentas digitais que enfocam as questões da natureza para resolver problemas de projeto é um tema recente e em expansão relacionado à produções científicas (GOEL; MCADAMS; STONE, 2014). Estes estudos destacam que a natureza pode ser considerada uma das maiores fontes de soluções adaptáveis para o contexto de projeto com vistas a resultados sustentáveis e inovadores. O desenvolvimento de modelos computacionais auxiliam na manipulação do grande volume de informações e dos possíveis mapeamentos para estabelecer as analogias entre os distintos domínios (GOEL, 2015). Os sistemas identificados com este propósito são o *Idea-Inspire* (CHAKRABARTI et al., 2005), o AskNature (ASKNATURE.ORG, 2017) e o DANE (GOEL, 2012).

O *software Idea-Inspire* foi o primeiro repositório computacional de sistemas biológicos desenvolvido para apoiar os processos de projeto a partir da biomimética no design. Desenvolvida por Chakrabarti et al (2005), a ferramenta auxilia na resolução de problemas de projeto utilizando, como fonte de inspiração, sistemas naturais e artificiais armazenados em um banco de dados. Cada sistema é descrito e indexado na base de dados através de dois modelos de representações: o SBF e o SAPPiRE, juntamente com imagens e vídeos. O modelo SAPPiRE (*State change, Action, Parts, Phenomenon, Input, oRgan and Effect*) consiste em uma linguagem desenvolvida para descrever o funcionamento de sistemas naturais e artificiais (CHAKRABARTI et al., 2017). O *Idea-Inspire* foi a primeira ferramenta que permitiu ao usuário incluir novos sistemas no banco de dados.

O experimento realizado com designers e alunos apontou que o *software* foi capaz de estimular os usuários a produzir, em geral, o dobro de ideias para um problema de projeto. Além disso, foi observado que o número de ideias exploradas e estimuladas pela ferramenta se correlaciona fortemente com a variedade das soluções geradas nos projetos analisados. Apesar dos bons resultados, o *software* é executado de forma isolada no computador do cliente, consequentemente as soluções de design incluídas pelo usuário não podem ser reutilizadas por outros usuários. Por isso, a nova versão – *Idea-Inspire 3.0* – que está em desenvolvimento, representa a evolução do sistema atual para a plataforma *web*, proporcionando um ambiente colaborativo para o compartilhamento do conhecimento de projetos entre os seus utilizadores. Também, a nova versão deve aprimorar as funcionalidades, tais como a integração da base de dados lexical *Wordnet* para melhorar o processamento da linguagem natural do sistema (CHAKRABARTI et al., 2017; GOEL; MCADAMS; STONE, 2014).

O projeto AskNature (ASKNATURE.ORG, 2017) é uma plataforma *web* resultante do esforço da organização *The Biomimicry Institute* (TBI) e sua comunidade em tornar informações biológicas acessíveis e aplicáveis à contextos não

biológicos, servindo como fonte de inspiração para o campo da Biomimética e Biônica.

O acesso ao *software* e banco de dados é gratuito e permite o usuário realizar consultas às diversas soluções e estratégias biológicas utilizadas pela natureza para superar os desafios naturais. As informações indexadas são obtidas de publicações científicas e avaliadas por revisão em pares antes de serem catalogadas no sistema. As relações estabelecidas entre os mecanismos, em um alto nível de abstração e apoiadas pela taxonomia *Biomimicry Taxonomy* – sistema de classificação para organizar o conteúdo biológico –, utilizam as funções como estratégia para estabelecer o elo de ligação entre a biologia e campos como a engenharia, a arquitetura, o design industrial, a robótica, entre outros.

Entretanto, o *software* apresenta ao usuário as soluções em um alto nível de abstração, com poucos detalhes técnicos de aplicação em projeto. Além disso, os usuários recuperaram as informações através de pesquisas tradicionais baseadas em palavras-chave, além de permitir explorar e navegar na base de conhecimento através da taxonomia adotada. Apesar de prover acesso público pela web, o *AskNature* não oferece a funcionalidade para que os usuários adicionem novos sistemas ao banco de dados. Todavia, o usuário pode solicitar a permissão para se tornar um colaborador, mediante avaliação e aprovação dos organizadores do projeto (ASKNATURE.ORG, 2017; GOEL, 2015; THE BIOMIMICRY INSTITUTE, 2008).

Semelhante ao *Idea-Inspire*, o *software* DANE (GOEL, 2012) é outra ferramenta computacional que explora o estímulo de soluções biológicas para a resolução de problemas de projeto. As soluções de projeto são modeladas a partir da modelagem SBF e armazenadas no banco de dados, formando um conjunto de sistemas conectados entre si e com diferentes níveis de complexidade. Da mesma forma que o *Idea-Inspire*, o DANE é um *software* executado no cliente e oferece a possibilidade de inclusão de novas soluções de design pelos seus usuários.

No entanto, o experimento realizado com os usuários demonstrou que diversos sistemas estavam representados de forma incompleta na aplicação. Uma das razões identificadas para tal inconformidade apontou que a modelagem SBF foi realizada por um grupo de alunos com pouca ou nenhuma experiência em sistemas biológicos, assim como a incapacidade dos participantes de modelar adequadamente o conhecimento de design através dos modelos SBF. Outro fator, este ligado ao nível de detalhamento das soluções, foram as próprias limitações da modelagem utilizada, onde um modelo hierárquico pode não ser suficiente para representar um sistema biológico altamente complexo, no qual os eventos podem ocorrer sequencialmente e simultaneamente em vários níveis sistêmicos para alcançar uma funcionalidade pretendida (GOEL, 2012; GOEL; MCADAMS; STONE, 2014).

## 5. ANÁLISE DOS SOFTWARES REVISADOS

De modo geral, todos os *softwares* em DbA analisados compartilham elementos comuns que compõem estes sistemas: uma Interface Gráfica do Usuário (GUI) que permite a interação com o sistema, um vocabulário controlado para a aplicação de algoritmos baseados em DbA e a representação do conhecimento e uma base de dados. Cabe ressaltar que, no caso das ferramentas analisadas, os algoritmos implementados sob uma determinada abordagem DbA, em sua totalidade, utilizam a analogia das funcionalidades de um determinado produto ou sistema.

Além disso, esta revisão permite compreender sob quais aspectos foram desenvolvidas as primeiras ferramentas digitais em DbA e de que forma os estudos realizados nesta área vêm sendo conduzidos até o presente momento. As

pesquisas científicas que deram origem à *Functional Basis* e ao *Design Repository* (STONE; WOOD, 2001; SZYKMAN et al., 2000) proporcionaram novos estudos e aplicações em outras ferramentas computacionais relacionadas ao DbA (BOHM; STONE; SZYKMAN, 2005; BRYANT; STONE; MCADAMS, 2008; LUCERO, 2014; MURPHY et al., 2014). Estes trabalhos consistem em esforços de vários anos de estudos por grupos de pesquisa, e servem como base para a construção de novos *softwares* neste contexto, permitindo a sua combinação com outras implementações para produzir ferramentas digitais inéditas.

Para auxiliar na análise das características dos *softwares* revisados, o Quadro 1 sintetiza os pontos relevantes observados de cada *software*, com o objetivo de identificar os requisitos necessários para a elaboração do modelo proposto neste estudo. O quadro foi elaborado conforme os critérios estabelecidos na metodologia deste artigo e as ocorrências listadas em ordem crescente da data de publicação.

Os resultados dos estudos selecionados comprovaram as potencialidades das analogias no processo de design e o fator decisivo que os *softwares* podem exercer como instrumentos de tomada de decisão em projetos, no apoio aos projetistas na geração de conceitos e soluções inovadoras. Porém, o desafio observado está no equilíbrio de implementar *softwares* que se integrem de forma natural às atividades do usuário, na etapa de geração de alternativas, no processo de design. Isso implica no aprofundamento da compreensão das necessidades do usuário/projetista e do contexto das atividades desenvolvidas nas etapas projetuais de intensa criatividade do designer, para que, deste modo, seja possível o desenvolvimento de sistemas cada vez mais inteligentes que se aproximem da linguagem e dos processos mentais humanos.

Em relação aos fatores de usabilidade, as Interfaces Gráficas do Usuário analisadas apresentaram deficiências. Os *softwares* que utilizam somente representações textuais dificultam a interpretação dos dados nos casos aplicados. Para as atividades apoiadas por tais ferramentas, a utilização de imagens, vídeos e outros esquemas visuais na interface exercem uma influência positiva para os processos de analogia (ORIAKHI; LINSEY; PENG, 2011). Outro fator importante para a melhoria da interação e experiência de uso do sistema é oferecer diferentes possibilidades para o usuário explorar as informações da base de conhecimento, estabelecendo mecanismos de navegação coerentes aos processos cognitivos do indivíduo para a produção de analogias. Porém, para que estes recursos de interação possam ser melhor explorados, a participação do usuário é essencial durante o projeto da interface gráfica. A compreensão das necessidades do usuário e o seu envolvimento oferecem condições para projetar interfaces que alcancem efetivamente os objetivos pretendidos pelo *software* (PREECE; ROGERS; SHARP, 2013).

Entre as limitações apontadas pelos experimentos das ferramentas descritas, de modo geral, um dos aspectos percebidos está relacionado à qualidade dos resultados que os *softwares* retornam a partir dos requisitos de projeto informados pelo usuário que, em determinados casos, foi constatado que poderiam ser mais eficientes. Geralmente, esta ocorrência é associada aos aspectos de funcionalidade, responsável pela aplicação dos algoritmos que proporcionam a inteligência do *software* e estabelecem as regras de funcionamento, processando as entradas e saídas de dados entre o usuário e sistema. Por isso, as questões relacionadas à semântica são um fator decisivo na implementação de sistemas mais intuitivos, ou seja, capazes de melhor interpretar a linguagem humana. A base de dados lexical *WordNet*, por exemplo, pode ser utilizada para ampliar a potencialidade do *software* em compreender as solicitações

do usuário através do seu repositório de sinônimos e suas relações semânticas (FU et al., 2014; ORIAKHI; LINSEY; PENG, 2011).

Outra questão observada pelos participantes nos estudos de Bohm (2003) e Goel (2012), foi o nível elevado de complexidade para representar as soluções na base de conhecimento, geralmente ocasionado pela pouca experiência

dos usuários na modelagem funcional, atividade fundamental que, se mal desempenhada, compromete a eficiência da recuperação do conhecimento nestas bases. Inclusive para os profissionais com maior experiência, a modelagem funcional não é uma tarefa trivial, exige conhecimento e habilidade durante a aplicação. Mesmo assim, é considerada uma das principais técnicas para comunicar o conhecimento em design.

**Quadro 1:** Análise dos softwares em DbA

<b>Software</b>	<b>Representação do conhecimento</b>	<b>Objetivo</b>	<b>Características de qualidade</b> <input checked="" type="checkbox"/> <input type="checkbox"/>
<b>IDEAL*</b> (BHATTA; GOEL, 1996)	Modelagem SBF	Sugerir melhorias para o projeto de design com base no banco de soluções existente	<b>Funcionalidade</b> <input checked="" type="checkbox"/> Automatiza tarefas de design <input type="checkbox"/> Dificuldade na modelagem SBF
<b>Kitric2 *</b> (GOEL; BHATTA; STROULIA, 1997)			<b>Usabilidade</b> Não apresentou
<b>Design Repository</b> (BOHM; STONE; SZYKMAN, 2005)	<i>Functional Basis</i>	Repositório compartilhado de conhecimento em design	<b>Funcionalidade</b> <input checked="" type="checkbox"/> Modelo robusto de repositório de conhecimento em design <input checked="" type="checkbox"/> Ambiente compartilhado <input type="checkbox"/> Não permite inserir soluções <input type="checkbox"/> Não oferece mecanismos de busca <b>Usabilidade</b> <input checked="" type="checkbox"/> Permite navegação na estrutura do repositório <input checked="" type="checkbox"/> Utiliza imagens e modelos para representação de produtos
<b>Idea-Inspire</b> (CHAKRABARTI et al., 2005)	Modelagem SBF e SAPPHiRE	Explorar estratégias de solução inspirada nas analogias da natureza	<b>Funcionalidade</b> <input checked="" type="checkbox"/> Permite a busca por modelos funcionais <input type="checkbox"/> Não permite inserir soluções <b>Usabilidade</b> <input checked="" type="checkbox"/> Utiliza imagens e vídeos
<b>MEMIC</b> (BRYANT; STONE; MCADAMS, 2008)	<i>Functional Basis</i>	Sugerir alternativas de solução para o problema de projeto	<b>Funcionalidade</b> <input checked="" type="checkbox"/> Automatiza a geração de alternativas para o projeto conceitual <input type="checkbox"/> Não permite inserir soluções <b>Usabilidade</b> <input type="checkbox"/> Dificuldade de interpretação dos resultados (somente textual)
<b>Asknature</b> (THE BIOMIMICRY INSTITUTE, 2008)	<i>Taxonomia Biomimicry</i>	Explorar estratégias de solução inspiradas em analogias da natureza	<b>Funcionalidade</b> <input checked="" type="checkbox"/> Ambiente compartilhado <input type="checkbox"/> Nível de abstração para as soluções é alto (único) <input type="checkbox"/> Não permite inserir soluções <b>Usabilidade</b> <input checked="" type="checkbox"/> Interface permite explorar a taxonomia e realizar buscas
<b>WordTree Express</b> (ORIAKHI; LINSEY; PENG, 2011)	Árvores hierárquicas e <i>Wordnet</i>	Estimular o aumento do número de analogias para o método <i>WordTree</i>	<b>Funcionalidade</b> <input checked="" type="checkbox"/> Utiliza recursos semânticos <input type="checkbox"/> Informação somente textual <input type="checkbox"/> Não tem banco de dados <b>Usabilidade</b> <input type="checkbox"/> Necessita acesso de <i>softwares</i> terceiros para entregar o resultado
<b>DANE</b> (GOEL, 2012)	Modelagem SBF	Explorar estratégias de solução inspiradas em analogias da natureza	<b>Funcionalidade</b> <input checked="" type="checkbox"/> Informações detalhadas sobre as analogias da natureza relaciona o design <input type="checkbox"/> Dificuldade na modelagem SBF <b>Usabilidade</b> Não apresentou
<b>DRACULA</b> (LUCERO, 2014)	<i>Functional Basis</i>	Sugerir analogias com base nas funções críticas e em um parâmetro de engenharia	<b>Funcionalidade</b> <input checked="" type="checkbox"/> Utiliza estratégias de analogia considerando parâmetros de engenharia <input type="checkbox"/> Não informa como utilizar a analogia <b>Usabilidade</b> <input type="checkbox"/> Informação somente textual
<b>Patent Search</b> (MURPHY et al., 2014)	<i>Functional Basis, Wordnet e taxonomia UPSTO</i>	Sugerir soluções para o problema de projeto com base na analogia funcional de patentes	<b>Funcionalidade</b> <input checked="" type="checkbox"/> Utiliza uma base expressiva de patentes <b>Usabilidade</b> Não apresentou

\* Modelos computacionais

Elaborado pelos autores.

Portanto, para facilitar a representação deste conhecimento em meio digital, os *softwares* devem proporcionar diferentes níveis de abstração para armazenar as informações de projeto e, ao mesmo tempo, oferecer alternativas distintas de acesso ao conhecimento. O desafio consiste em viabilizar a indexação e a recuperação destas informações no ambiente digital.

Outro ponto, pouco explorado pelas ferramentas existentes, é a possibilidade de promover ambientes colaborativos que acelerem a produção e a troca de conhecimento entre os usuários. A *web*, por exemplo, proporciona a infraestrutura necessária para a implementação de ambientes deste tipo. Esta característica está diretamente relacionada com o ambiente que o *software* foi programado para ser executado. Entre os *softwares* analisados somente o *AskNature.org* e o *Design Repository* executam na *web*, porém, ainda, não permitem a inclusão de novas soluções de projetos pelos seus usuários. Entretanto, a nova versão do *software Idea-Inspire*, em desenvolvimento – verificado em julho de 2017 –, tem o objetivo de criar uma rede de compartilhamento de conhecimento, onde os usuários

possam pesquisar, desenvolver e armazenar novas ideias de projeto em sua base dados e, conseqüentemente, transferir o conhecimento gerado entre os projetistas (CHAKRABARTI et al., 2017).

Porém, a possibilidade do livre acesso e manipulação das informações pelos usuários do *software* requer o estudo de mecanismos que garantam a integridade e a consistência da base de conhecimento, sem comprometer o propósito dos ambientes colaborativos que visam a transferência do conhecimento gerado pelos seus usuários.

Portanto, as ferramentas digitais geradas ao longo do tempo e seus respectivos estudos permitem a identificação das tendências e oportunidades no desenvolvimento de novos *softwares* no campo do DbA, expondo lacunas que possibilitam explorar alternativas para minimizar as restrições dos *softwares* existentes através de novas abordagens, sem desconsiderar o conhecimento adquirido até o momento. Esta reflexão e análise permitem a seleção de determinadas características apontadas pelos estudos, fornecendo informações para o mapeamento dos requisitos necessários para a elaboração do modelo proposto neste artigo (Figura 1).

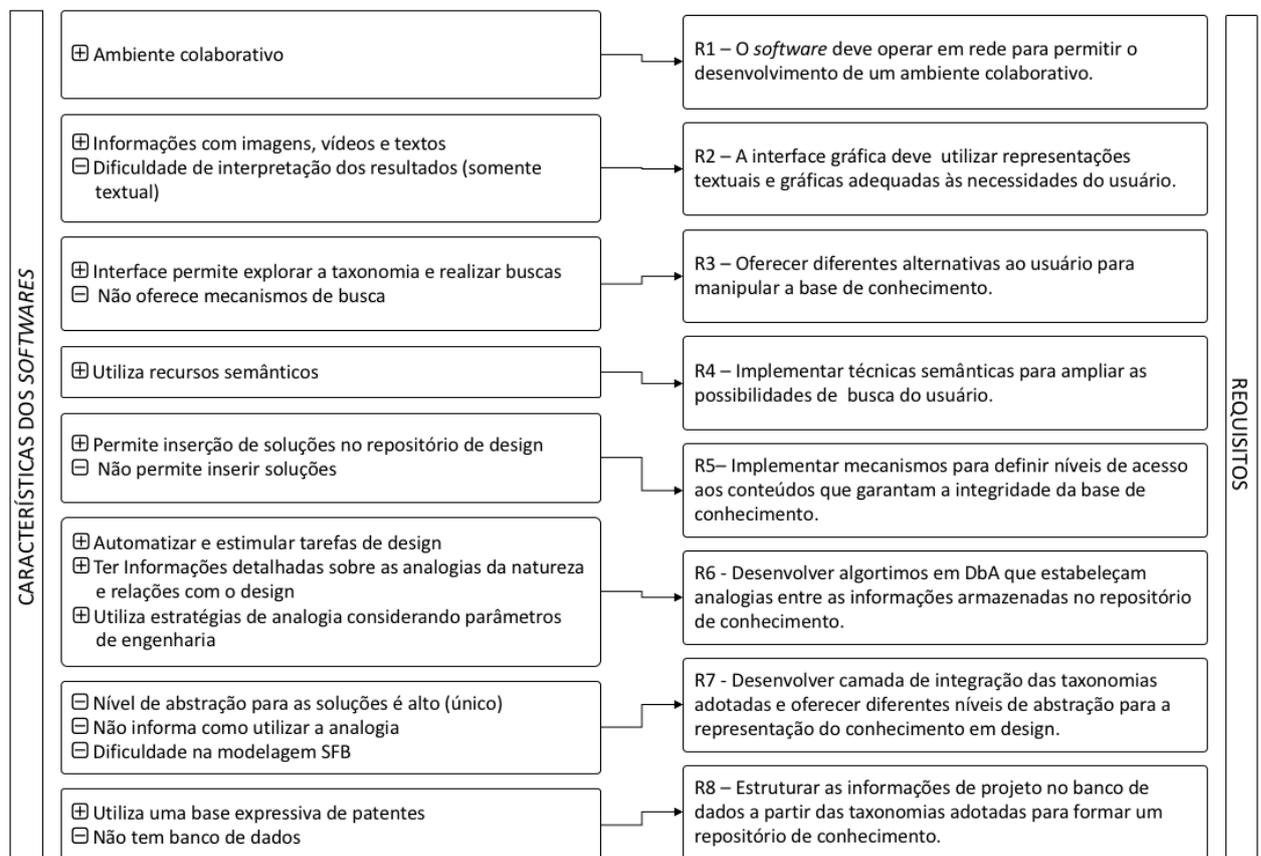


Figura 1: Mapeamento dos requisitos. Elaborado pelos autores.

## 6. PROPOSTA DE UM MODELO DE SOFTWARE EM DBA

De acordo com o contexto investigado e os resultados expostos pelos estudos dos *softwares* revisados, este trabalho considera os aspectos evidenciados, tanto positivos quanto negativos, para propor um modelo de desenvolvimento para um *software* em DbA.

Este estudo será a base para o desenvolvimento de uma ferramenta computacional que integra os princípios em DbA e preenche as lacunas identificadas nesta revisão. Para isso, os elementos observados que compõem o modelo proposto são: (i) uma interface gráfica do usuário que permita consultar e explorar a base de conhecimento; (ii) algoritmos baseados nas

técnicas de DbA que permitam o acesso ao conhecimento em diferentes níveis de abstração e complexidade; (iii) as taxonomias adotadas para estabelecer as relações entre os conceitos; (iv) e a base estruturada de dados.

A partir desta configuração, o modelo proposto, observado na Figura 2, descreve os elementos que devem atender ao conjunto de requisitos identificados. O modelo, portanto, explora as potencialidades de cada categoria dos *softwares* em DbA revisados, assim como utiliza a combinação de parte do conhecimento gerado neste campo para gerar uma proposta que possibilite o desenvolvimento de um novo instrumento de apoio ao processo de design, na etapa de projeto conceitual.

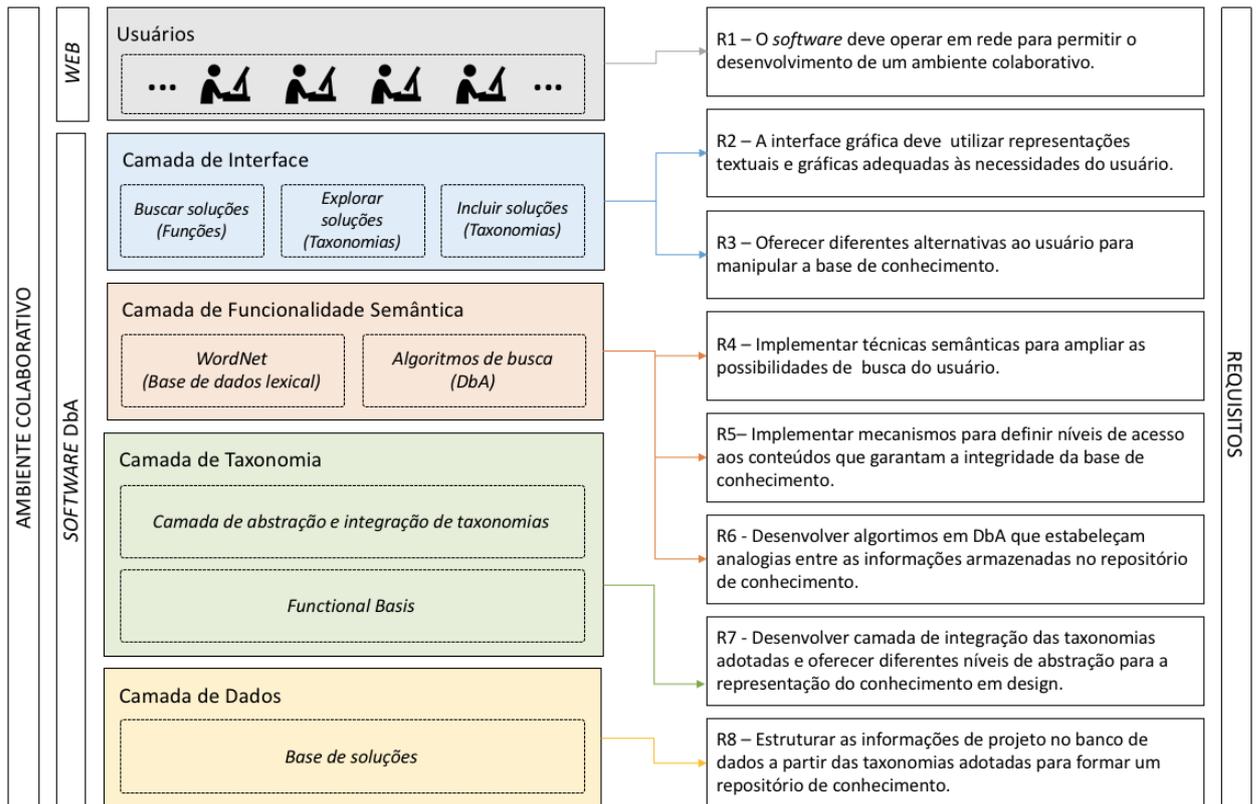


Figura 2: Modelo proposto para o software e requisitos relacionados. Elaborado pelos autores.

Para atender aos requisitos de acesso ao sistema e ao compartilhamento do conhecimento em um ambiente colaborativo, o software deve ser implementado na plataforma web. Isto porque, além de oferecer uma fonte para a descoberta de analogias no design, o sistema também deve registrar o conhecimento gerado pelos seus usuários e promover formas de compartilhamento entre eles. Durante o uso, por exemplo, a medida que o designer estabelece uma nova analogia, esta pode ser armazenada e, conseqüentemente, descoberta e reutilizada por um outro usuário em um novo contexto de projeto, e assim sucessivamente, gerando novos conhecimentos e expandindo o repositório de design.

Conforme analisado no item anterior, o projeto da interface gráfica deve atender as expectativas do usuário durante a sua interação com o sistema para explorar as analogias no repositório de conhecimento. Para aprimorar esta tarefa, além dos recursos gráficos bem empregados, a utilização da base de dados lexical do projeto *WordNet* combinada aos algoritmos de busca em DbA podem melhorar a interpretação das solicitações do usuário através da semântica.

Para representar o conhecimento em meio digital, a taxonomia *Functional Basis* (HIRTZ; STONE; MCADAMS, 2002) demonstrou, através da continuidade de seus estudos e da sua aplicação nos softwares em DbA analisados (BOHM; STONE; SZYKMAN, 2005; BRYANT; STONE; MCADAMS, 2008; LUCERO, 2014; MURPHY et al., 2014), ser o vocabulário mais adequado para classificar os produtos ou sistemas em um repositório de conhecimento em design, a partir das suas funcionalidades.

Entretanto, para que a combinação de todos estes elementos atenda aos requisitos da proposta, é necessária uma camada intermediária, responsável por conectar a interface gráfica do usuário com a taxonomia adotada. Esta camada consiste em estabelecer as conexões dos recursos

semânticos e de linguagem natural com a taxonomia *Functional Basis*. E com isso, ampliar a capacidade do sistema de interpretar e processar as solicitações do usuário, gerando novas formas de consulta e interação dos projetistas com o software na busca por analogias, em diferentes contextos, na base de conhecimento.

Portanto, a continuação deste estudo consiste no refinamento do modelo proposto para sua posterior validação. O estudo e detalhamento de cada elemento sugerido pelo modelo, a fim de refinar os requisitos, seguido da validação por especialistas que atuam no processo de design, irão permitir confirmar a viabilidade do modelo para a implementação do software.

## 7. CONSIDERAÇÕES FINAIS

O estudo realizado demonstra um amplo campo de oportunidades para o desenvolvimento de novas ferramentas digitais que apoiem o processo de design. No escopo do DbA, especificamente, estas ferramentas são projetadas para estimular o designer na identificação de analogias, através do reuso do conhecimento em design aplicado em diferentes contextos. Esta abordagem, surge como alternativa promissora para a área do design, permitindo explorar novos caminhos no desenvolvimento de soluções realmente inovadoras, circunstância recorrente e fundamental para o projeto conceitual de produtos.

O campo da Computação, por sua vez, fornece o aporte necessário para atender o desenvolvimento destes softwares que, diante do progresso tecnológico e do avanço no campo da IA, constantemente ampliam as possibilidades de aplicação de novos recursos no desenvolvimento de sistemas mais inteligentes. Esta condição permite a aplicação de novas técnicas computacionais, que podem auxiliar a superar as limitações observadas nos sistemas existentes analisados neste estudo.

Entre estes *softwares*, além das limitações já identificadas, o levantamento realizado permitiu o mapeamento das técnicas e abordagens utilizadas em DbA e expôs como estas foram adaptadas para o uso em sistemas computacionais. Ademais, a revisão da literatura reuniu os conhecimentos necessários restantes para a identificação do conjunto de requisitos para este estudo.

O resultado deste trabalho considerou todos os temas abordados para a elaboração do modelo proposto, onde a combinação de técnicas já existentes somada a novos componentes que explorem caminhos alternativos promovam avanços para o campo do DbA.

A continuidade deste estudo prevê a validação e o refinamento das especificações deste modelo, para então, dar início ao desenvolvimento do *software*. Nesta etapa, a participação de especialistas da área do design – usuários da futura ferramenta – será fundamental durante as atividades de avaliação e detalhamento do modelo. Para mapear e conduzir este processo, os autores deste trabalho deverão adotar uma metodologia adequada à execução das etapas necessárias durante a pesquisa para a projetar e avaliar um novo artefato.

Outro trabalho futuro relevante para a continuação deste estudo, na fase de desenvolvimento do *software*, será o aprofundamento das técnicas de IA que poderão ser aplicadas a este projeto. Isto porque esta área da Computação é a responsável pelo estudo e desenvolvimento de sistemas capazes de interpretar e relacionar informações, características diretamente relacionadas com as necessidades do modelo em questão. O uso da linguagem natural no processo de busca do usuário e a geração de analogias com base na relação das informações do repositório digital, funcionalidades pretendidas para o *software* planejado, exigem a implementação de algoritmos que utilizem estas estratégias para atingir os objetivos propostos do *software*.

A partir deste caminho, o conjunto destes estudos deve formar a base necessária para o desenvolvimento de uma ferramenta computacional capaz de registrar e reaproveitar o conhecimento de design gerado. E ainda, acrescente novas possibilidades que darão continuidade aos estudos em ferramentas computacionais de apoio ao processo de design.

## REFERÊNCIAS

- [1]. ABNT. **NBR ISO/IEC 9126-1: Engenharia de software – qualidade de produto. Parte 1: modelo de qualidade**. São Paulo: Associação Brasileira de Normas Técnicas, 2003.
- [2]. ASKNATURE.ORG. **About AskNature**. Disponível em: <<https://asknature.org/about-asknature>>. Acesso em: 1 maio. 2017.
- [3]. BHATTA, S. R.; GOEL, A. K. From design experiences to generic mechanisms: Model-based learning in analogical design. **Artificial Intelligence for Engineering Design, Analysis and Manufacturing: AIEDAM**, v. 10, n. 2, p. 131–136, 1996.
- [4]. BOHM, M. R. **Information archival and reuse : drawing conclusions from the past**. Doctoral Dissertations—Columbia, Missouri: Missouri University Of Science & Technology, 2009.
- [5]. BOHM, M. R.; STONE, R. B.; SZYKMAN, S. Enhancing Virtual Product Representations for Advanced Design Repository Systems. **Journal of Computing and Information Science in Engineering**, v. 5, n. 4, p. 360, 2005.
- [6]. BRYANT, C. R. **A Computational Theory for the Generation of Solutions During Early Conceptual Design**. Columbia, Missouri: University of Missouri-Rolla, 2007.
- [7]. BRYANT, C. R.; STONE, R. B.; MCADAMS, D. A. Memic: An interactive morphological matrix tool for automated concept generation. **IIE Annual Conference and Expo 2008**, p. 1196–1201, 2008.
- [8]. CHAKRABARTI, A. et al. A functional representation for aiding biomimetic and artificial inspiration of new ideas. **AIE EDAM**, v. 19, n. 2, p. 113–132, 2005.
- [9]. CHAKRABARTI, A. et al. Computer-Based Design Synthesis Research: An Overview. **Journal of Computing and Information Science in Engineering**, v. 11, n. 2, p. 21003, 2011.
- [10]. CHAKRABARTI, A. et al. **Idea Inspire 3.0 – A Tool For Analogical Design**. Bengaluru, India: Centre for Product Design and Manufacturing, Indian Institute of Science, 2017.
- [11]. DEMIAN, P. **CoMem: Design Knowledge Reuse from a Corporate Memory**. Standford: Standford University, 2004.
- [12]. FU, K. et al. Design-by-analogy: experimental evaluation of a functional analogy search methodology for concept generation improvement. **Research in Engineering Design**, v. 26, n. 1, p. 77–95, 2014.
- [13]. GOEL, A. **DANE: Design by Analogy to Nature Engine**. Disponível em: <<http://dilab.cc.gatech.edu/dane/>>. Acesso em: 1 maio. 2017.
- [14]. GOEL, A. K. Biologically inspired design: A New Paradigm for AI Research on Computational Sustainability? **IEEE Intelligent Systems**, v. 28, n. 3, p. 80–84, 2015.
- [15]. GOEL, A. K.; BHATTA, S. R.; STROULIA, E. **Kritik: An Early Case-Based Design System**. Atlanta: Georgia Institute of Technology, 1997.
- [16]. GOEL, A. K.; MCADAMS, D. A.; STONE, R. B. **Biologically Inspired Design: Computational Methods and Tools**. New York: Springer, 2014.
- [17]. HIRTZ, J.; STONE, R. B.; MCADAMS, D. A. A Functional Basis for Engineering Design : Reconciling and Evolving Previous Efforts. **Research in Engineering Design**, v. 13, p. 65–82, 2002.
- [18]. HUNDAL, M. A Systematic Method for Developing Function Structures, Solutions and Concept Variants. **Mechanism and Machine Theory**, p. 25(3):243-256, 1990.
- [19]. JANSSON, D. G.; SMITH, S. M. Design fixation. **Design Studies**, v. 12, 1991.
- [20]. LINSEY, J. S. **Design-by-analogy and representation in innovative engineering concept generation**. Texas: University of Texas at Austin, 2007.
- [21]. LINSEY, J.; TEXAS, A.; MARKMAN, A. WordTrees : A Method for Design-by-Analogy. **American Society for Engineering Education**, 2008.
- [22]. LÖBACH, B. **Design industrial: bases para a configuração dos produtos industriais**. Rio de Janeiro, Brasil: Editora Edgard Blücher Ltda., 2001.
- [23]. LUCERO, B. M. **Design-Analogy Performance Parameter System (D-APPS)**. Doctor of Philosophy (Engineering Systems)—Golden, Colorado: Colorado School of Mines, 2014.
- [24]. MILLER, G. A. WordNet: a lexical database for English. **Communications of the ACM**, v. 38, n. 11, p. 39–41, 1995.
- [25]. MORENO, D. P. et al. Fundamental studies in Design-by-Analogy: A focus on domain-knowledge experts and applications to transactional design problems. **Design Studies**, v. 35, n. 3, p. 232–272, 2014.
- [26]. MURPHY, J. et al. Facilitating Design-By-Analogy: Development Of A Complete Functional Vocabulary And Functional Vector Approach To Analogical Search. **Journal of Mechanical Design**, v. 136, n. 10, p. 101102, 2014.

- [27]. ORIAKHI, E. V.; LINSEY, J. S.; PENG, X. Design-by-analogy using the wordtree method and an automated wordtree generating tool. **International Conference on Engineering Design, ICED'11**, n. August, 2011.
- [28]. PAHL, G.; BEITZ, W. **Engineering Design A Systematic Approach**. London: Springer, 1988.
- [29]. PREECE, J.; ROGERS, Y.; SHARP, H. **Interaction Design: Beyond Human-Computer Interaction**. United States of America: John Wiley & Sons, Inc, 2013.
- [30]. ROMEIRO FILHO, E. **Projeto do Produto**. São Paulo, Brasil: Elsevier, 2010.
- [31]. ROSENFELD, H. et al. **Gestão de Desenvolvimento de Produtos**. São Paulo: Editora Saraiva, 2006.
- [32]. STONE, R. B.; WOOD, K. L. Development of a Functional Basis for Design. **Journal of Mechanical Design**, p. 1–36, 2001.
- [33]. SZYKMAN, S. et al. Design Repositories: Engineering Design's New Knowledge Base. **IEEE Intelligent Systems**, p. 48–55, 2000.
- [34]. SZYKMAN, S.; SRIRAM, R. D. Design and implementation of the web-enabled nist design repository. **ACM Transactions on Internet Technology**, v. 6, n. 1, p. 85–116, 2006.
- [35]. THE BIOMIMICRY INSTITUTE. Ask Nature! **BioInspired!**, v. 6, n. 3, 2008.
- [36]. TOMKO, M. et al. Establishing Functional Concepts Vital for Design by Analogy. **IEEE**, 2015.
- [37]. TSENG, I. et al. The role of timing and analogical similarity in the stimulation of idea generation in design. **Design Studies**, p. 203–221, 2008.
- [38]. ULLMAN, D. **The mechanical design process**. New York, New York, USA: McGraw-Hill, 2010.
- [39]. ULRICH, K. T.; EPPINGER, S. D. **Product Design and Development**. 6. ed. New York, New York, USA: McGraw-Hill, 2015.
- [40]. VERHAEGEN, P. A. et al. Identifying candidates for design-by-analogy. **Computers in Industry**, v. 62, n. 4, p. 446–459, 2011.
- [41]. WANG, L. et al. Collaborative conceptual design - state of the art and future trends. **Computer-Aided Design**, v. 34, n. 13, p. 981–996, 2002.

*O presente trabalho foi realizado com apoio da CAPES, Coordenação de Aperfeiçoamento de Pessoal de Nível Superior – Brasil.*