

UNIVERSIDADE TECNOLÓGICA FEDERAL DO PARANÁ
DEPARTAMENTO ACADÊMICO DE COMPUTAÇÃO
CURSO DE BACHARELADO EM CIÊNCIA DA COMPUTAÇÃO

NOEMI PEREIRA SCHERER

**AVALIAÇÃO HEURÍSTICA E TESTE DE
USABILIDADE PARA SOFTWARES DE DESIGN DE
INTERIORES**

MONOGRAFIA

CAMPO MOURÃO

2018

NOEMI PEREIRA SCHERER

**AVALIAÇÃO HEURÍSTICA E TESTE DE
USABILIDADE PARA SOFTWARES DE DESIGN DE
INTERIORES**

Trabalho de Conclusão de Curso de graduação apresentada à disciplina de Trabalho de Conclusão de Curso 2, do Curso de Bacharelado em Ciência da Computação do Departamento Acadêmico de Computação da Universidade Tecnológica Federal do Paraná, como requisito parcial para obtenção do título de Bacharel em Ciência da Computação.

Orientador: Prof. Dr. André Luiz Satoshi Kawamoto

**CAMPO MOURÃO
2018**



ATA DE DEFESA DO TRABALHO DE CONCLUSÃO DE CURSO

Às **10:00** do dia **15 de junho de 2018** foi realizada na sala **E104** da UTFPR-CM a sessão pública da defesa do Trabalho de Conclusão do Curso de Bacharelado em Ciência da Computação do(a) acadêmico(a) **Noemi Pereira Scherer** com o título **Avaliação heurística e teste de usabilidade para softwares de design de interiores**. Estavam presentes, além do(a) acadêmico(a), os membros da banca examinadora composta por: **Prof. Dr. Andre Luiz Satoshi Kawamoto** (orientador), **Prof. Dr. Ivanilton Polato** e **Prof. Dr. Marco Aurélio Graciotto Silva**. Inicialmente, o(a) acadêmico(a) fez a apresentação do seu trabalho, sendo, em seguida, arguido(a) pela banca examinadora. Após as arguições, sem a presença do(a) acadêmico(a), a banca examinadora o(a) considerou _____ na disciplina de Trabalho de Conclusão de Curso **2** e atribuiu, em consenso, a nota ____ (_____). Este resultado foi comunicado ao(à) acadêmico(a) e aos presentes na sessão pública. A banca examinadora também comunicou ao acadêmico(a) que este resultado fica condicionado à entrega da versão final dentro dos padrões e da documentação exigida pela UTFPR ao professor Responsável do TCC no prazo de **onze dias**. Em seguida foi encerrada a sessão e, para constar, foi lavrada a presente Ata que segue assinada pelos membros da banca examinadora, após lida e considerada conforme.

Observações: _____

Campo Mourão, **15 de junho de 2018**

Prof. Dr. Ivanilton Polato

Membro 1

**Prof. Dr. Marco Aurélio Graciotto
Silva**

Membro 2

**Prof. Dr. Andre Luiz Satoshi
Kawamoto**
Orientador

A ata de defesa assinada encontra-se na coordenação do curso.

Resumo

Scherer, Noemi. Avaliação heurística e teste de usabilidade para softwares de design de interiores. 2018. 53. f. Monografia (Curso de Bacharelado em Ciência da Computação), Universidade Tecnológica Federal do Paraná. Campo Mourão, 2018.

Um *designer de interiores* é o profissional responsável por criar e modificar qualquer espaço interno por meio do planejamento e da execução de um projeto. Como auxílio para seu trabalho, esse profissional utiliza softwares conhecidos como *Computer-Aided Design-CAD*, os quais possuem capacidade de criar e mostrar um ambiente por meio de projeções bidimensionais ou tridimensionais.

Muitos desses programas podem apresentar problemas em suas interfaces, dificultando ou impossibilitando o seu uso pelos usuários. Dessa forma, surge a necessidade de realizar uma avaliação capaz de identificar e classificar esses problemas.

O presente trabalho visa aplicar duas avaliações de usabilidade, uma heurística e um teste de usabilidade. A primeira baseia-se na utilização das heurísticas propostas por Nielsen, originalmente específicas para aplicações *Web*, que foram adaptadas nesse trabalho para o contexto de *design de interiores*. Já a segunda utiliza o modelo *System Usability Scale - SUS*. Para comprovar a importância dessas avaliações, elas foram aplicadas a um software real.

Os resultados obtidos permitiram classificar a usabilidade da interface do programa avaliado. Com a avaliação heurística foi possível determinar erros específicos. Já com o teste de usabilidade foi determinado e classificado todos os problemas encontrados, categorizando o sistema em geral.

Palavras-chaves: *design de interiores*, usabilidade, avaliação heurística, teste de usabilidade, *system usability scale*

Abstract

Scherer, Noemi. Heuristic evaluation and usability test for interior design software. 2018. 53. f. Monograph (Undergraduate Program in Computer Science), Federal University of Technology – Paraná. Campo Mourão, PR, Brazil, 2018.

An interior designer is the professional responsible for creating and modifying any internal space by planning and executing a project. To assist his work, this professional uses software known as Computer-Aided Design (CAD), which have the ability to create and display an environment through two-dimensional or three-dimensional projections.

Many of these programs may have problems in their interfaces, turning it difficult or impossible for users to use them. Thus, it is necessary arises to carry out an evaluation capable of identifying and classifying these problems.

The present work apply two usability evaluations, a heuristic and a usability test. The first is based on the heuristics proposed by Nielsen, originally specific to Web applications, that were adapted in this work to the context of interior design. The second uses the System Usability Scale - SUS model. To prove the importance of these evaluations, they were applied to real software.

The results allowed to classify the interface software of the evaluated program. With a heuristic evaluation it was possible to determinated specifics errors. With the usability test was determined and classified all the problems found, categorizing the system in general.

Keywords: interior design, usability, heuristic evaluation, usability test, system usability scale

Lista de figuras

2.1	Projeto de ambiente criado no (a) MinCAD e (b) Sweet Home 3D.	12
2.2	Exemplo de questão do SUS.	19
3.1	Status do sistema: Progresso de instalação de um objeto no Sketchup.	22
3.2	Compatibilidade do sistema com o mundo real: A ferramenta de criação de paredes do Sweet Home 3D.	22
3.3	Controle do usuário e liberdade: Botão de interromper uma ação do Sktechup.	23
3.5	Prevenção de erros: Opções de filtro do Sweet Home 3D.	24
3.6	Reconhecimento em vez de memorização: Ícones de ferramentas do Sketchup.	24
3.7	Flexibilidade e eficiência de uso: Alguns atalhos permitidos no Sketchup. . .	25
3.8	Estética e design minimalista: Interface inicial do Sweet Home 3D.	26
3.9	Ajuda aos usuários no reconhecimento, diagnóstico e correção de erros: Janela de erro do Sweet Home 3D.	26
3.10	Ajuda e documentação: Menu de ajuda.	27
4.1	Interface do software Sweet Home 3D.	29
4.2	Planta baixa para o minicurso.	30
4.3	Planta baixa importada no software.	31
5.1	Problema de estética e <i>design</i> minimalista.	39
5.2	Resultados dos SUS em porcentagem.	40
5.3	Aplicação do SUS para o programa Sweet Home 3D: Resultados por participante.	41
5.4	Resultados das questões do SUS.	42

Lista de tabelas

2.1	As questões do SUS	20
5.1	Problemas classificados na heurística 2 - Compatibilidade do sistema com o mundo real.	33
5.2	Problema classificado na heurística 3 - Controle do usuário e liberdade.	34
5.3	Problema classificado na heurística 4 - Consistência e padrões.	35
5.4	Problemas classificados na heurística 5 - Reconhecimento em vez de memorização.	35
5.5	Problema classificado na heurística 7 - Flexibilidade e eficiência de uso.	37
5.6	Problema classificado na heurística 8 - Estética e <i>design</i> minimalista.	37
5.7	Problema classificado na heurística 10 - Ajuda e documentação.	38
5.8	Resultado do SUS.	40
5.9	Respostas de algumas questões do usuário 23.	42
5.10	Problemas encontrados pela avaliação heurística.	44
5.11	Classificação do Sweet Home 3D de acordo com os resultados do teste de usabilidade.	44

Sumário

1	Introdução	8
1.1	Contextualização	8
1.2	Motivação	9
1.3	Objetivo	9
1.4	Justificativa	10
1.5	Estrutura do trabalho	10
2	Referencial Teórico	11
2.1	Ferramentas de <i>design de interiores</i>	11
2.2	Interação Humano-Computador	12
2.2.1	Qualidade de software	13
2.2.2	A norma ISO/IEC 14598	13
2.2.3	Usabilidade: ISO 9241-11	14
2.2.4	Métodos indiretos	16
2.2.4.1	Avaliação heurística	16
2.2.5	Métodos diretos	18
2.2.5.1	Teste de usabilidade	18
2.2.5.2	System Usability Scale - SUS	19
3	Avaliações de Usabilidade Propostas	21
3.1	Adaptação do conjunto de heurísticas de Nielsen para software de <i>design de interiores</i>	21
3.1.1	Status do sistema	21
3.1.2	Compatibilidade do sistema com o mundo real	22
3.1.3	Controle do usuário e liberdade	22
3.1.4	Consistência e padrões	23
3.1.5	Prevenção de erros	23
3.1.6	Reconhecimento em vez de memorização	24
3.1.7	Flexibilidade e eficiência de uso	24
3.1.8	Estética e design minimalista	25
3.1.9	Ajuda aos usuários no reconhecimento, diagnóstico e correção de erros	26

3.1.10	Ajuda e documentação	26
3.2	Teste de usabilidade aplicado a software de <i>design de interiores</i>	27
4	Aplicação das Avaliações	28
4.1	Aplicação da avaliação heurística	28
4.2	Aplicação do teste de usabilidade	29
4.2.1	Instrumento de avaliação	31
5	Resultados e Discussão	32
5.1	Resultados da avaliação heurística	32
5.2	Resultados do teste de usabilidade	39
5.3	Análise dos resultados por aluno	39
5.4	Análise dos resultados de cada questão	42
5.5	Considerações finais	43
6	Conclusão	45
	Referências	46
	Apêndices	49
A	Formulário do Teste de Usabilidade	50
B	Instruções para realizar a avaliação heurística	53

Introdução

1.1. Contextualização

A partir do momento em que pessoas começaram a compartilhar ambientes fechados, surgiu a ideia de melhorar esteticamente os espaços em que habitam e, como consequência, a necessidade de otimizar esse espaço. Com isso, a profissão de decorador, que anteriormente era associada a arquitetura e *design de interiores*, ganhou maior importância (PULE, 2005).

Um *designer de interiores* ou *designer de ambientes* é o profissional que atua na área de *design de interiores* e é responsável por cuidar de espaços internos, tanto residenciais como corporativos, seguindo leis ergonômicas, acústicas, térmicas e luminotécnicas. Ele deve verificar as necessidades dos seus clientes e as atribuir ao projeto, delinear completamente o ambiente, mas não alterar nenhuma estrutura física (VELA et al., 2016). O seu trabalho possui duas principais etapas: o planejamento e a execução de um projeto (VEIGA, 2015). projeta um espaço

A evolução da decoração aconteceu devido a movimentos artísticos com características únicas de ornamentação, como o Renascimento, Barroco, *Art Nouveau*, Cubismo, Futurismo, entre outros. Apesar desses movimentos revolucionarem a área, foi apenas com a Revolução Industrial (séculos XVIII e XIX) que ocorreu o auge da decoração de ambientes devido à redução da produção manual e ao impulsionamento para o uso de máquinas. Essa revolução possibilitou a produção em alta escala de produtos que antes eram criados manualmente como acessórios tecidos, papéis de parede e móveis para casas (VELA et al., 2016). Outra importante influência na área de decoração foi a Escola de Design de Bauhaus (PULE, 2005), que, mesmo fechada, propagou seus conhecimentos sobre ornamentação por meio de seus alunos egressos.

O trabalho do *designer* é baseado em um planejamento para criação de um projeto,

que pode ser representado por plantas, memoriais, referenciais e visualizações bidimensionais (2D) ou tridimensionais (3D). Mediante ele são mostradas as ideias, orçamentos e o ambiente planejado (VEIGA, 2015).

Tradicionalmente um projeto de arquitetura é realizado em nove etapas: reunião de *briefing*, orçamento de projeto, levantamento métrico e fotográfico, definição do estado atual, criação da planta baixa, elaboração da modelagem 3D, detalhamento técnico, orçamento de execução e serviços complementares (SALIS, 2016). As fases de definição do estado atual, criação da planta baixa e elaboração da modelagem 3D implicam em representar graficamente o desenho do ambiente em uma ferramenta computacional do tipo Desenho Assistido por Computador (*Computer-Aided Design-CAD*), criado inicialmente para projetar desenhos mecânicos (HUA; KUI, 2015).

A tecnologia possibilitou criação de diversos softwares de projeção 2D e 3D que podem ser utilizados por *designer de interiores*. O que muda de uma ferramenta para outra são as funcionalidades oferecidas, responsáveis por agradar ou não ao usuário (MONTENEGRO, 1987). O seu uso é essencial para o trabalho de um *designer*, pois o torna mais fácil e prático, reduz o tempo de criação, revisão e alteração do projeto, garante um trabalho mais econômico, uma boa precisão e melhora do fluxo das informações.

Muitos programas do tipo CAD apresentam problemas em suas interfaces, dificultando o seu uso. Dessa forma, surge a necessidade de realizar uma avaliação capaz de identificar problemas de interface nesses softwares.

1.2. Motivação

Os programas oferecidos pelo computador estão cada vez mais presentes no cotidiano das pessoas, além de agilizar e facilitar a execução de tarefas originalmente manuais. Isso também é aplicado para softwares de *design de interiores*. Entretanto, de acordo com estudos, as interfaces desses programas são projetadas de maneira incorreta, tornando a execução de uma tarefa mais difícil e de maior tempo (MEDEIROS; CYBIS, 2000).

A usabilidade permite avaliar a funcionalidade de um sistema. Ela pode ser aplicada por meio de uma avaliação heurística e por um teste de usabilidade, que determinam quais recursos do programa não foram projetados de maneira efetiva aos usuários. A primeira é realizada por especialistas da área de Interação Humano-Computador(IHC) e a segunda por um grupo de usuários que utilizam o sistema.

1.3. Objetivo

O objetivo desse trabalho é aplicar uma avaliação heurística e um teste de usabilidade. Para a primeira, é adaptado o conjunto de heurísticas originalmente proposto por Nielsen

(NIELSEN; MACK, 1994), para o contexto de programas de *design de interiores*. As duas avaliações tem como finalidade determinar problemas na interface de um software, e por meio de seus resultados será reforçado a importância de avaliar a usabilidade de um sistema.

Os objetivos específicos são:

1. Adaptar as heurísticas propostas por Nielsen para a área de *design de interiores*;
2. Aplicar as heurísticas estabelecidas a um software da área por meio de uma avaliação heurística;
3. Aplicar um teste de usabilidade em um programa de *design de interiores*;
4. Analisar os problemas de usabilidade encontrados por meio da avaliação heurística e do teste de usabilidade;
5. Comparar os resultados das duas avaliações e definir a importância de realizar uma avaliação de usabilidade em programas da área, bem como os problemas que devem ser melhorados no sistema avaliado.

1.4. Justificativa

A usabilidade de um sistema influencia na qualidade percebida pelos seus usuários durante sua interação com esse sistema. A avaliação da usabilidade é uma ferramenta que permite determinar erros na interface. Os dois instrumentos de avaliação propostos são usados por diferentes tipos de avaliadores: a avaliação heurística deve ser feita por especialistas, e o teste de usabilidade por usuários finais. A aplicação desses dois tipos de avaliação deve ser capaz de indicar diferentes tipos de erros e, ao mesmo tempo, indiretamente aferir a qualidade de um sistema.

1.5. Estrutura do trabalho

Este trabalho está organizado da seguinte forma: é apresentado no Capítulo 2 um breve histórico das ferramentas de *design*, conceito de qualidade de software e dois métodos de avaliação de usabilidade; já no Capítulo 3 são definidas as heurísticas adaptadas para o contexto de *design de interiores* e a aplicação do teste de usabilidade; no Capítulo 4 são especificadas as etapas para a aplicação da avaliação heurística e do teste; o Capítulo 5 apresenta os resultados obtidos dessas duas avaliações; por fim, o Capítulo 6 define as conclusões obtidas por meio dos resultados.

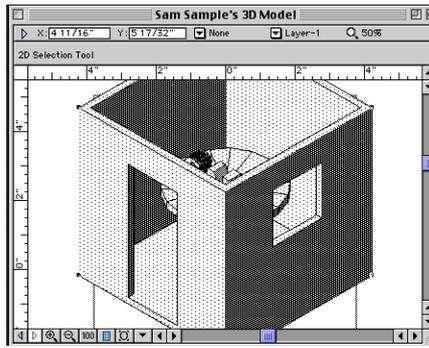
Referencial Teórico

2.1. Ferramentas de *design de interiores*

Na metade dos anos 1980, os computadores de alto custo começaram a ser substituídos por computadores pessoais (PCs - *Personal Computers*). Entretanto, as ferramentas de modelagens 2D e 3D para *design* ainda possuíam recursos limitados e eram uma alternativa de lazer e não comercial, pois a profissão de *design de ambientes* ainda ganhava reconhecimento e, para profissionais executarem desenhos técnicos, deveriam utilizar plataformas mais caras, as quais ofereciam recursos acessíveis apenas para empresas de médio ou grande porte.

No final dos anos 1980 e começo dos 1990, o uso de computação gráfica começou a expandir nas áreas de arquitetura, tendo como consequência a evolução dos programas com projeção de objetos tridimensionais. A empresa Autodesk Inc. é responsável por diversos programas da área de arquitetura e *design de interiores*. No final de 1982, ela lançou o AutoCAD, considerado um dos primeiros softwares do tipo CAD da época, o qual funcionava apenas em computadores de elevado custo, conhecidos como *mainframes* (DUARTE, 2014). O AutoCAD tornou-se um dos softwares mais utilizados, por não exigir muitos recursos computacionais para sua execução.

A modelagem tridimensional dos objetos em um programa começou a melhorar após os anos 2000, devido ao surgimento e aprimoramento de ferramentas complexas, diferentes das versões anteriores, com imagens mais realistas e novos recursos para criação dos projetos. Essa evolução pode ser percebida por meio das interfaces de softwares utilizados ao longo da época (Figura 2.1(a) e Figura 2.1(b)). Na figura à esquerda, temos a interface da primeira versão do MinCAD, de 1984, que possuía recursos escassos em comparação com a figura da direita, do software Sweet Home 3D, de 2017, que apresenta maiores funcionalidades, detalhes e formas de criar e visualizar o projeto.

(a) Fonte: Sandro Corsi¹(b) Fonte: Sweet Home 3D²**Figura 2.1.** Projeto de ambiente criado no (a) MinCAD e (b) Sweet Home 3D.

Existem diversos programas utilizados por *designers*, como o AutoCAD, Revit, SketchUp, Sweet Home 3D, entre outros (AUTODESK, 2018a), (AUTODESK, 2018b), (SKETCHUP, 2018), (Sweet Home 3D, 2018a). A interface desses sistemas devem atender a diferentes objetivos dos usuários.

2.2. Interação Humano-Computador

A Interação Homem-Computador ou Interface Humano-Computador- IHC é uma área da computação que investiga o *design*, avalia e desenvolve interfaces para que usuários finais possam interagir com sistemas computacionais de uma forma eficiente e intuitiva (OLIVEIRA, 2011).

A IHC começou a consolidar-se no início dos anos 80 como uma área da computação científica, na qual considerava fatores humanos na criação de programas. A partir de então, percebeu-se sua importância e o seu estudo foi crescendo cada vez mais (GLORIA, 2015).

Uma interação é o processo de comunicação entre o usuário e o sistema e é realizada mediante uma interface, que é o meio pelo qual os usuários irão se comunicar com o sistema, permitindo a entrada e saída de dados (OLIVEIRA, 2011). Dessa forma, um projeto interativo possui a preocupação da qualidade de uso associado à interação do usuário com a interface do sistema

Um dos maiores desafios da IHC é acompanhar a evolução tecnológica e atender, ao mesmo tempo, a todos os grupos de usuários. Nem todos os usuários conseguem dominar as funcionalidades de um sistema, dessa forma os desenvolvedores dos softwares possuem o desafio de criar novas interfaces, mas que não deixem de lado a objetividade e a clareza (OLIVEIRA, 2011).

¹ http://www.sweethome3d.com/blog/2017/05/12/how_to_show_interior_parts_hidden_by_walls.html

² http://www.sanedraw.com/LEARN/OVRV00_1/OV150/MAC0EN_P.HTM

2.2.1. Qualidade de software

A qualidade de um produto é subjetiva e varia de acordo com quem a avalia, com o local e com a época. Por esse motivo, a qualidade é medida por meio de atributos do produto de forma que seja possível verificar se diferentes versões desse produto apresentam a mesma qualidade (CÔRTEZ; CHIOSSI, 2001).

Para que um produto de software possua qualidade, a administração e outros setores internos da empresa devem possuir esforço e comprometimento, tornando necessária uma mudança cultural, como o cumprimento de normas técnicas. Nesse sentido, duas importantes organizações internacionais são responsáveis pela criação de normas, a Comissão Eletrotécnica Internacional (*International Electrotechnical Commission - IEC*) que cuida dos regulamentos na área de eletricidade e eletrônica, e a Organização Internacional de Normalização (*International Organization Standardization - ISO*) responsável por outras áreas (CÔRTEZ; CHIOSSI, 2001).

A associação ISO/IEC cria normas gerais para todas as áreas (CÔRTEZ; CHIOSSI, 2001). No Brasil, essas normas são representadas pela Associação Brasileira de Normas Técnicas (ABNT), que é um fórum nacional de criação de regulamentos brasileiros, com envolvimento de produtores, consumidores, universidades e laboratórios (ABNT, 2003).

2.2.2. A norma ISO/IEC 14598

A norma ISO/IEC 14598:2001 define o processo de avaliação e é dividida em seis partes (PARREIRA JÚNIOR et al., 2009). O procedimento de avaliação da ISO 14598 é baseado na ISO/IEC 9126, que pode ser utilizada para analisar produtos prontos ou em desenvolvimento. De acordo com a norma da ISO 14598-1, a avaliação é dividida em quatro etapas:

1. estabelecimento de requisitos;
2. especificação da avaliação;
3. definição do projeto da avaliação;
4. execução da avaliação.

O **estabelecimento de requisitos** é feito em três passos: inicialmente é necessário estabelecer o propósito da avaliação, identificar os tipos de produto(s) a serem avaliados e especificar o modelo de qualidade. Em seguida, define-se o objetivo de realizar a avaliação. Já no segundo passo, estabelece-se o produto/software que será avaliado. Finalmente, o modelo de qualidade utilizado na avaliação é definido (PARREIRA JÚNIOR et al., 2009).

A **especificação da avaliação** também é dividida em três passos: selecionar métricas, estabelecer níveis de pontuação para as métricas e estabelecer critérios para julgamento. O primeiro especifica que as características e sub-características da qualidade não podem ser medidas diretamente. O segundo define o tipo de classificação utilizada, pois

toda métrica envolve um nível de classificação. O último define se os requisitos do produto são positivos ou negativos (PARREIRA JÚNIOR et al., 2009).

A **definição do projeto da avaliação** define um passo nomeado "plano de avaliação", que descreve os métodos de avaliação e especifica a criação de um cronograma contendo as etapas realizadas pelo avaliador (PARREIRA JÚNIOR et al., 2009).

A **execução da avaliação** possui três partes: obter medidas, comparar critérios e julgar os resultados. Na primeira verifica-se se a pontuação obtida corresponde a escala da métrica utilizada. Já na segunda, compara-se o requisitos predeterminados com os valores medidos. Por fim, na última julga-se os resultados obtidos (PARREIRA JÚNIOR et al., 2009).

Como o objetivo desse trabalho é realizar duas avaliações de usabilidade, essa norma torna-se uma guia, possibilitando determinar e adaptar todos os passos necessários para as avaliações.

2.2.3. Usabilidade: ISO 9241-11

A usabilidade é uma medida que permite o uso de determinados produtos por usuários para eles alcançarem objetivos específicos com eficácia, eficiência e satisfação (ALVES et al., 2015). A **eficácia** refere-se a quão bom é o sistema em fazer algo esperado. A **eficiência** está vinculada em como o sistema auxilia seus usuários a realizarem determinadas tarefas (PREECE et al., 2005). Por fim, a **satisfação** é o conforto dos usuários ao utilizar determinado sistema (GLORIA, 2015).

A norma ISO 9241-10 esclarece os benefícios de medir usabilidade em termos de satisfação e desempenho dos usuários durante a realização de uma tarefa.

A interface de um sistema apresenta problemas de usabilidade quando um ou mais usuários encontram dificuldades em realizar qualquer tarefa com o mesmo. Diversos fatores podem contribuir para essas coisas, causando perda de dados, diminuição da produtividade e até a desistência do uso do software (WINCKLER; PIMENTA, 2002).

Os problemas de interface podem ser gerados por uso de recursos multimídia de maneira inadequada; por funções incorretas; ou por fatores culturais. De forma geral, é difícil determinar todos os possíveis problemas de usabilidade, porém, podem ser definidas quatro métricas ou fatores a serem observados para determina-los, desempenho do usuário durante a realização de tarefa, satisfação subjetiva do usuário, correspondência com os objetivos do usuário e adequação a padrões (WINCKLER; PIMENTA, 2002):

O **desempenho do usuário durante a realização de tarefa** é a observação das tarefas realizadas pelo usuário na qual possibilita a verificação de algumas métricas, sendo elas: a conclusão de tarefas, nas quais as atividades que são parcialmente ou não concluídas podem indicar a existência de algum problema de usabilidade; o tempo de realização da tarefa, no qual uma tarefa concluída em um tempo excessivamente longo pode indicar esforço

desnecessário exigido pelo usuário; e ocorrência de erros onde muitos tipos de erros podem acontecer durante a realização de uma determinada tarefa. Se o usuário é quem comete o erro, verifica-se se a interface não induz ao mesmo por meio de ações complexas ou falta de alertas ou informações adequadas. Se o erro acontece devido a alguma atividade do sistema, verifica-se como o usuário o recebe e quais suportes são oferecidos pela interface para solucionar o erro.

A **satisfação subjetiva do usuário** acontece quando os usuários estão completamente satisfeitos com a interface do software e o efeito de eventuais problemas são reduzidos.

Na **correspondência com os objetivos do usuário** deve-se verificar se os objetivos dos usuários foram alcançados, independente dos recursos suportados pelo sistema;

Na **adequação a padrões** (norma, recomendações ergonômicas, etc) uma grande parte da usabilidade é organizada de acordo com normas, tais como a ISO 9241. Elas são responsáveis por determinar padrões e problemas, e muitas vezes, propõem soluções ou alternativas para evitá-los.

Ainda, de acordo com a ISO 9241, para avaliar a usabilidade de um sistema não basta apenas medir questões relacionadas a funcionalidade do mesmo, mas também à facilidade do seu uso, tendo como um dos principais desafios a redução do tempo para aprender a utilizar o programa (MACIEL et al., 2004).

A compreensão das necessidades é muito importante para a produção ou alteração de um sistema, pois com elas é possível atender todos os objetivos e expectativas do usuário, tornando o produto eficiente. Essa compreensão é denominada metas de usabilidade (PREECE et al., 2005), sendo elas: eficácia, eficiência, segurança, utilidade, facilidade de aprender (*learnability*) e facilidade de lembrar como se usa (*memorability*).

A **eficácia e eficiência** já foram definidas anteriormente. A **segurança** refere à proteção dos usuários de situações perigosas ou inesperadas. A **utilidade** define se as funções do sistema estão proporcionadas corretamente para o usuário realizar as tarefas desejadas. A **facilidade de aprender** define se é fácil ou não aprender utilizar o programa. Por fim, a **facilidade de lembrar** caracteriza se é fácil de lembrar a usar o sistema após ter aprendido a utilizá-lo (PREECE et al., 2005).

Para solucionar problemas de usabilidade deve-se sugerir correções dos projetos quando em desenvolvimento; realizar revisões, ajustes ou customização em produtos finalizados; aceitar ou não projetos encomendados; e comparar o desempenho de softwares iterativos (MACIEL et al., 2004).

Existem cinco especificações as quais permitem uma descrição melhor de um problema de usabilidade e facilitam a sua correção:

- contexto no qual um problema pode ser verificado ou resolvido;
- causa que é o motivo que propicia o problema;

- efeito sobre o usuário que é a consequência da interação sobre o mesmo, podendo existir sobrecarga, desorientação ou hesitação;
- efeito sobre a tarefa que são consequências causadas de uma determinada ação, podendo haver trabalho adicional, perda de dados ou de tempo;
- correção possível, que indica ao desenvolvedor do sistema possíveis alterações.

Além das especificações, um problema de usabilidade pode ser classificado em duas categorias, como **principal** ou **secundário**. O primeiro refere-se à problemas frequentes ou importantes, já o segundo à pouco frequentes ou importantes (MACIEL et al., 2004).

Para garantir que a usabilidade esteja presente em um sistema são realizadas avaliações de usabilidade. Essa técnicas podem ser classificados como métodos indiretos e métodos diretos.

2.2.4. Métodos indiretos

Os profissionais de IHC preocupam-se com métodos rápidos e baratos de avaliação de interfaces para substituírem testes laboratoriais, que apresentam alto custo e necessitam de uma infraestrutura adequada. Dessa forma, são indicadas técnicas de avaliação denominadas métodos de inspeção de usabilidade ou métodos indiretos (MACIEL et al., 2004).

Os métodos indiretos são aqueles nos quais os avaliadores analisam a usabilidade do sistema aplicando técnicas de coleta de dados sem a necessidade de envolver usuários finais. Seu principal objetivo é encontrar problemas de usabilidade (GLORIA, 2015).

A avaliação heurística é um exemplo de método indireto. Ela é realizada por meio de um conjunto de heurísticas, ou seja, princípios ou regras, nas quais têm como objetivo encontrar os problemas da interface (GLORIA, 2015).

Esses métodos são realizados por um grupo de três a cinco especialistas da área da computação, os quais serão os avaliadores. Recomenda-se que esses utilizem o sistema pelo menos duas vezes antes de iniciar a avaliação, com o objetivo de familiarizar-se com as características do software. A avaliação é realizada individualmente por cada avaliador, a fim de evitar a influência de comentários dos outros especialistas. Durante o percurso o avaliador deve identificar problemas de usabilidade descrito em cada heurística.

2.2.4.1. Avaliação heurística

A avaliação heurística é uma técnica de avaliação a qual permite encontrar diferentes problemas em interfaces de sistemas. Ela consiste em um conjunto de regras gerais (heurísticas) que descrevem propriedades comuns em interfaces derivadas do conhecimento de aspectos psicológicos, computacionais e sociológicos (GLORIA, 2015).

Essa avaliação pode ser utilizada em qualquer momento do desenvolvimento do software, porém é aconselhável nas fases iniciais, pois a interface pode se restringir ao modelo descrito em papel.

Como a avaliação heurística é um método indireto, recomenda-se pelo menos três avaliadores com conhecimento em usabilidade, devido à subjetividade da avaliação e à possível dificuldade de aceitação dos desenvolvedores do software. Quando surgiu esse método, não era necessário que os avaliadores fossem especialistas em usabilidade (MACIEL et al., 2004).

A aplicação desse método é capaz de identificar 60%, 70% e 75% de problemas na interface, quando utilizados, respectivamente, três, quatro e cinco avaliadores. Dessa forma, de acordo com Nielsen, a aplicação de três avaliações com cinco avaliadores é melhor do que com quinze, pois existe a possibilidade, após cada avaliação, de verificar os problemas de usabilidade, redesenhar a interface e testá-la novamente, diminuindo os problemas que cada alteração realizada possa causar (NIELSEN; MACK, 1994).

A aplicação dessa avaliação em software de conhecimento geral possibilita ao avaliador conduzir seu teste sem ajuda adicional. Entretanto, se o software pertence a um domínio de aplicação específico, o avaliador necessita de apoio de forma que ele apresente condições de utilizar o sistema de forma correta, como por exemplo cenários com etapas reais que devem ser seguidas (MACIEL et al., 2004).

A avaliação heurística consiste em quatro fases que devem ser seguidas pelos especialistas. A primeira é a preparação das heurísticas; a segunda é o acompanhamento e execução da avaliação; a terceira é consolidação das heurísticas; e a quarta é a análise dos problemas encontrados de acordo com o grau de severidade, documentação e apresentação dos resultados. Porém, nem todas as fases precisam ser seguidas, podendo descartar algumas delas.

Os princípios de usabilidade fornecem uma fundamentação para realizar uma avaliação de produtos de softwares. Essa avaliação é baseada em uso de heurísticas (PREECE et al., 2005). Existem dez princípios fundamentais desenvolvido por Nielsen e seus colegas, conhecidos como heurísticas, que são específicas para aplicações *Web*, entretanto elas podem ser utilizadas em outros tipos de aplicações e combinadas com outras análises, desde que essas realizem avaliações em produtos com o objetivo de determinar a usabilidade da interface do sistema. Essas heurísticas são (PREECE et al., 2005):

1. visibilidade do *status* do sistema, o sistema fornece *feedback* do que está acontecendo para o usuário;
2. compatibilidade do sistema com o mundo real, a linguagem do sistema (palavras, frases e conceitos) deve ser familiar ao usuário;
3. controle do usuário e liberdade, o usuário consegue sair facilmente do lugar em que se encontra, pois cometeu algum engano ao escolher opções do sistema;

4. consistências e padrões: usuários não precisam pensar se palavras, situações ou ações diferentes possuem os mesmos significados;
5. ajuda os usuários a reconhecer, diagnosticar e recuperar de erros: forma simples de apresentar o problema e uma solução para ele;
6. prevenção de erros: evita acontecimento de erros;
7. reconhecimento em vez de memorização: objetos, ações e opções são visíveis;
8. flexibilidade e eficiência de uso: usuários mais experientes podem realizar tarefas com mais rapidez;
9. estética e design minimalista: informações irrelevantes não são utilizadas;
10. ajuda e documentação: define passos e informações fáceis de encontrar e seguir.

Com o objetivo de realizar uma avaliação heurística, serão adaptados os dez princípios de Nielsen, originalmente voltados para interfaces *Web* (NIELSEN; MACK, 1994), para o contexto de software *design de interiores*.

2.2.5. Métodos diretos

Os métodos diretos utilizam ferramentas e instruções em simulações e as aplicam ao próprio usuário do sistema, com o objetivo de observar seu comportamento e ouvir sua opinião. As interações com usuários acontecem por meio de questionários ou sobre a experiência de uso com o sistema (MACHADO et al., 2014).

2.2.5.1. Teste de usabilidade

O principal objetivo do teste de usabilidade é avaliar a qualidade de interação entre o usuário e o sistema, determinando e medindo impactos da interação e identificar problemas na interface responsáveis por gerar desconfortos aos usuários (MACHADO et al., 2014).

O ambiente de realização desse teste deve ser real ou próximo do real (simulação do cenário). Os avaliadores são responsáveis por criarem um roteiro (tarefa) nas quais os participantes da pesquisa devem seguir. Durante o desenvolvimento da tarefa, os avaliadores deverão acompanhar e monitorar todas as decisões realizadas pelos participantes.

Algumas etapas devem ser seguidas para o planejamento do teste de usabilidade (MACHADO et al., 2014). São elas:

- **planejamento do teste**, definir qual tarefa deve ser realizada pelo usuário;
- **organização dos materiais**, necessita-se de protótipos para a realização dos testes;
- **local de observação**, local em que o teste será realizado, ou onde simule o ambiente real de utilização do sistema avaliado;
- **seleção dos usuários**, deverá conhecer primeiro os objetivos do sistema para garantir que os participantes possuem conhecimentos e dificuldades parecidas aos reais utilizadores, com o objetivo de garantir melhor qualidade dos resultados.

- **análise dos resultados**, compreender os resultados obtidos;
- **correção da interface**, repassar as informações e resultados obtidos pelo teste para os responsáveis pelo desenvolvimento do software, garantindo assim correções em novas versões da interface.

Algumas dessas etapas serão seguidas para realizar o teste de usabilidade. O questionário e a análise dos resultados seguirão o formato *System Usability Scale* (SUS) que é descrito a seguir.

2.2.5.2. System Usability Scale - SUS

A escala *System Usability Scale* - SUS foi criada por John Brooke em 1986, no Reino Unido. Ela consiste em um questionário composto por dez questões, com cinco opções de respostas cada uma (BROOKE et al., 1996).

As respostas desse questionário são dadas por meio de uma escala *Likert* que varia de "Discordo totalmente" a "Concordo totalmente" (Figura 2.2).

Eu gostaria de utilizar este sistema frequentemente *

	1	2	3	4	5	
Discordo Totalmente	<input type="radio"/>	Concordo Totalmente				

Figura 2.2. Exemplo de questão do SUS.

O resultado do SUS é a soma da contribuição individual de cada questão, definida como *score*. Para as questões ímpares o *score* é dado pela subtração de um à resposta do usuário, já para as pares é dado pela subtração da resposta do usuário por cinco. Após obter o *score* de cada questão, esses são somados e multiplicados por 2,5. Assim, o resultado final será um índice de satisfação do utilizador, que varia de 0 a 100 (BROOKE et al., 1996).

As questões do SUS estão descritas a seguir:

Tabela 2.1. As questões do SUS

Número	Questão
1	Eu gostaria de utilizar este sistema frequentemente.
2	Eu achei o sistema desnecessariamente complexo.
3	Eu achei o sistema fácil de utilizar.
4	Eu acho que precisaria do apoio de um suporte técnico para ser possível utilizar este sistema.
5	Eu achei que diversas funções deste sistema foram bem integradas.
6	Eu achei que houve muita inconsistência neste sistema.
7	Eu imaginaria que a maioria das pessoas aprenderiam a utilizar esse sistema rapidamente.
8	Eu achei o sistema muito pesado para uso.
9	Eu me senti muito confiante utilizando esse sistema.
10	Eu precisei aprender uma série de coisas antes que eu pudesse continuar a utilizar esse sistema.

Por meio de cada questão do SUS é possível determinar os componentes de qualidade indicados por Nielsen (BOUCINHA; TAROUÇO, 2013):

- a facilidade de aprendizagem é determinada nas questões 3, 4, 7 e 10 do questionário;
- a eficiência nas questões 5, 6 e 8;
- a facilidade de memorização na questão 2;
- a minimização dos erros na questão 6;
- e a satisfação nas questões 1, 4 e 9.

Avaliações de Usabilidade Propostas

3.1. Adaptação do conjunto de heurísticas de Nielsen para software de *design de interiores*

As adaptações das heurísticas propostas por Nielsen para a área de *design de interiores* consistem em, com base na definição de cada uma, apresentar exemplos de funcionalidades nos quais programas dessa área devem conter, de forma que atendam a todas as heurísticas. Essas adaptações são descritas a seguir.

3.1.1. Status do sistema

De acordo com Nielsen e Mack (1994), o programa deve fornecer *feedback* ao usuário quando realizada alguma ação.

Para ações que são divididas em etapas o usuário deve ser informado constantemente em qual etapa a ação se encontra. Por exemplo, durante a importação de um objeto, o sistema deve apresentar uma barra indicando o progresso do andamento da operação (Figura 3.1).

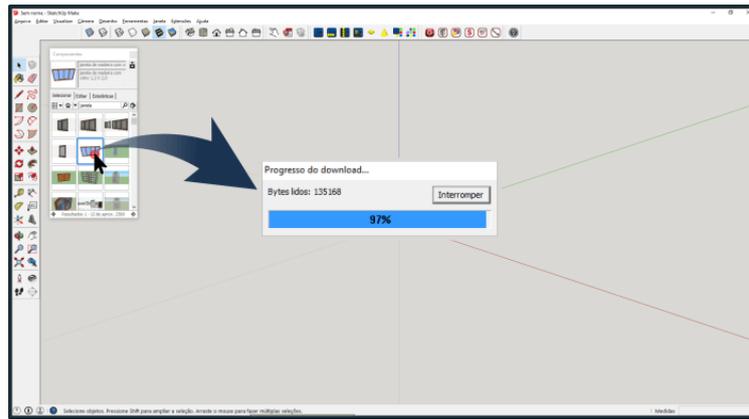


Figura 3.1. Status do sistema: Progresso de instalação de um objeto no Sketchup. Fonte: Gabster¹.

3.1.2. Compatibilidade do sistema com o mundo real

A linguagem do sistema deve ser clara, de forma que o usuário esteja familiarizado com ela. Para facilitar a compreensão, é possível utilizar imagens que correspondem à descrição.

Além disso, é essencial o uso de linguagem do público alvo e a organização das informações de forma lógica e natural para cada tipo de usuário. Deve-se evitar termos técnicos ou jargões desconhecidos (WINCKLER; PIMENTA, 2002).

Um programa de *design de interiores* deve possuir diversas ferramentas necessárias para seu uso e, para facilitar a localização dessas ferramentas, podem existir imagens metafóricas correspondentes às suas descrições, como o símbolo de parte de um cubo, como uma analogia que corresponde à criação de paredes, mostrado na Figura 3.2

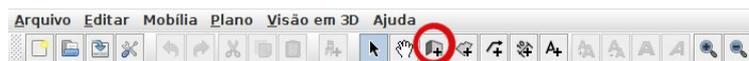


Figura 3.2. Compatibilidade do sistema com o mundo real: A ferramenta de criação de paredes do Sweet Home 3D.

3.1.3. Controle do usuário e liberdade

Os usuários precisam da sensação de controle sob todas as ações do sistema. Para garantir isso, erros no programa devem ser evitados, como o fechamento da interface gráfica durante o seu uso, ou propagandas que são abertas na execução de um recurso do sistema.

Em qualquer sistema o usuário deve conseguir cancelar a ação e a mesma ser cancelada, como por exemplo, ao importar um novo modelo para o programa, deve ser possível cancelar essa opção e retomar ao estado anterior (Figura 3.3).

¹ <https://gabster.zendesk.com/hc/pt-br/articles/210168523-Baixar-componentes-do-Armaz%C3%A9m-3D-para-SketchUp-2015>

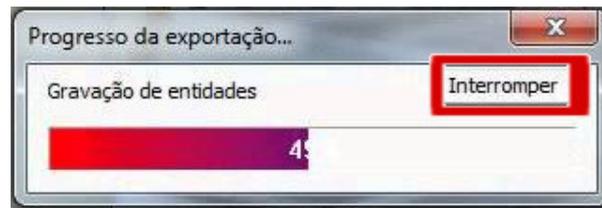


Figura 3.3. Controle do usuário e liberdade: Botão de interromper uma ação do Sktechup.

3.1.4. Consistência e padrões

O programa deve possuir a linguagem, janelas, mensagens de erros, botões, hierarquia de cores e imagens padronizadas.

Em um programa de *design de interiores* as mensagens de erros, avisos e confirmações de uma ação devem ser padronizadas, ou seja, devem apresentar os mesmos ícones e linguagem utilizada. Como demonstra a Figura 3.4, na qual as mensagens de aviso apresentam a mesma fonte, estilo linguístico, os ícones estão sempre localizadas ao lado esquerdo da mensagem e os botões apresentam o mesmo formato e cor.

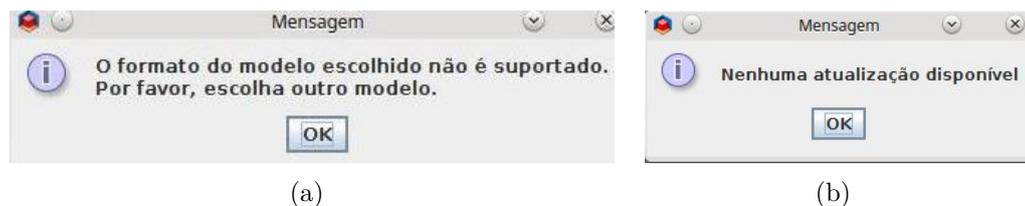


Figura 3.4. Consistência e padrões: Mensagens de aviso do Sweet Home 3D.

3.1.5. Prevenção de erros

Os erros são desagradáveis e também as principais causas de frustração com o programa, por esse motivo o software deve evitar que eles aconteçam (NIELSEN; MACK, 1994).

As informações devem ser bem organizadas e as ferramentas que o programa apresenta devem ser bem descritas para não haver erros ao selecioná-la, não obtendo ações esperadas.

Quando não é possível realizar uma ação em um programa, ela deve ser desabilitada para que o usuário não corra o risco de selecioná-la e obter nenhuma resposta. Por exemplo, filtros nas janelas para impedir a importação de arquivos em formato errado ou não reconhecido pelo sistema, como demonstra a Figura 3.5.

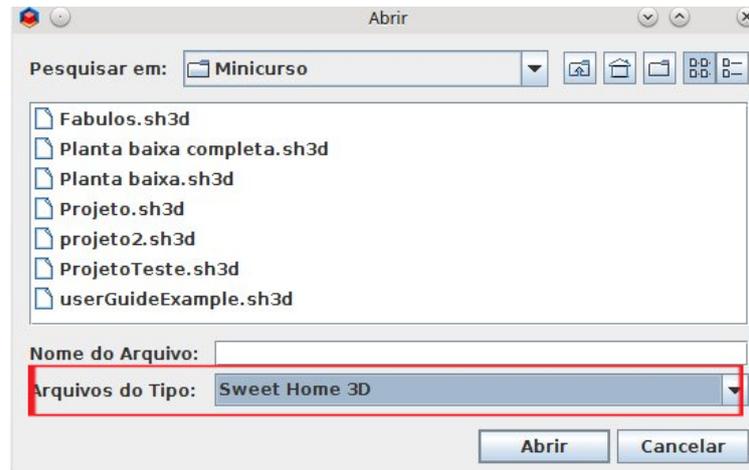


Figura 3.5. Prevenção de erros: Opções de filtro do Sweet Home 3D.

3.1.6. Reconhecimento em vez de memorização

As informações ou opções disponíveis no programa devem ser intuitivas, de forma que o usuário não necessite lembrar o seu significado. Para um bom reconhecimento, é necessário informar ao usuário em qual janela ou qual ação está sendo realizada.

Um exemplo mais comum, e que a maioria dos programas deve oferecer, é o menu. Nele deve ser possível identificar a localização e as ações possíveis, como os diversos atalhos demonstrados no menu ferramentas da Figura 3.6.



Figura 3.6. Reconhecimento em vez de memorização: Ícones de ferramentas do Sketchup. Fonte: Yting¹.

3.1.7. Flexibilidade e eficiência de uso

O programa deve ser criado de forma que atenda as necessidades de diferentes perfis de usuários, desde os iniciantes aos experientes.

O sistema deve fornecer interações que atendem a todos os tipos de usuários. Por exemplo, usuários mais experientes usam o atalho do teclado, já os usuários novatos navegam pelos menus. A Figura 3.7 apresenta alguns atalhos permitidos no programa Google Sketchup.

¹ <https://i.ytimg.com/vi/eEV0cZPJ1eo/maxresdefault.jpg>



Figura 3.7. Flexibilidade e eficiência de uso: Alguns atalhos permitidos no Sketchup. Fonte: Gabster²

3.1.8. Estética e design minimalista

As principais informações e opções do programa devem estar presentes ao iniciá-lo. Elas devem ser destacadas, ou seja, devem ser exibidas de forma que o usuário não precise procurar por elas (fácil visualização).

Por exemplo, um programa de projeção 3D deve exibir uma barra de tarefa contendo as suas principais funções (criação de paredes, pisos, seleção e ajuda), uma área que permite a visualização ou criação do projeto e dos objetos nele contido, como demonstra a Figura 3.8.

² <https://gabster.zendesk.com/hc/pt-br/articles/200595744-Cortes-com-a-ferramenta-Plano-de-sec%C3%A7%C3%A3o>

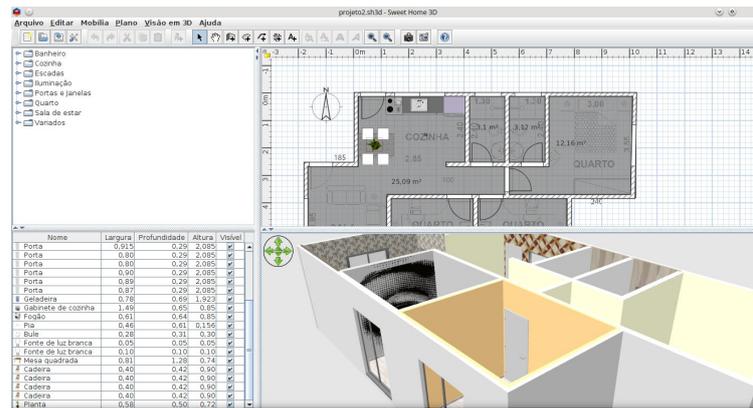


Figura 3.8. Estética e design minimalista: Interface inicial do Sweet Home 3D.

3.1.9. Ajuda aos usuários no reconhecimento, diagnóstico e correção de erros

O programa deve fornecer uma forma para que os usuários solucionem possíveis erros encontrados no mesmo, garantindo o reconhecimento da gravidade do problema. Por exemplo, ao encontrar uma mensagem informando determinado erro, a mesma deve explicar o motivo ou conter uma solução (Figura 3.9).

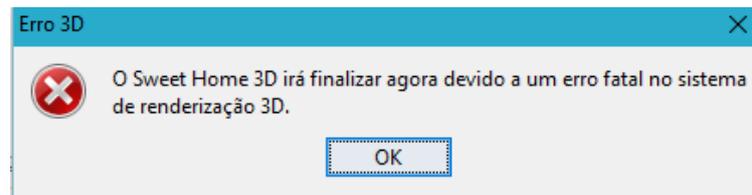


Figura 3.9. Ajuda aos usuários no reconhecimento, diagnóstico e correção de erros: Janela de erro do Sweet Home 3D.

3.1.10. Ajuda e documentação

O programa deve fornecer ajuda para tirar dúvida dos usuários.

Um exemplo específico para um programa na área de *design de interiores* é a documentação apresentada ao usuário quando se seleciona a opção 'AJUDA' no menu, que explica como o usuário deve proceder quando deseja acionar alguma funcionalidade específica (Figura 3.10).



Figura 3.10. Ajuda e documentação: Menu de ajuda.

3.2. Teste de usabilidade aplicado a software de *design de interiores*

O teste de usabilidade aplicado aos usuários finais utilizou o SUS. Não foi necessário adaptar esse modelo, pois as suas questões são abrangentes o suficiente para detectar problemas de usabilidade na interface de sistemas.

Por meio do teste aplicado é possível examinar a facilidade de aprendizagem do software; verificar a eficiência do programa; avaliar a facilidade de memorização; identificar inconsistências do sistema; verificar a satisfação dos usuários e identificar oportunidades de melhoria do sistema.

Aplicação das Avaliações

As duas avaliações propostas foram aplicadas em um software conhecido como Sweet Home 3D, um programa de *design de interiores Open Source*, distribuído sob licença GPLv2 (GNU, 1991), o qual permite criar um ambiente em um plano 2D e simultaneamente realizar visualização em 3D (Figura 4.1). Ele está disponível em 25 idiomas diferentes, incluindo português, e também permite criação de projetos no modo *online* por meio do seu site oficial¹. As suas principais características são:

- importar planta baixa como base para criação de um ambiente;
- inserir portas, janelas e móveis por meio do catálogo de objetos fornecido pelo software;
- alterar cores, formas, texturas e posições dos objetos inseridos no projeto;
- criar o projeto em 2D com visualização simultânea em 3D;
- adicionar comentários no projeto, como textos, linhas dimensionais ou direções;
- criar vídeos do projeto;
- configurar efeitos de luz natural e/ou artificial pela definição do horário do dia e a localização geográfica;
- exportar projeto em formato PDF, *bitmap* ou imagens vetoriais;
- permitir uso de *plug-ins* para adicionar recursos ao software.

4.1. Aplicação da avaliação heurística

Para realizar a avaliação heurística, foram convidados cinco especialistas da área de Interação Humano-Computador (sendo dois com formação em arquitetura em urbanismo), uma vez que esse é o número que proporciona melhores resultados e o máximo recomendado.

Para cada avaliador, foram enviados dois documentos, um contendo as especificações

¹ <http://www.sweethome3d.com/pt/features.jsp>

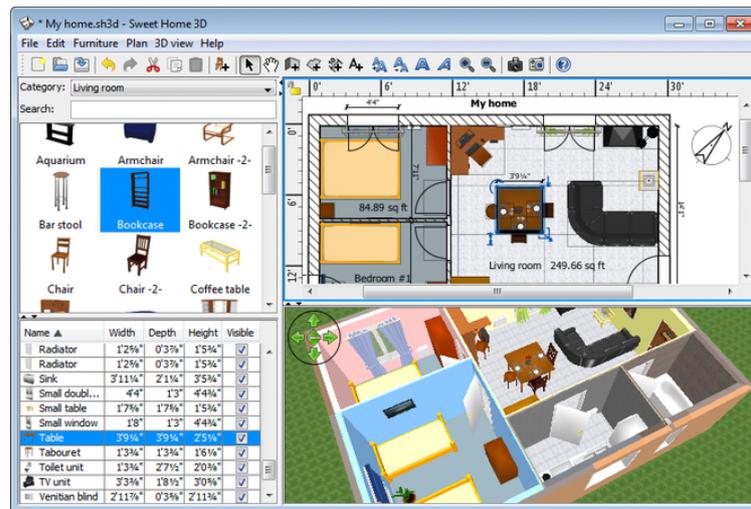


Figura 4.1. Interface do software Sweet Home 3D.

das dez heurísticas adaptadas nesse trabalho e, o outro com instruções sobre como a avaliação deve ser realizada e os passos para instalar o programa avaliado, que está descrito no Apêndice B.

Inicialmente, cada avaliador, separadamente, navegou na interface do programa até encontrar um problema de usabilidade. Feito isso, o problema encontrado foi classificado de acordo com as heurísticas propostas. Por exemplo, se não fosse localizada uma ferramenta de ajuda no programa, deveria categorizar esse problema como a heurística 10: ajuda e documentação. Esse passo foi realizado no mínimo três vezes.

Cada problema encontrado foi anotado em uma tabela e depois atribuído um grau de severidade variando de 0 a 4, sendo 0 sem importância; 1 cosmético, ou seja, não precisa ser corrigido, ao menos que se tenha tempo extra; 2 simples; 3 grave e; 4 catastrófico.

Também foram anotados o contexto, a causa, o efeito sobre o usuário e sobre a tarefa, e uma possível correção do problema. Essas informações facilitam a análise da falha e meios de corrigi-la.

4.2. Aplicação do teste de usabilidade

O teste de usabilidade foi aplicado ao final de um minicurso para capacitar usuários familiarizados com programas da área de *design e interiores*.

Para isso, a maior preocupação foi selecionar adequadamente esses usuários. Assim, foram priorizados alunos, professores e profissionais com conhecimento em programas da área de *design de interiores*. Para encontrar usuários que atendessem tal critério, entrou-se em contato com algumas instituições educacionais de Campo Mourão, como a Unicampo e o Senac.

Foi obtido retorno do coordenador do curso de Arquitetura e Urbanismo da Faculdade Unicampo, que confirmou a participação de trinta e dois alunos.

Também foram marcadas e realizadas duas reuniões com o coordenador de arquitetura. A primeira foi para mostrar a ideia inicial: um trabalho de conclusão de curso com o objetivo de realizar um teste de usabilidade, bem como a importância de uma quantidade mínima de doze usuários para realizarem o mesmo, a ideia do minicurso do software avaliado, desde que ao final, os alunos realizassem o questionário para o teste. Já a segunda foi para mostrar as atividades realizadas no curso, o material criado, os questionários e definir local e data.

Para a aplicação do minicurso, foi criado um material baseado no tutorial fornecido pelo próprio programa¹, que foi disponibilizado aos alunos em formato PDF.

Nesse material, foram oferecidos tutoriais e exemplos de como utilizar as ferramentas disponíveis e os processos básicos para a criação de um projeto no programa, tais como: a instalação do software em diferentes sistemas operacionais; como importar casa, desenhar e editar paredes, adicionar portas, janelas e móveis; importar modelos 3D e; desenhar cômodos.

A atividade escolhida foi a criação de uma casa de $19,50\text{ m}^2$, cuja planta baixa foi obtida pela internet, contendo a estrutura e medidas da casa: dois cômodos, um banheiro e uma cozinha junto com o quarto (Figura 4.2). Essa planta foi importada no software pelos usuários (Figura 4.3) e, em seguida foi proposto que cada um realizasse um projeto. A Figura 4.4 apresenta um dos projetos criados durante o minicurso. A Figura 4.4(a) mostra a parte frontal da casa, já a Figura 4.4(b) mostra a vista superior da casa, contendo várias móveis em cada cômodo, como a cama e a mesa de jantar.

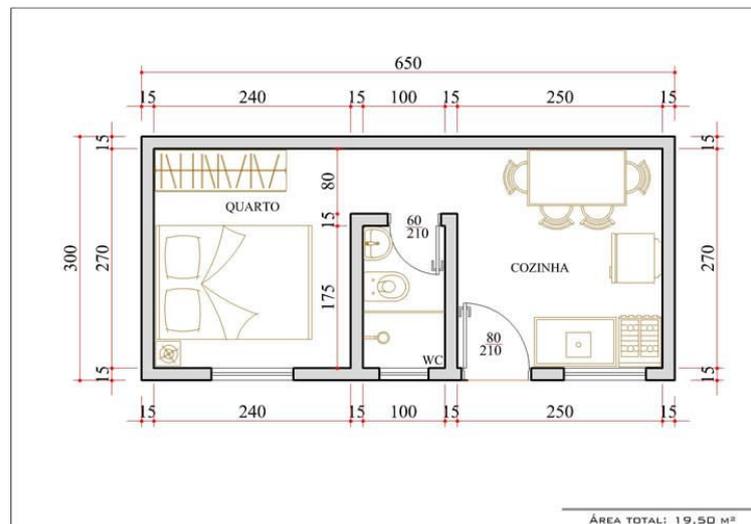


Figura 4.2. Planta baixa para o minicurso. Fonte: Construindo Decor².

Também foi organizada uma equipe responsável por auxiliar o minicurso e observar os usuários durante a aplicação do teste de usabilidade. Esses foram devidamente treinados, ou

¹ <http://www.sweethome3d.com/pt/userGuide.jsp>

² <http://construindodecor.com.br/edricula-planta-projeto-modelo/>

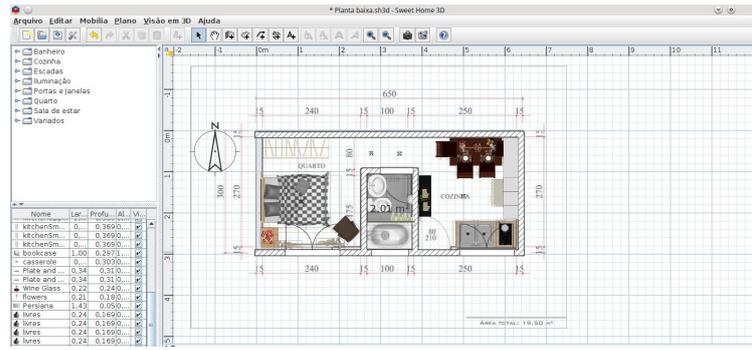
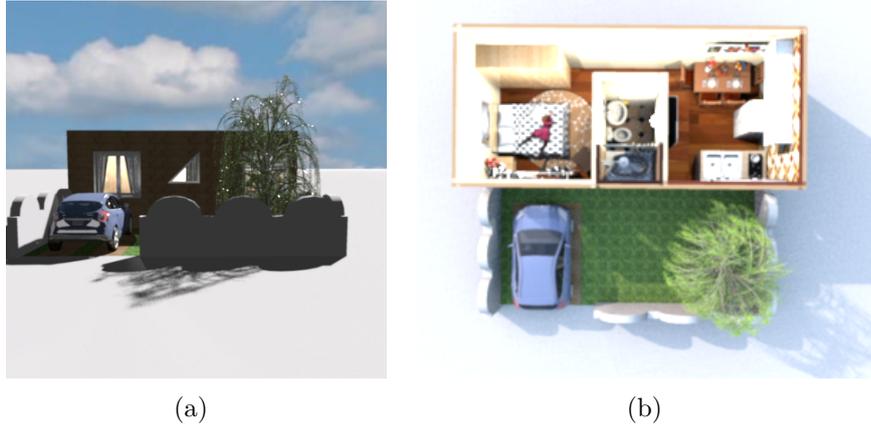


Figura 4.3. Planta baixa importada no software.



(a)

(b)

Figura 4.4. Projeção da planta baixa com vista (a) frontal e (b) superior.

seja, estudaram e aprenderam de forma aprofundada o funcionamento do programa escolhido.

Para a escolha da equipe, foram convidados seis alunos do Curso de Ciência da Computação da Universidade Tecnológica Federal do Paraná de Campo Mourão (UTFPR-CM), porém apenas três confirmaram participação.

O local utilizado para a avaliação foi o laboratório de informática da UTFPR-CM, pois a quantidade de computadores atende aos trinta e dois alunos que realizaram o teste. Foi criado um *login* de acesso para o uso dos computadores pelos alunos, com usuário e senha padronizado a todos. Além disso, o programa escolhido foi instalado em todas as máquinas.

4.2.1. Instrumento de avaliação

As dez questões do SUS (Tabela 2.1) foram disponibilizadas para todos os alunos por meio de um formulário online. Além dessas questões, foi inserida uma questão discursiva não obrigatória na qual o avaliador poderia descrever oportunidades de melhoria do sistema. O formulário aplicado está disponível no Apêndice A.

Como o estudo é realizado em apenas um grupo de alunos e em um único software, esse teste é considerado um estudo de caso, ou seja, seus resultados e conclusões não podem ser generalizados para outros programas.

Resultados e Discussão

Neste capítulo são apresentados os resultados gerados por meio de duas avaliações: heurística e teste de usabilidade.

5.1. Resultados da avaliação heurística

Dos cinco especialistas convidados, apenas três puderam realizar a avaliação heurística, sendo que um deles era um profissional da arquitetura. No total, onze problemas na interface do Sweet Home 3D foram encontrados e distribuídos nas seguintes heurísticas:

- compatibilidade do sistema com o mundo real;
- controle e liberdade do usuário;
- consistências e padrões;
- reconhecimento em vez de memorização;
- flexibilidade e eficiência de uso;
- estética de *design* minimalista;
- ajuda e documentação.

Os três problemas classificados na heurística de "compatibilidade do sistema com o mundo real" estão representados na Tabela 5.1.

Tabela 5.1. Problemas classificados na heurística 2 - Compatibilidade do sistema com o mundo real.

2.Compatibilidade do sistema com o mundo real			
Características	Problema 1	Problema 2	Problema 3
Contexto	Para alterar a cor da parede, o usuário escolhe uma cor do lado esquerdo e outra cor do lado direito. Mas, dependendo da posição não há um lado esquerdo e direito e sim interno e externo. Ficaria mais claro se a escolha da cor fosse pelo "lado interno"ou "lado externo".	O usuário deseja inserir um piso ou uma laje no ambiente superior. Para isso ele deverá usar o comando "inserir cômodo"que não reflete a ação "inserir piso".	Usuário clica com o botão direito sobre um objeto para ativar o módulo de edição das características do objeto.
Grau de severidade	2-simples.	3-grave.	2-simples.
Causa	Nomenclatura não corresponde a realidade.	Nome do comando confunde o usuário. o comando "Construir cômodo"na verdade insere o piso do ambiente, tanto no térreo quanto em outros pavimentos.	O botão direito pressionado sobre um objeto não dá opções sobre aquele objeto e sim de funcionalidades gerais. O clique duplo é que aciona as opções sobre o objeto.
Efeito sobre o usuário	Para saber qual é o lado esquerdo ou direito (quando o usuário queria alterar a cor do lado interno ou externo) o usuário precisa fazer uma tentativa e verificar se o resultado seria o esperado. Se não for o esperado, deverá refazer.	Frustração por achar que não consegue inserir um piso ou frustração na busca do comando por tentativas e erros.	O usuário não entende porque não pode acessar as opções de alteração das características do objeto com o botão direito, causando frustração.

Continuação da Tabela 5.1			
Características	Problema 1	Problema 2	Problema 3
Efeito sobre a tarefa	Perda de tempo na execução.	Perda de tempo na execução.	Usuário se sentirá perdido, tentando encontrar outra forma de alterar as características do objeto.
Correção possível	Mudar o nome da opção alterar cor para: "lado interno" e "lado externo".	Alterar o nome do comando "Inserir cômodo" para "inserir piso/laje" ou algo semelhante.	Implementar a funcionalidade no botão direito ao invés do duplo clique.
Fim da tabela 5.1			

Já para a heurística de "controle do usuário e liberdade", seus problemas estão descritos na Tabela 5.2.

Tabela 5.2. Problema classificado na heurística 3 - Controle do usuário e liberdade.

3. Controle do usuário e liberdade	
Características	Problema
Contexto	Desenhar paredes.
Grau de severidade	3-Grave.
Causa	O usuário não consegue desenhar paredes no contexto 3D. Se houver esta funcionalidade, ela não fica claro no sistema.
Efeito sobre o usuário	Frustração quando o desenho exige um desenho num plano tridimensional, como criar um plano inclinado.
Efeito sobre a tarefa	O usuário não consegue executar a tarefa.
Correção possível	Inserir a opção de desenhar no plano tridimensional (aquele que fica na parte direita inferior da tela, que, a meu ver, serve apenas para visualização).

Um problema foi classificado como "consistência e padrões" e esse é exposto na Tabela 5.4.

Tabela 5.3. Problema classificado na heurística 4 - Consistência e padrões.

4.Consistência e padrões	
Características	Problema
Contexto	Mesmo após a aplicação dos <i>patches</i> para correção do tamanho da fonte, algumas telas de diálogo permaneceram com fonte bastante diferente.
Grau de severidade	2-Simples.
Causa	A correção do tamanho da fonte proposto na documentação não foi abrangente a todas as telas de interface do sistema.
Efeito sobre o usuário	Desconforto/dificuldade para ler as mensagens na tela.
Efeito sobre a tarefa	O usuário perderá algum tempo até adequar a visão à situação. Uma vez que o sistema não contempla usuários com baixa visão, não afetará a correta execução da tarefa, apenas a tornará mais difícil.
Correção possível	Fazer com que a correção proposta na documentação afete o texto de todas as mensagens do sistema (inclusive telas de diálogo e não apenas a janela principal da aplicação).

Para a heurística "reconhecimento em vez de memorização", os avaliadores detectaram três problemas descritos na Tabela 5.4.

Tabela 5.4. Problemas classificados na heurística 5 - Reconhecimento em vez de memorização.

5.Reconhecimento em vez de memorização			
Características	Problema 1	Problema 2	Problema 3
Contexto	Inserção de dimensões exatas ao desenhar paredes ou cômodos.	Ao inserir uma porta, deveria ser possível selecionar facilmente como ela deve abrir (se a abertura é no sentido horário ou anti-horário). Isso seria resolvido facilmente se houvesse 1 botão do tipo "espelhar horizontalmente"o objeto na interface, ou similar.	Menu "Plano".
Grau de severidade	3-Grave	2-simples.	2-simples.

Continuação da Tabela 5.4			
Características	Problema 1	Problema 2	Problema 3
Causa	Não há no menu ou na caixa de ferramenta uma opção para o usuário digitar os valores das dimensões de uma determinada parede ou cômodo. O usuário precisa se lembrar que isso é feito acionando a tecla "Enter" enquanto está desenhando uma parede.	Não foi considerada possibilidade de inserir portas com aberturas em ambos os sentidos. E não foi encontrada essa possibilidade durante o uso.	A opção Exportar para o formato SVG deveria estar no menu "arquivo", mas está no menu "Plano". Aliás, todas as opções de exportar deveriam estar no menu "arquivo".
Efeito sobre o usuário	Frustração quando não se lembrar desta ação.	O usuário pode ficar frustrado por não ser capaz de reproduzir facilmente o ambiente que deseja.	O usuário está acostumado a outros softwares que tem a exportação de arquivos no menu "arquivo" e terá que ficar procurando esta opção ou pensará que não é possível.
Efeito sobre a tarefa	O usuário pode não realizar esta tarefa por falta de conhecimento.	Perda de tempo na tentativa de encontrar essa possibilidade. Não foi encontrada. Assim, foi necessário adaptar o ambiente projetado para atender as limitações encontradas no uso do software.	Perda de tempo na execução ou desistirá de realizar esta tarefa imaginando não haver a opção de exportação.
Correção possível	Deixar claro no sistema como construir paredes com dimensões precisas e exatas.	Inserir essa funcionalidade no sistema, ou torna-la mais visível como opção no menu de alterar o objeto.	Alterar as funcionalidades de exportação para o menu "Arquivo".
Fim da tabela 5.4			

O problema encontrado e classificado como "flexibilidade e eficiência de uso" está representado na Tabela 5.5.

Tabela 5.5. Problema classificado na heurística 7 - Flexibilidade e eficiência de uso.

7.Flexibilidade e eficiência de uso	
Características	Problema
Contexto	Inserção de paredes com uma dimensão específica.
Grau de severidade	3-Grave.
Causa	Não há alternativas para inserir as dimensões de uma parede, como um campo de entrada ou um comando específico no menu. O usuário vai descobrir como inserir as dimensões por acaso, usando a tecla "enter" para executar esta tarefa.
Efeito sobre o usuário	Frustração ao realizar o dimensionamento de paredes, achando que deverá arriscar um valor (que não é preciso) no plano milimetrado.
Efeito sobre a tarefa	Falta de segurança quanto a eficiência do sistema.
Correção possível	Deixar mais claro para os usuários como inserir paredes ou cômodos através de dimensões exatas e precisas.

Para heurística "estética e *design* minimalista" um problema foi encontrado, e esse está descrito na Tabela 5.6.

Tabela 5.6. Problema classificado na heurística 8 - Estética e *design* minimalista.

8.Estética e <i>design</i> minimalista.	
Características	Problema
Contexto	Ao iniciar o aplicativo, a Interface aparece com a lista de itens do menu à direita sobrepostos (imagem foi salva). Todos os textos da interface sofrem do mesmo problema. Mesmo maximizando a janela, ou clicando nos itens, os textos permanecem ilegíveis. A versão do programa é 5.7, em SO Windows 10. Resolução da tela 3480 x 2160.
Grau de severidade	4-Catastrófico.
Causa	Acredito que a resolução de tela utilizada seja a causa do problema.
Efeito sobre o usuário	Frustração, por não poder visualizar corretamente os itens do menu. Dificuldade (quase impossibilidade) de utilizar o sistema.

Continuação da Tabela 5.6	
Características	Problema
Efeito sobre a tarefa	A visualização força o usuário a alterar a resolução do dispositivo, caso queira utilizar o sistema. Se a resolução não for alterada, impossibilita o uso do sistema.
Correção possível	Ajustar a interface para que se adapte a diferentes resoluções de tela. Provavelmente o sistema não previa resoluções de tela tão altas.
Fim da tabela 5.6	

Finalmente, um único problema foi classificado como "ajuda e documentação", e esse está representado na Tabela 5.7.

Tabela 5.7. Problema classificado na heurística 10 - Ajuda e documentação.

10.Ajuda e documentação.	
Características	Problema
Contexto	Ao tentar solucionar o problema relatado de textos sobrepostos, busquei ajuda no fórum/ajuda. Uma solução reportada dizia para colocar um arquivo de configuração na pasta do aplicativo. Segui as instruções, porém a interface continuou sobreposta, agora com o texto em fonte de tamanho grande. O SO permanece o Windows 10, com a mesma resolução de tela. A solução correta (que funcionou) foi encontrada numa lista de discussão de usuários.
Grau de severidade	3-Grave.
Causa	A documentação afirma categoricamente que o problema é devido ao uso de resolução alta, e aponta uma solução não satisfatória.
Efeito sobre o usuário	Frustração e necessidade de buscar por soluções alternativas. Caso fosse um usuário menos experiente ou não habituado a buscar pelas soluções em fóruns de discussão, provavelmente teria abandonado o uso do sistema.
Efeito sobre a tarefa	Necessidade de inserir manualmente correções (patches) a fim de fazer o sistema funcionar adequadamente.
Correção possível	Inserir a descrição do problema, bem como a maneira correta de corrigir na documentação. Durante a instalação, tentar detectar (automaticamente, ou questionar o usuário - via imagens, por exemplo) se é necessário aplicar a correção. Caso seja, fornecer instruções claras de como proceder.

Nenhum dos avaliadores encontrou problema classificado como heurísticas 1 (status do sistema), 5 (prevenção de erros) e 9 (ajuda aos usuários no reconhecimento, diagnóstico e correção de erros). Porém, isso não garante que o software avaliado não apresente problemas de interface relacionados a essas 3 heurísticas, uma vez que diferentes especialistas poderiam apontar problemas que se enquadram nessas classificações.

Apesar de não ter sido considerado um erro, foi sugerido que o software disponibilize vídeos explicativos, uma vez que, atualmente, a ajuda é composta apenas por textos e muitos usuário desistem de ler e realizam qualquer tarefa em forma de "tentativa-e-erro". Essa sugestão está relacionada à heurística 10 (ajuda e documentação).

O grau de severidade facilita o estabelecimento de prioridades, determinando a ordem das correções dos problemas. Dessa forma, o pior problema encontrado, classificado como catastrófico, diz respeito à interface, que apresenta textos sobrepostos (Figura 5.1), impossibilitando a utilização do programa.

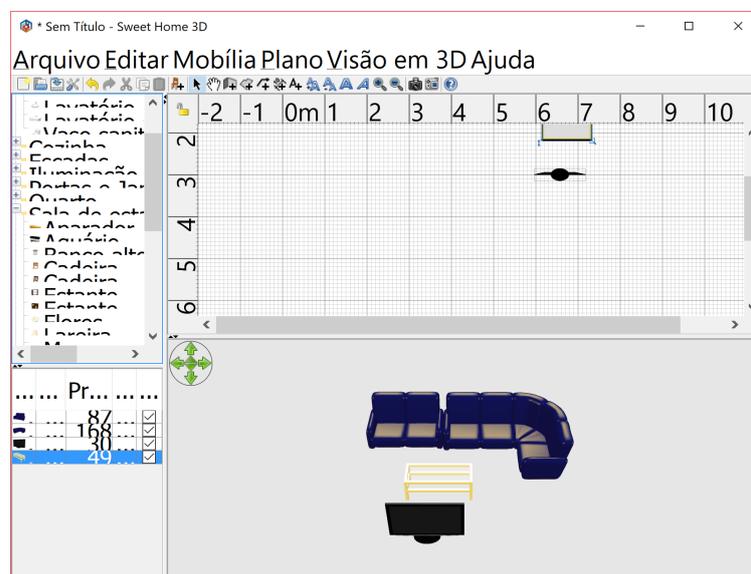


Figura 5.1. Problema de estética e *design* minimalista.

5.2. Resultados do teste de usabilidade

A Tabela 5.8 mostra as 32 respostas obtidas, distribuídas entre "discordo totalmente"(DT), "discordo"(D), "não concordo nem discordo"(N), "concordo"(C) e "concordo totalmente"(CT). Já a Figura 5.2 mostra a porcentagem dessas respostas.

5.3. Análise dos resultados por aluno

Utilizando a fórmula proposta pela técnica SUS, foi calculada a pontuação para cada avaliador (Figura 5.3).

Tabela 5.8. Resultado do SUS.

QUESTÕES	DT	D	N	C	CT	TOTAL
Eu gostaria de utilizar este sistema frequentemente	1	1	7	10	13	32
Eu achei o sistema desnecessariamente complexo	19	9	3	1	0	32
Eu achei o sistema fácil de utilizar	0	0	6	16	10	32
Eu acho que precisaria do apoio de um suporte técnico para ser possível utilizar este sistema	9	11	8	3	1	32
Eu achei que diversas funções deste sistema foram bem integradas	0	4	8	9	11	32
Eu achei que houve muita inconsistência neste sistema	7	13	5	7	0	32
Eu imaginaria que a maioria das pessoas aprenderiam a utilizar esse sistema rapidamente	0	1	5	14	12	32
Eu achei o sistema muito pesado para uso	17	9	2	3	1	32
Eu me senti muito confiante utilizando esse sistema	0	3	6	13	10	32
Eu precisei aprender uma série de coisas antes que eu pudesse continuar a utilizar esse sistema	13	10	3	5	1	32

**Figura 5.2.** Resultados dos SUS em porcentagem.

De acordo com Sauro (2009), após analisar resultados do SUS para mais de 5.000 usuários em 500 avaliações diferentes, pontuações média abaixo de 68 indicam que a usabilidade pode estar comprometida, sendo aconselhável realizar mudanças na interface desse sistema.

É possível observar que, do total de 32, a grande maioria (28 usuários) pontuou o sistema com resultados muito próximos ou superiores a 68. Dos usuários restantes, 3 pontuaram abaixo (usuário 3 com 62.5 pontos e os usuários 12 e 16 com 65 pontos), e apenas

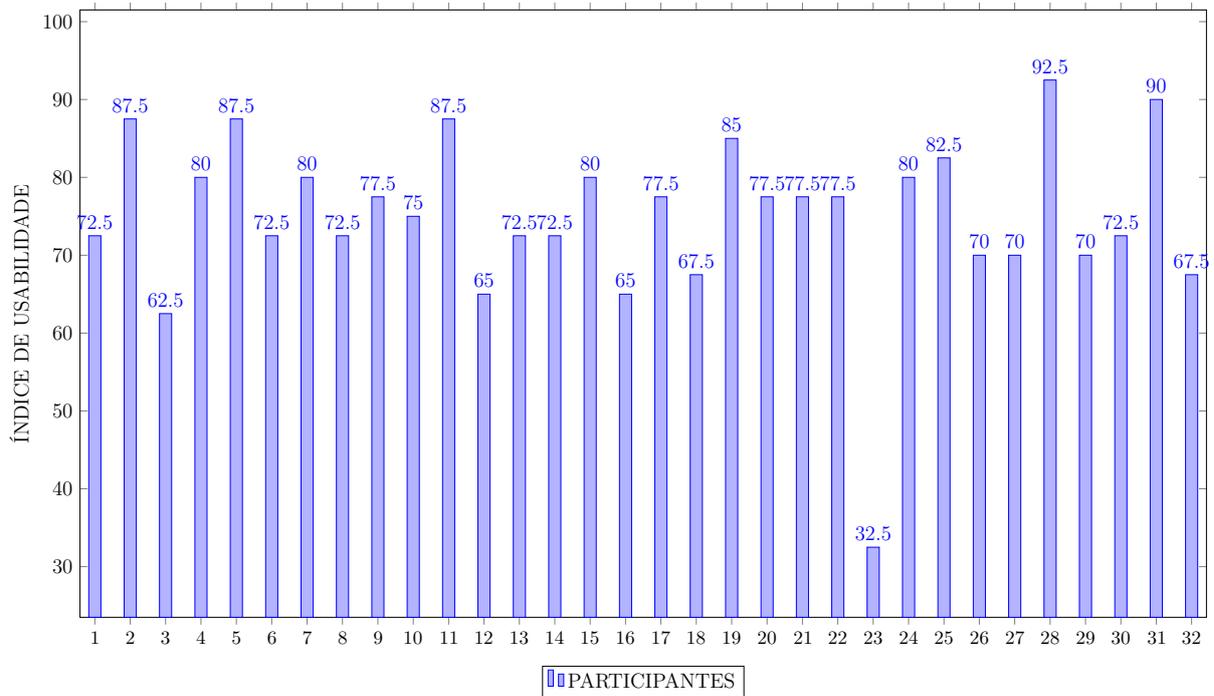


Figura 5.3. Aplicação do SUS para o programa Sweet Home 3D: Resultados por participante.

um usuário pontuou com um valor muito inferior (usuário 23 com 32.5 pontos).

Afim de investigar melhor os resultados dos usuários que menos pontuaram o sistema, esses foram separados em dois grupos, o primeiro contendo os usuários 3, 12 e 16, e o segundo apenas com o usuário 23. Nessa investigação, verificou-se se houve respostas para as questões discursivas e quais das dez questões foram respondidas de forma negativa.

Para o primeiro grupo, observou-se que a maioria dos usuários ficou satisfeitas com as funcionalidades do programa, em especial o usuário 16, que deixou uma opinião positiva quanto ao mesmo. Entretanto, eles apontaram a necessidade de apoio técnico para o uso do sistema, bem como melhorias que poderiam ser introduzidas.

O usuário 23, por sua vez, não respondeu a questão discursiva, que poderia contribuir para determinar qual foi a principal causa de frustração com o programa. Porém, levando em consideração as suas respostas na escala *Likert* (Tabela 5.9), foi possível concluir que esse usuário não se sentiu confiante em usar o software, pois ele achou que houve muitas inconsistências, além de ser um sistema complexo e pesado para uso, tendo a necessidade de um apoio técnico. Além disso, as questões pares são as que possuem maior peso para a avaliação negativa do sistema, e todas elas foram mal avaliadas por esse usuário, justificando uma pontuação tão inferior quando comparada com as outras.

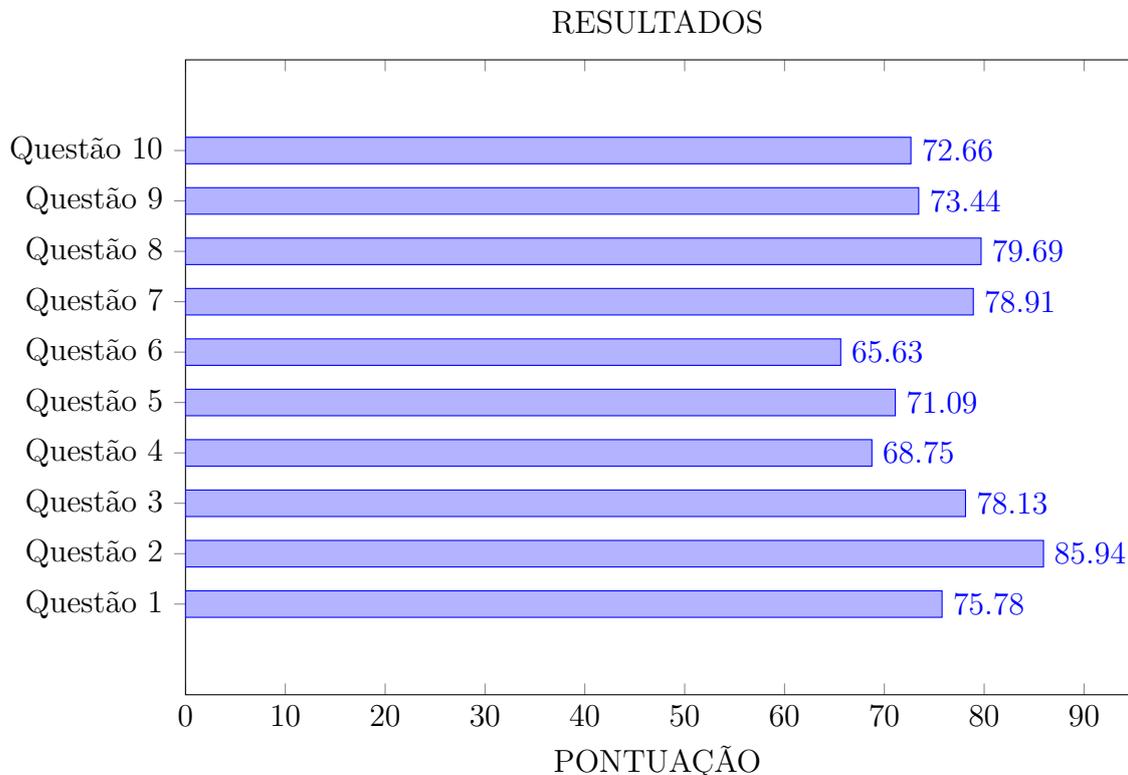
Apesar de um usuário ter avaliado negativamente, não influenciou na boa avaliação do sistema. A média das 32 avaliações foi de 75 pontos, indicando que a usabilidade do programa não está comprometida.

Tabela 5.9. Respostas de algumas questões do usuário 23.

Questão do SUS	Resposta do usuário
Eu achei o sistema desnecessariamente complexo	4-Concordo
Eu acho que precisaria do apoio de um suporte técnico para ser possível utilizar este sistema	5-Concordo totalmente
Eu achei que houve muitas inconsistências neste sistema	4-Concordo
Eu achei o sistema muito pesado para uso	5-Concordo totalmente
Eu precisei aprender uma série de coisas antes que eu pudesse continuar a utilizar esse sistema	4-Concordo

5.4. Análise dos resultados de cada questão

A pontuação obtida em cada questão foi ponderada para a escala de 0 a 100. A Figura 5.4 mostra esses resultados.

**Figura 5.4.** Resultados das questões do SUS.

Brooke et al. (1996) afirma que, médias inferiores a 50 pontos indicam usabilidade muito ruim, entre 51 e 64 indicam usabilidade ruim, entre 65 e 74 indicam usabilidade neutra, valores iguais ou superiores a 75 pontos indicam uma boa usabilidade e valores iguais ou superiores a 80 pontos indicam uma usabilidade muito boa.

Tenório et al. (2010) associa cada componente de qualidade indicado por Nielsen à média de um conjunto de questões específicas do SUS.

A **facilidade de aprendizagem do sistema** está associada as questões 3, 4, 7 e

10. A média dos resultados dessas questões no teste aplicado foi de 74.61, uma pontuação considerada neutra. Apesar dos usuários terem conseguido realizar a tarefa proposta, eles consideram que o sistema poderia ser mais intuitivo, facilitando o seu uso.

A **eficiência do sistema** é indicada pelas questões 5, 6 e 8. A média dos resultados dessas questões foi de 72.41, uma pontuação nem boa nem ruim. Dessa forma, é possível concluir que o sistema poderia apresentar um melhor nível de produtividade e desempenho.

A **facilidade de memorização** está associada à questão 2 e a sua pontuação foi de 85.94, um resultado considerado muito bom. Assim, é possível afirmar que os usuários foram capazes de realizar suas tarefas no sistema sem a necessidade de reaprender como interagir com o mesmo.

As **inconsistências do sistema** ou **minimação dos erros** são identificadas por meio da questão 6, cuja pontuação foi de 65.625, um valor considerado ruim. Assim, é possível afirmar que o sistema não apresenta soluções simples ou rápidas para eventuais erros. Alguns testes mais elaborados podem ser feitos no programa com o objetivo de encontrar outras irregularidades.

A **Satisfação dos usuários** com o sistema está representada nas questões 1, 4 e 9. A média dos resultados dessas questões foi de 72.66 pontos. Em geral, a satisfação refere-se ao nível de conforto que o usuário sente ao utilizar o sistema, de forma que ele alcance seus objetivos ao navegar no mesmo. Dessa forma, é possível concluir que os usuários poderiam estar mais confortáveis com o sistema se algumas mudanças na interface fossem realizadas.

Identificar oportunidade de melhoria do sistema: Embora a usabilidade do Sweet Home 3D tenha sido classificada como boa em quase todas as questões, a observação das respostas dadas às questões discursivas possibilitou analisar a existência de insatisfações e dificuldades em relação ao programa, como: a interface com um visual muito antigo e bordas quadradas; dificuldade em movimentar a planta da casa; necessidade de mais opções de visualização do projeto; o mal funcionamento da ferramenta *zoom*; necessidade de aumentar a sensibilidade do teclado para diminuir a dependência do *mouse* para ações com o sistema, ou seja, mais teclas de atalhos; e facilitar a inserção de telhados ao projeto.

5.5. Considerações finais

A aplicação da avaliação heurística apontou problemas específicos no Sweet Home 3D e a gravidade de cada um deles. A Tabela 5.10 sumariza esses problemas.

É possível observar que o pior problema encontrado enquadrou-se em estética e *design* minimalista, no qual a sobreposição da fonte do sistema impossibilitou o uso correto do mesmo, devido à alta resolução do monitor utilizado, porém esse problema foi realizado por um avaliador em um contexto específico. Já os erros classificados como simples e grave estão distribuídos em diferentes heurísticas, não existindo uma predominante.

Tabela 5.10. Problemas encontrados pela avaliação heurística.

Heurística	Grau de severidade		
	Simple	Grave	Catastrófico
Compatibilidade do sistema com o mundo real	2	1	0
Controle do usuário e liberdade	0	1	0
Consistência e padrões	1	0	0
Reconhecimento em vez de memorização	2	1	0
Flexibilidade e eficiência de uso	0	1	0
Estética e <i>design</i> minimalista	0	0	1
Ajuda e documentação	0	1	0
TOTAL	5	5	1

Já por meio do teste de usabilidade, foi possível realizar diferentes análises no Sweet Home 3D. Uma delas foi a classificação de acordo com os resultados de cada usuário, possibilitando a conclusão de uma boa usabilidade do sistema, mediante a uma média de 75 pontos obtida desses resultados. A outra foi a possibilidade de identificar recursos que devem ser melhorados no programa.

Além disso, foi possível classificar a usabilidade do Sweet Home 3D de acordo com as componentes de qualidade indicado por Nielsen (Tabela 5.11).

Tabela 5.11. Classificação do Sweet Home 3D de acordo com os resultados do teste de usabilidade.

Componente de qualidade	Pontuação	Classificação
Facilidade de aprendizagem	74.61	Neutra
Eficiência	72.41	Neutra
Facilidade de memorização	85.94	Muito boa
Inconsistências	65.62	Ruim
Satisfação dos usuários	72.66	Neutra

É possível analisar que quase todas as componentes são consideradas boas, entretanto existem inconsistências em algumas funcionalidades no programa que prejudicam o seu uso e que carecem de melhorias.

Dessa forma, as duas avaliações realizadas proporcionam diferentes resultados, a avaliação heurística estabelece erros específicos e o teste determina e classifica todos os problemas encontrados, retornando um valor capaz de avaliar o sistema em geral. Porém, ambas são capazes de identificar erros de usabilidade do sistema que prejudicam seu uso, definindo o que deve ser melhorado no mesmo.

Conclusão

Esta pesquisa tem como objetivo propor um conjunto de heurísticas e uma abordagem para aplicação de teste de usabilidade a programas de *design de interiores*. Esse conjunto de heurísticas é adaptado a partir das dez heurísticas de Nielsen, originalmente específicas para aplicações *Web*; e o teste trata-se da aplicação direta do questionário SUS. Com elas, foi possível realizar duas avaliações em um programa conhecido como Sweet Home 3D.

Os resultados dessas avaliações determinaram diferentes tipos de erros. O principal problema encontrado na avaliação heurística enquadrou-se em estética e *design* minimalista, porém esse problema não foi percebido pelos outros avaliadores, tão pouco durante o teste de usabilidade. Por outro lado, os problemas do teste não foram percebidos pelos especialistas, como o mal funcionamento da ferramenta *zoom*. Esse fato evidencia a importância de aplicar diferentes avaliações, determinando o maior número de falhas para melhor classificar a interface do sistema.

A principal contribuição desse trabalho foi o desenvolvimento das heurísticas para o contexto de *design de interiores*. O SUS mostrou-se um excelente modelo de questionário que permitiu classificar a qualidade da interface do Sweet Home 3D, bem como identificar as componentes de qualidade que necessitam de mudança e, ainda avaliar a opinião e classificação do sistema por cada usuário.

Como trabalhos futuros, propõe-se aplicar o conjunto de heurísticas adaptado nesse trabalho para outros softwares da área, a fim de validar sua aplicabilidade e utilidade nesse contexto. Também, é possível aplicar o teste de usabilidade usando outros questionários, quer seja na forma como são originalmente propostos, quer seja adaptando-os para o contexto de software de interiores, como por exemplo o QUIZ (Jeff Sauro, 2013) e o USE (LUND, 2001), a fim de observar se os resultados são condizentes com a qualidade aferida pelo SUS nesse trabalho.

Referências

- ALVES, João Marcus; SAVARIS, Alexandre; WANGENHEIM, Christiane Gresse von; WANGENHEIM, Aldo von. Criando e validando um checklist de avaliação heurística a para sistemas de laboratórios clínicos. *Computer on the Beach*, p. 1–10, 2015.
- ANDRADE, Vinicius Abib. avaliação da ferramenta owncloud como solução de cloud computing privada. 2014.
- ASSOCIAÇÃO BRASILEIRA DE NORMAS TÉCNICAS (ABNT). *NBR 9126-1: Engenharia de software - Qualidade de produto parte 1: Modelo de qualidade*. 2003.
- AUTODESK. *Família Revit. Software BIM. Autodesk.* 2018. <https://www.autodesk.com.br/products/revit-family/overview>. Acessado em 20-04-2018.
- AUTODESK. *Produtos. Novas versões. Autodesk.* 2018. <https://www.autodesk.com.br/products>. Acessado em 20-04-2018.
- BOUCINHA, Rafael Marimon; TAROUCO, Liane Margarida Rockenbach. Avaliação de ambiente virtual de aprendizagem com o uso do sus-system usability scale. *RENOTE*, v. 11, n. 3, 2013.
- BROOKE, John et al. Sus-a quick and dirty usability scale. *Usability evaluation in industry*, London–, v. 189, n. 194, p. 4–7, 1996.
- CÔRTEZ, Mario Lúcio; CHIOSSI, Thelma C. dos Santos. *Modelos de qualidade de software*. [S.l.]: Unicamp, 2001. ISBN 9788425223587.
- DUARTE, Jose Renato. *História da Computação Gráfica e Maquete Eletrônica em PC*. 2014. <http://www.jrrio.com.br/computacao-grafica/historia-da-cg-e-maquete-eletronica-em-pc.html>. Acessado em 28-03-2017.
- GARCIA, Gabriel Guimarães; JUNIOR, Luiz Antonio da Ponte. Visualização de dados institucionais. 2017.
- GENERAL PUBLIC LICENSE (GNU). *GNU General Public License Version 2*. 1991.
- GIBBS, Jenny. *Design de interiores: guia útil para estudantes e profissionais*. [S.l.]: Editorial Gustavo Gili, S.L., 2010. ISBN 9788425223587.
- GLORIA, Heloisa de Souza. Avaliação de um conjunto de heurísticas de usabilidade para aplicativos de smartphones na área da saúde por meio de testes de usabilidade. p. 1–75, 2015.
- HUA, Fang; KUI, Zhang Qiu. Study on visual expression of cad interior design drawing. In: *International Conference on Intelligent Systems Design and Engineering Applications (ISDEA)*. [S.l.: s.n.], 2015. p. 782–785.

- Jeff Sauro. *Sweet Home 3D*. 2013. <https://measuringu.com/usability-quiz/>. Acessado em 11-06-2018.
- KOSCIANSKI, André; SOARES, Michel dos Santos. Qualidade de software. *São Paulo: Novatec*, v. 3, 2006.
- LUND, Arnold. Measuring usability with the use questionnaire. v. 8, 01 2001.
- MACHADO, Lais; FERREIRA, Evelise Pereira; VERGARA, Lizandra Garcia Lupi. Métodos de avaliação de usabilidade: Características e aplicações. *3o. CONEPRO-SUL, Joinville. 3o. CONEPROSUL*, 2014.
- MACIEL, Cristiano; NOGUEIRA, José Luis T; CIUFFO, Leandro Neumann; GARCIA, Ana Cristina Bicharra. Avaliação heurística de sítios na web. *Escola de Informática do SBC-Centroeste*, v. 7, 2004.
- MEDEIROS, Marco Aurélio; CYBIS, Walter de Abreu. Método de avaliação de usabilidade de software a partir da satisfação de usuários e da aplicação de quesitos da norma iso 9241. In: *IHC'2000-III Workshop sobre Fatores Humanos em Sistemas de Computação*. [S.l.: s.n.], 2000. p. 93-101.
- MEDEIROS, Marco Aurélio et al. Iso 9241: uma proposta de utilização da norma para avaliação do grau de satisfação de usuários de software. Florianópolis, SC, 1999.
- MONTENEGRO, Gildo. *A invenção do projeto*. [S.l.]: Edgard Blucher, 1987. ISBN 9788521200079.
- NIELSEN, Jakob; MACK, Robert L. *Usability Inspection Methods Computer*. [S.l.]: John Wiley & Sons, 1994. ISBN 0471018775.
- OLIVEIRA, Igor Aguiar. Interface de usuário: A interação homem-computador através dos tempos. *Olhar Científico*, v. 1, n. 2, p. 178-184, 2011.
- PARREIRA JÚNIOR, Walteno Martins; PRADELA, Izaura Pereira; OLIVEIRA, Lucineida Nara de Andrade. O uso da norma 14598 na avaliação de software com relação à qualidade. *Intercursos - Revista das unidades acadêmicas da fundação educacional de Ituiutaba*, v. 8, n. 1, p. 1-11, 2009.
- PREECE, Jennifer; ROGERS, Yvone; SHARP, Helen. *Design de Interação*. [S.l.]: Bookman, 2005. ISBN 9788536304946.
- PULE, John. *A History of Interior Design*. [S.l.]: Laurence King Publishing LTD, 2005. ISBN 1-85669-418-6.
- SALIS, Enrico. *As etapas de um projeto de interiores*. 2016. <https://www.linkedin.com/pulse/etapas-de-um-projeto-interiores-enrico-salis>. Acessado em 20-04-2018.
- SANTINHO, Miguel. Avaliação heurística e testes com utilizadores: dois métodos, dois resultados. 2001.
- SAURO, Jeff. *Measuring Usability with the System Usability Scale (SUS)*. 2009. <https://measuringu.com/sus/>. Acessado em 08-04-2018.

SKETCHUP. *3D modeling for everyone*. 2018. <https://www.sketchup.com/>. Acessado em 20-04-2018.

Sweet Home 3D. *Sweet Home 3D*. 2018. <http://www.sweethome3d.com/pt/>. Acessado em 20-04-2018.

Sweet Home 3D. *Sweet Home 3D: Recursos*. 2018. <<http://www.sweethome3d.com/pt/features.jsp>>. Acessado em 20-04-2018.

TENÓRIO, Josceli Maria; COHRS, Frederico Molina; SDEPANIAN, Vera Lúcia; PISA, Ivan Torres; MARIN, Heimar de Fátima. Desenvolvimento e avaliação de um protocolo eletrônico para atendimento e monitoramento do paciente com doença celíaca. *Revista de Informática Teórica e Aplicada*, v. 17, n. 2, p. 210–220, 2010.

VEIGA, Marília. *O que faz um designer de interiores?* 2015. <http://decoracaoosaopaulo.com.br/decoracao/o-que-faz-um-designer-de-interiores/>. Acessado em 20-04-2018.

VELA, João Carlos; NASCIMENTO, Beatriz Andrielly de Souza; FEDER, Franciane; ISTVANDIC, Mikaella; JAHN, Ana Claudia. Design de interiores: Entre o bacharelado e o tecnológico: uma reflexão. In: *Congresso brasileiro de pesquisa e desenvolvimento em design*. [s.n.], 2016. v. 2, n. 9, p. 1–12. Disponível em: <<http://pdf.blucher.com.br/s3-sa-east-1.amazonaws.com/designproceedings/ped2016/0237.pdf>>.

WINCKLER, Marco; PIMENTA, Marcelo Soares. Avaliação de usabilidade de sites web. *Escola de Informática da SBC SUL (ERI 2002) ed. Porto Alegre: Sociedade Brasileira de Computação (SBC)*, v. 1, p. 85–137, 2002.

Apêndices

Formulário do Teste de Usabilidade

Esse formulário está disponível no link: <<https://docs.google.com/forms/d/e/1FAIpQLScwnmtMuNgx8cE4rb-5epA7TUayXF-2yxNpSqyoC7JbWa5z6g/closedform>>.

1. Eu gostaria de utilizar este sistema frequentemente *

Marcar apenas uma oval.

	1	2	3	4	5	
Discordo Totalmente	<input type="radio"/>	Concordo Totalmente				

2. Eu achei o sistema desnecessariamente complexo *

Marcar apenas uma oval.

	1	2	3	4	5	
Discordo Totalmente	<input type="radio"/>	Concordo Totalmente				

3. Eu achei o sistema fácil de utilizar *

Marcar apenas uma oval.

	1	2	3	4	5	
Discordo Totalmente	<input type="radio"/>	Concordo Totalmente				

4. Eu acho que precisaria do apoio de um suporte técnico para ser possível utilizar este sistema *

Marcar apenas uma oval.

	1	2	3	4	5	
Discordo Totalmente	<input type="radio"/>	Concordo Totalmente				

Opiniões sobre o Sweet Home 3D

11. Eu identifiquei oportunidades de melhoria do sistema

Instruções para realizar a avaliação heurística

A seguir estão descritas as instruções criadas para os avaliadores da avaliação heurística:

Os resultados dessa avaliação contribuirão para encontrar características problemas de usabilidade na interface do Sweet Home 3D.

O primeiro passo é instalar o Sweet Home 3D, que está disponível para download em: <http://www.sweethome3d.com/pt/download.jsp>. Se você utiliza o linux, não é necessário instalar o programa, apenas executar o script contido na pasta após o seu download. Caso não consiga acessar ou instalar o programa, entre em contato com: noemischerer13@gmail.com.

Existe uma versão web do programa, porém não é completa, por isso não é aconselhável utilizá-la para avaliação.

Foram adaptadas as dez heurísticas de Nielsen para software de design de interiores, elas estão disponíveis no documento Heurísticas.pdf. O segundo passo é ler e entender todas essas heurísticas descritas no documento.

Entendidas as heurísticas, você deverá encontrar no máximo 3 problemas de usabilidade na interface do Sweet Home 3D, e depois classificá-los em uma das dez heurísticas. Uma heurística pode possuir mais de um problema.

Cada problema deve ser anotado em um formulário diferente, que está disponível em: Formulário. Se você encontrar 3 problemas, deverá responder o questionário 3 vezes, um para cada problema.

Boa Avaliação!