



UNIVERSIDADE  
FEDERAL  
DE PERNAMBUCO



UNIVERSIDADE FEDERAL DE PERNAMBUCO

Graduação em Ciência da Computação

**FILIFE FERNANDO CABRAL DE MELO SOUZA**

**Utilização de Ferramentas de Gerenciamento e  
Visualização de Banco de Dados em Dispositivos Móveis e  
suas Dificuldades**

RECIFE

2016

FILIPPE FERNANDO CABRAL DE MELO SOUZA

**Utilização de Ferramentas de Gerenciamento e  
Visualização de Banco de Dados em Dispositivos Móveis e  
suas Dificuldades**

Trabalho de conclusão de curso apresentado à disciplina de TCC para obter o grau de Bacharel em Ciência da Computação sob a orientação do professor Fernando da Fonseca de Souza.

RECIFE

2016

## **AGRADECIMENTOS**

Agradeço a todos que me apoiarem durante a elaboração deste trabalho e durante todos os momentos difíceis que passei na universidade, me dando todo o suporte necessário para seguir em frente. Agradeço principalmente a minha namorada Márcia Cavalcante, a minha mãe Monica Lira, ao meu pai Fernando Souza, aos meus irmãos Áureo Souza e Leonardo Souza e aos meus amigos. Agradeço também ao meu orientador Fernando Fonseca por toda a orientação e suporte fornecido durante o desenvolvimento deste trabalho.

## RESUMO

O banco de dados é uma parte essencial de toda aplicação que envolve armazenamento, manipulação e consumo de dados por usuários, pois é necessário adicionar estrutura e organização aos dados para facilitar futuras consultas.

Com a popularização dos dispositivos móveis e o crescente uso nas mais diversas áreas surge a necessidade de ferramentas para a visualização e manipulação de dados a partir desses dispositivos, em busca de facilitar o acesso a simples consultas, comandos de alteração de dados, mudanças estruturais, e até de atividades de gerenciamento do próprio sistema de banco de dados. Esse tipo de ferramenta seria bastante útil tanto para administradores como para usuários que acessam diretamente o banco de dados.

No entanto, existem dificuldades na visualização e manipulação de dados a partir de dispositivos móveis referentes à intuitividade da interface com o usuário, facilidade de entrada de dados e a forma com que os dados são apresentados. Isso acontece principalmente por causa das restrições inerentes à dimensão reduzida dos dispositivos quando comparados a computadores e *notebooks* e à forma de entrada de dados menos confortável para escrever consultas um pouco maiores e mais complexas que o normal.

Neste trabalho serão analisadas essas e outras limitações de dispositivos móveis, como esses problemas são contornados por outras aplicações no mercado e como otimizar a usabilidade da interface de forma que facilite e torne mais confortável a entrada de dados e de que maneira a apresentação de dados pode ser mais amigável. Ao final do estudo, uma ferramenta será elaborada aplicando os aspectos analisados durante o desenvolvimento do trabalho.

**Palavras-chave:** Interface de Usuário, Usabilidade, Mobilidade.

## ÍNDICE DE FIGURAS

Figura 3.1 - Aspectos de aceitabilidade de um sistema segundo Shackel. ....	26
Figura 3.2 - Aspectos de aceitabilidade de um sistema segundo Nielsen. ....	27
Figura 4.1 - <i>Feedback</i> inapropriado na execução de comandos de alteração estrutural. ....	45
Figura 4.2 - <i>Feedback</i> inapropriado na execução de comandos de alteração de dados. ....	46
Figura 4.3 - Tela de cadastro de conexões não possui um botão para confirmar a criação. ....	47
Figura 4.4 - Tela de seleção de conexões e cadastro de novas conexões. ....	53
Figura 4.5 - Tela de escrita e execução de consultas. ....	54
Figura 4.6 - Menu lateral com a lista de tabelas, <i>views</i> e <i>procedures</i> . ....	55
Figura 4.7 - Menu local da conexão. ....	59
Figura 4.8 - Opção de ir para interface de escrita de consultas dentro do menu local da tela. ....	60
Figura 4.9 - Menu local da conexão. ....	66
Figura 4.10 - Tela de adição de coluna. ....	67
Figura 4.11 - Mensagem de erro apresentada na execução de um comando <i>select into</i> . ....	77
Figura 4.12 - Erro apresentado ao tentar cadastrar um tipo <i>VARCHAR</i> em um campo <i>INT</i> . ....	78
Figura 5.1 - Tela de seleção de bancos de dados existentes na conexão. ....	89
Figura 5.2 - <i>Feedback</i> apresentado após a inserção de dados com sucesso. ....	90
Figura 5.3 - Execução de consulta com a opção de cancelamento. ....	91
Figura 5.4 - Escrita de consultas com a opção de auto completar. ....	91
Figura 5.5 - Botão de <i>zoom</i> disponível na tela de visualização de dados. ....	92
Figura 5.6 - Cadastro de colunas do tipo <i>date</i> . ....	93

## ÍNDICE DE GRÁFICOS

Gráfico 2.1 - Usuários de Smartphone e Penetração no Mundo. ....	16
Gráfico 2.2 - Número de aplicativos disponíveis por loja de 2010 a 2014. ....	18

## ÍNDICE DE TABELAS

Tabela 2.1 - Países Ranqueados pelo Número de Usuários de Smartphone. ....	17
Tabela 2.2 - Venda de <i>smartphones</i> para usuários finais por SO. ....	18

## ÍNDICE DE QUADROS

Quadro 3.1 - Aplicação do teste de usabilidade. ....	37
Quadro 4.1 - Resumo das avaliações realizadas no capítulo.....	86
Quadro 5.1 - Resumo de análise realizada em todos os aplicativos. ....	104

# SUMÁRIO

1. Introdução.....	12
1.1. Motivação.....	12
1.2. Objetivos .....	13
1.2.1. Objetivos Específicos.....	13
1.3. Estrutura do Trabalho .....	13
2. Dispositivos Móveis.....	15
2.1. Cenário Atual.....	15
2.2. Plataformas .....	17
2.3. Limitações .....	19
2.3.1. Dimensões da Tela.....	19
2.3.2. Entrada de Dados.....	20
2.3.3. Energia .....	20
2.4. Portabilidade x Mobilidade .....	21
2.5. Banco de Dados em Dispositivos Móveis.....	22
2.6. Considerações Finais .....	24
3. Usabilidade.....	25
3.1. Definições de Usabilidade.....	26
3.1.1. Shackel .....	26
3.1.2. Nielsen .....	27
3.1.3. Jordan.....	29
3.2. Avaliação de Usabilidade.....	31
3.2.1. Métodos de Avaliação Analíticos.....	32

3.2.1.1.	Avaliação Heurística.....	33
3.2.1.2.	Percurso Cognitivo .....	34
3.2.2.	Métodos de Avaliação Empíricos.....	36
3.3.	Considerações Finais .....	39
4.	Análise de Aplicações Existentes .....	40
4.1.	Forma de Análise e Avaliação.....	40
4.1.1.	<i>Checklist</i> de Funções .....	41
4.1.2.	Avaliação Heurística.....	41
4.1.3.	Percurso Cognitivo .....	42
4.2.	Aplicações Selecionadas .....	43
4.2.1.	SQL Client.....	43
4.2.1.1.	<i>Checklist</i> de Funções .....	44
4.2.1.2.	Avaliação Heurística.....	44
4.2.1.3.	Percurso Cognitivo .....	48
4.2.2.	Mobile SQL Client .....	51
4.2.2.1.	<i>Checklist</i> de Funções .....	52
4.2.2.2.	Avaliação Heurística.....	52
4.2.2.3.	Percurso Cognitivo .....	55
4.2.3.	SQL Studio Lite.....	57
4.2.3.1.	<i>Checklist</i> de Funções .....	58
4.2.3.2.	Avaliação Heurística.....	58
4.2.3.3.	Percurso Cognitivo .....	61
4.2.4.	Mobile SQL Server Manager .....	64
4.2.4.1.	<i>Checklist</i> de Funções .....	65
4.2.4.2.	Avaliação Heurística.....	65
4.2.4.3.	Percurso cognitivo .....	68

4.2.5.	RemoDB.....	76
4.2.5.1.	<i>Checklist</i> de Funções .....	76
4.2.5.2.	Avaliação Heurística.....	77
4.2.5.3.	Percurso Cognitivo .....	78
4.3.	Considerações Finais .....	84
5.	Solução Desenvolvida: SQL Manager .....	87
5.1.	Detalhes do Desenvolvimento.....	87
5.2.	Melhorias Implementadas .....	88
5.3.	<i>Checklist</i> de Funções.....	94
5.4.	Avaliação Heurística .....	94
5.5.	Percurso Cognitivo .....	95
5.6.	Considerações Finais .....	104
6.	Considerações Finais .....	105
6.1.	Limitações .....	105
6.2.	Trabalhos Futuros .....	106
	Referências .....	107

## 1. Introdução

Os avanços que vêm ocorrendo ao longo dos últimos anos nas tecnologias de hardware e de rede estão tornando os dispositivos móveis cada vez mais capacitados para serem usados em diversas atividades que antes eram realizadas apenas por computadores e *notebooks*. Esses avanços são decorrentes da constante melhora dos componentes físicos utilizados nos celulares, como memória RAM, espaço em disco, melhoria no consumo de energia e poder de processamento, tudo isso sem interferir na mobilidade desses dispositivos. Além disso a evolução na tecnologia de comunicação telefônica (3G<sup>1</sup>, 4G<sup>2</sup>, redes wireless) vem tornando cada vez mais comum a transmissão de grandes volumes de dados em qualquer lugar.

Todos esses avanços aliados à crescente popularização dos *smartphones* e à facilidade de publicação de aplicativos nos principais sistemas operacionais desses aparelhos fazem com que surjam cada vez mais aplicações para ajudar nas mais variadas atividades do cotidiano, aproveitando a mobilidade que é proporcionada.

Entre as possíveis aplicações que podem ser úteis em dispositivos móveis estão as relacionadas a sistemas de gerenciamento de banco de dados, como bancos de dados móveis e ferramentas de acesso a banco de dados em servidores remotos. Essas aplicações podem vir a ser muito convenientes para administradores de bancos de dados ou usuários que utilizem o banco de dados diretamente (sem fazer uso de nenhum sistema que consuma os dados), seja para fazer alterações estruturais ou de dados, verificar possíveis problemas de desempenho no servidor, ou realizar consultas remotamente de maneira prática e rápida.

### 1.1. Motivação

Existem aplicações de gerenciamento, visualização e manipulação de dados oriundos de sistemas de gerenciamento de banco dados para dispositivos móveis nas mais diversas plataformas, porém ainda existem dificuldades na visualização e manipulação de dados a partir de dispositivos móveis referentes à intuitividade da interface, à facilidade de entrada de dados e à forma com que os dados são apresentados. Principalmente por causa da dimensão reduzida

---

<sup>1</sup> 3G é a terceira geração de padrões e tecnologias de telefonia móvel.

<sup>2</sup> 4G é a quarta geração de padrões e tecnologias de telefonia móvel.

quando comparados a computadores e *notebooks* e a forma de entrada de dados menos confortável para escrever consultas, por se tratar de dispositivos que precisam ser portáteis.

## 1.2. Objetivos

Este trabalho tem como objetivo analisar e buscar formas de contornar as limitações e dificuldades referentes à utilização de aplicações móveis com foco em visualização e manipulação de dados oriundos de sistemas de gerenciamento de banco de dados. Além da realização de outras possíveis melhorias referentes à interface e à entrada de dados, comparado a outros aplicativos nesse contexto.

### 1.2.1. Objetivos Específicos

Para atingir o objetivo geral do trabalho e pôr em prova o resultado do estudo têm-se como objetivos específicos:

- Realizar uma análise sobre as principais aplicações disponíveis no mercado para a plataforma Android<sup>3</sup> de modo a identificar as limitações existentes e problemas de usabilidade; e
- Pôr em prática o resultado do estudo e da análise do mercado para especificar e desenvolver uma aplicação Android voltada para sistemas de bancos de dados Microsoft SQL Server<sup>4</sup> com o objetivo de facilitar e tornar mais comum o uso desse tipo de aplicação.

## 1.3. Estrutura do Trabalho

Esse trabalho será dividido em seis (6) capítulos. Além do capítulo de introdução, têm-se:

- Dispositivos Móveis - Apresenta características importantes sobre esses aparelhos, cenário atual, as principais limitações que podem influenciar no desenvolvimento de aplicativos e conceitos sobre sistemas de bancos de dados em dispositivos móveis;

---

<sup>3</sup> <https://www.android.com/>

<sup>4</sup> <https://www.microsoft.com/pt-br/server-cloud/products/sql-server/>

- Usabilidade - Discorre sobre os conceitos de usabilidade e de que formas se pode avaliar e melhorar a usabilidade de interfaces;
- Análise de aplicações existentes - Destaca algumas das aplicações existentes, os critérios para seleção, como estas aplicações serão avaliadas e o resultado dos testes e análises sobre elas;
- Solução desenvolvida - Expõe a solução desenvolvida, pontos que foram melhorados em comparação com as soluções apresentadas e análise da solução desenvolvida;
- Considerações finais - Mostra as conclusões do trabalho, suas limitações e apresenta sugestões de trabalhos futuros.

Por último, são listadas as referências bibliográficas utilizadas.

## 2. Dispositivos Móveis

Dispositivos móveis para computação são dispositivos com a capacidade de realizar processamento, trocar informações via rede e de ser transportado facilmente por seu usuário. Para isso, é importante que o dispositivo computacional tenha tamanho reduzido e não necessite de cabos para conectá-lo à rede de dados ou à fonte de energia elétrica. Assim, equipamentos deste tipo devem ter as seguintes características: menores dimensões comparados a computadores convencionais, possuir autonomia de bateria para evitar a necessidade de conexões à rede elétrica com cabos que limitariam muito a mobilidade, e ter acesso a informações por meio de tecnologias de redes sem fio (FIGUEIREDO; NAKAMURA, 2003).

### 2.1. Cenário Atual

Existem vários dispositivos que podem se enquadrar nessa descrição, entre eles PDA (*Personal Digital Assistant*), *smartphones* (telefone inteligente), *tablets* e *smartwatches* (relógio inteligente). Todos possuem tecnologia *touch screen*, o que facilita a interação com o usuário e otimiza o espaço do dispositivo para a tela. Esses dispositivos têm características similares, sendo diferenciados principalmente pelas tecnologias utilizadas, dimensões e, conseqüentemente, pelo seu propósito.

PDA é um computador de dimensões reduzidas, dotado de capacidade computacional e possibilidade de conexão com redes sem fio. Os PDA foram muito utilizados no passado e pode se dizer que os *smartphones* surgiram de uma combinação dos PDA e dos telefones celulares convencionais.

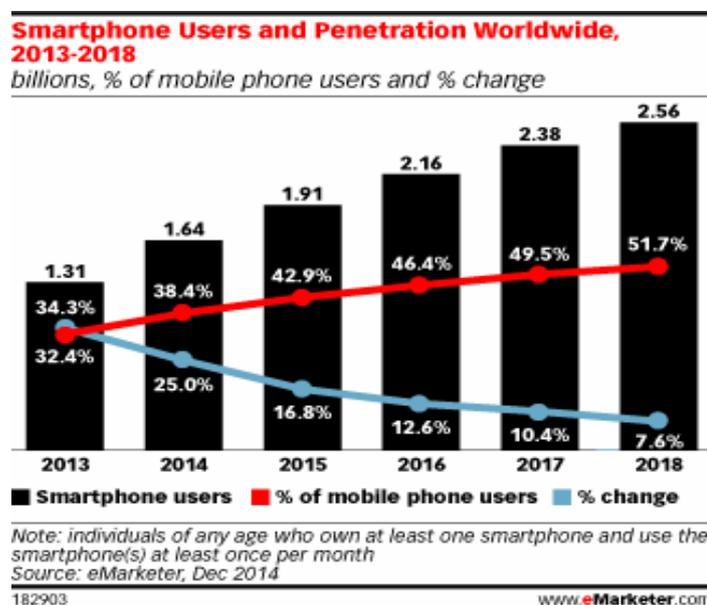
*Smartphone* é uma nova linhagem de telefones celulares que possui uma série de tecnologias integradas no mesmo aparelho, sendo capaz de atividades de computação em geral. É o principal dispositivo móvel utilizado atualmente e vem ganhando cada vez mais popularidade, já sendo largamente utilizado em diversas áreas e atividades, além da própria telefonia móvel.

*Tablets* são computadores portáteis semelhantes ao *smartphone*, porém com maiores dimensões e capacidades, além do fato de não serem destinados à telefonia móvel. Eles são muito usados para propósito geral, entretenimento e como equipamento de trabalho, por exemplo para edição de documentos e apresentações.

Já os *smarwatches* são dispositivos de pulso computadorizados que são comparáveis a PDA em suas funções. Ainda é uma tecnologia recente que não atingiu o nível de popularidade dos outros dispositivos citados.

Um estudo realizado pela eMarketer<sup>5</sup> em dezembro de 2014 mostra que o crescimento no número de usuários de *smartphones* de 2013 para 2014 foi de 25%, passando de 1.31 bilhões para 1.64 bilhões, cerca de 38.4% da população mundial. O estudo prevê que em 2016 haverá um crescimento de 12.6% em relação a 2015, alcançando 2.16 bilhões de pessoas ao redor do mundo, 46.4% da população. E em 2018 estima-se que esse número atinja 2.56 bilhões, 51.7% de todos os usuários de telefones móveis, o que significa que telefones convencionais finalmente se tornarão a minoria no mundo das telecomunicações. Essas informações podem ser verificadas no Gráfico 2.1.

Gráfico 2.1 - Usuários de Smartphone e Penetração no Mundo.



Fonte: eMarketer (2014).

A pesquisa anteriormente citada também mostra que no fim de 2014 o Brasil ocupava a sexta (6ª) posição na lista de países com maior número de usuários de *smartphones* com 38.8 milhões, atrás apenas da China, Estados Unidos, Índia, Japão e Rússia. A estimativa é que em 2016 esse valor atinja 58.5 milhões de usuários no Brasil, como mostra a Tabela 2.1.

<sup>5</sup> <https://www.emarketer.com/>

Tabela 2.1 - Países Ranqueados pelo Número de Usuários de Smartphone.

<b>Top 25 Countries, Ranked by Smartphone Users, 2013-2018</b>						
<i>millions</i>						
	<b>2013</b>	<b>2014</b>	<b>2015</b>	<b>2016</b>	<b>2017</b>	<b>2018</b>
1. China*	436.1	519.7	574.2	624.7	672.1	704.1
2. US**	143.9	165.3	184.2	198.5	211.5	220.0
3. India	76.0	123.3	167.9	204.1	243.8	279.2
4. Japan	40.5	50.8	57.4	61.2	63.9	65.5
5. Russia	35.8	49.0	58.2	65.1	71.9	76.4
6. Brazil	27.1	38.8	48.6	58.5	66.6	71.9
7. Indonesia	27.4	38.3	52.2	69.4	86.6	103.0
8. Germany	29.6	36.4	44.5	50.8	56.1	59.2
9. UK**	33.2	36.4	39.4	42.4	44.9	46.4
10. South Korea	29.3	32.8	33.9	34.5	35.1	35.6
11. Mexico	22.9	28.7	34.2	39.4	44.7	49.9
12. France	21.0	26.7	32.9	37.8	41.5	43.7
13. Italy	19.5	24.1	28.6	32.2	33.7	37.0
14. Turkey	15.3	22.6	27.8	32.4	37.2	40.7
15. Spain	18.9	22.0	25.0	26.9	28.4	29.5
16. Philippines	14.8	20.0	24.8	29.7	34.8	39.4
17. Nigeria	15.9	19.5	23.1	26.8	30.5	34.0
18. Canada	15.2	17.8	20.0	21.7	23.0	23.9
19. Thailand	14.4	17.5	20.4	22.8	25.0	26.8
20. Vietnam	12.4	16.6	20.7	24.6	28.6	32.0
21. Egypt	12.6	15.5	18.2	21.0	23.6	25.8
22. Colombia	11.7	14.4	16.3	18.2	19.7	20.9
23. Australia	11.4	13.2	13.8	14.3	14.7	15.1
24. Poland	9.4	12.7	15.4	17.4	19.4	20.8
25. Argentina	8.8	10.8	12.6	14.1	15.6	17.0
<b>Worldwide***</b>	<b>1,311.2</b>	<b>1,639.0</b>	<b>1,914.6</b>	<b>2,155.0</b>	<b>2,380.2</b>	<b>2,561.8</b>

*Note: individuals of any age who own at least one smartphone and use the smartphone(s) at least once per month; \*excludes Hong Kong; \*\*forecast from Aug 2014; \*\*\*includes countries not listed*  
Source: eMarketer, Dec 2014

182905 www.eMarketer.com

Fonte: eMarketer (2014).

## 2.2. Plataformas

Atualmente existem três (3) principais plataformas de *smartphones* no mercado: Android, iOS<sup>6</sup> e Windows Phone<sup>7</sup>. Segundo uma pesquisa realizada pela Gartner<sup>8</sup> em agosto de 2015, outras plataformas possuíam apenas 0,7% do mercado mundial, sendo 0,3% pertencente a BlackBerry. A plataforma Android aparece na pesquisa como a plataforma mais popular, com 271 mil unidades vendidas no segundo trimestre de 2015 (82,2%). Em segundo lugar aparece o iOS, com 48 mil (14,6%). E em terceiro o Windows Phone, com 8 mil unidades vendidas (2,5%). A plataforma iOS passou de 12,2% em 2014 para 14,6% em 2015, enquanto a plataforma Android passou de 83,8% para 82,2%. A Tabela 2.2 apresenta essas informações.

<sup>6</sup> <http://www.apple.com/ios/>

<sup>7</sup> <https://www.microsoft.com/pt-br/windows/phones>

<sup>8</sup> <http://www.gartner.com/>

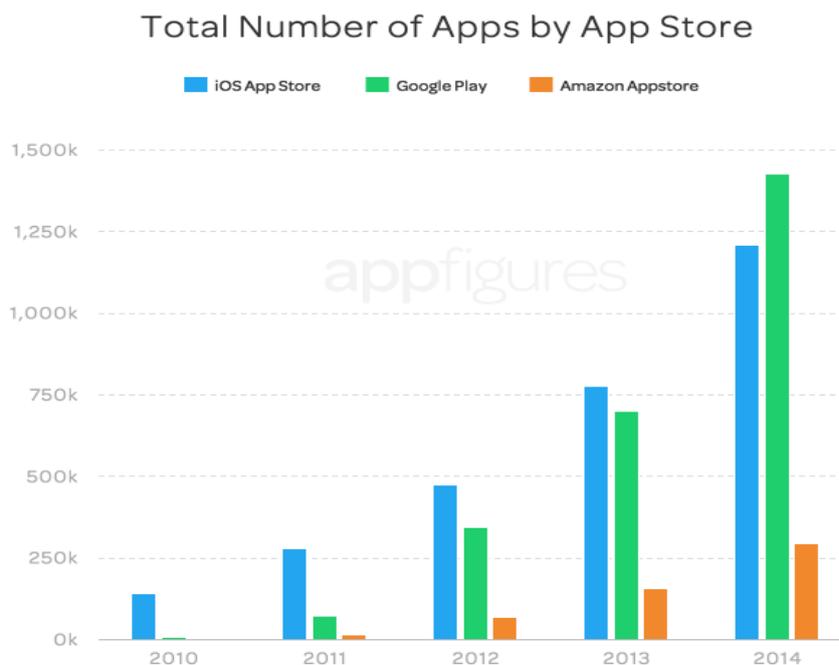
Tabela 2.2 - Venda de *smartphones* para usuários finais por SO.

Plataforma	2015		2014	
	Unidades	Porcentagem	Unidades	Porcentagem
Android	271.010	82,2	243.484	83,8
iOS	48.086	14,6	35.345	12,2
Windows Phone	8.198	2,5	8.095	2,8
BlackBerry	1.153	0,3	2.044	0,7
Outros	1.229	0,4	1.416,8	0,5
<b>Total</b>	<b>329.676,4</b>	<b>100</b>	<b>290.384,4</b>	<b>100</b>

Fonte: Gartner (2015).

Por ser o SO mais popular para *smartphones* e pela facilidade de publicar aplicações, o Android é a plataforma com a maior loja de aplicativos, possuindo 1.43 milhões de aplicativos disponíveis segundo uma pesquisa realizada pela appFigures<sup>9</sup> em 2014. O iOS vem em seguida com 1.21 milhões de aplicativos. A loja de aplicativos da Amazon aparece em terceiro lugar com 293 mil aplicativos publicados. No Gráfico 2.2 pode-se ver o crescimento do número de aplicativos para as plataformas Android e iOS ao longo dos anos.

Gráfico 2.2 - Número de aplicativos disponíveis por loja de 2010 a 2014.



Fonte: appFigures (2014).

<sup>9</sup> <https://appfigures.com/>

## 2.3. Limitações

Existem muitas diferenças entre os computadores convencionais e os dispositivos móveis: as estações de trabalho são estáticas, dependem de cabos, sua capacidade de memória e processamento não são problemas, sua tela é grande e a entrada de dados é realizada por meio de teclado, mouse e eventualmente outros dispositivos de entrada adequados para outras situações (BETIOL, 2004).

Apesar de hoje em dia os dispositivos móveis já estarem muito mais próximos em termos de processamento e memória, não sendo esses os principais problemas, ainda existem muitas limitações importantes a se considerar quando for desenvolver aplicações para esses dispositivos, como consumo de energia, dimensões da tela, entrada de dados e a própria mobilidade que pode gerar problemas de variação e indisponibilidade de rede.

As interfaces devem funcionar em muitas plataformas diferentes, desde um computador convencional a um *smartphone* sempre considerando o contexto e preservando a usabilidade. Atualmente é muito comum que as interfaces sejam trabalhadas separadamente para computadores de mesa e *smartphones* para garantir que se adequem ao uso e à forma de interação do usuário em cada um deles. Grande parte das páginas web, por exemplo, já possuem uma versão *mobile*.

### 2.3.1. Dimensões da Tela

Uma das características dos dispositivos móveis são as dimensões reduzidas de sua tela. Esta limitação no espaço da tela é um grande problema enfrentado, pois a tela não pode ser fisicamente maior sem tornar o dispositivo grande, o que faria com que ele perdesse uma de suas principais características, a portabilidade (BREWSTER; CRYER, 1999). Os desenvolvedores tentam enfrentar esta limitação construindo interfaces melhores utilizando várias técnicas de usabilidade, mas, mesmo assim, ainda é comum encontrar interfaces difíceis de serem usadas e com problemas na apresentação da informação.

A evolução da tecnologia usada nos dispositivos móveis atenua essas limitações cada vez mais com o passar do tempo, mas elas continuam presentes. Uma importante aquisição nesse sentido foi o surgimento e popularização do *touch screen* (tela sensível ao toque) que permite que o usuário interaja com a tela de uma forma totalmente diferente, trocando os botões e teclados físicos por opções virtuais que aparecem conforme a necessidade. Isso tornou

possível uma grande redução na quantidade de botões físicos permitindo aumentar o tamanho das telas, conseqüentemente facilitando a criação de interfaces e a visualização de dados e informações nesses aparelhos.

Apesar disso, a interface ainda é limitada pelo tamanho reduzido da tela e cabe aos desenvolvedores encontrar formas cada vez melhores, mais confortáveis, práticas e amigáveis para a interação com o usuário.

### **2.3.2. Entrada de Dados**

O processo de entrada de dados aparece como outra limitação para os dispositivos móveis. Para garantir a portabilidade do aparelho é necessário que as formas de entradas sejam pequenas e embutidas no aparelho, seja física ou logicamente.

Os meios de entrada de dados mais comuns em dispositivos móveis são teclados e telas sensíveis ao toque, sendo essa última uma tecnologia recente que foi uma evolução considerável para a usabilidade nesses aparelhos, como mencionado anteriormente.

Dispositivos de entrada de dados que costumam ser utilizados em outros tipos de computadores (*desktop*, *notebook* e *consoles*) como mouse, teclado convencionais e *joysticks* normalmente não são utilizados em dispositivos móveis, o que torna muitas vezes mais difícil a experiência do usuário com a interface. Isso acontece porque cada uma dessas ferramentas de entrada realizam uma tarefa específica, possuindo também suas limitações. Ou seja, cada uma é mais adequada para um tipo de interação, e é isso que torna muito mais fácil o uso de interfaces por meio de computadores convencionais (PRAYAGA, 2010).

Porém, os dispositivos móveis oferecem também mecanismos de entrada alternativos como sensibilidade ao movimento, giro da tela, câmera e voz. Tirar proveito desses meios de comunicação de forma a contornar a limitação da falta de meios de entrada a que os usuários estão acostumados é ao mesmo tempo uma solução e um desafio (CEREJO, 2012).

### **2.3.3. Energia**

Os dispositivos móveis necessitam de fontes de energia próprias para tornar possível sua mobilidade, mas o grande problema é que as baterias normalmente utilizadas não possuem uma boa duração devido à limitação do tamanho dos aparelhos (FIGUEIREDO; NAKAMURA, 2003).

As tecnologias utilizadas nos dispositivos móveis estão evoluindo cada vez mais rápido. A cada ano surgem novos aparelhos dotados de uma maior capacidade de processamento, quantidade de memória, capacidade de armazenamento, além do surgimento de novas tecnologias de redes móveis como o 4G, com melhores velocidades de conexão. Todos esses avanços fazem com que cada vez mais os dispositivos móveis consumam mais energia. O problema é agravado pois o avanço da tecnologia de baterias tem sido lento em acompanhar essas necessidades de consumo e ainda atender às limitações de tamanho dos dispositivos móveis (URRIZA et al., 2004).

Existem duas possíveis soluções para aumentar a duração das baterias sem considerar novas tecnologias: aumentar a bateria ou melhorar o gerenciamento de energia. Dentre os componentes de hardware presentes em um dispositivo móvel, o processador, o vídeo, o disco e a interface de rede são os principais consumidores de energia. Para cada um deles existem técnicas de software que têm por objetivo minimizar o consumo de energia (URRIZA et al., 2004).

## **2.4. Portabilidade x Mobilidade**

Dois conceitos muito importantes quando se fala de dispositivos móveis são portabilidade e mobilidade. Esses conceitos são muito frequentemente confundidos e considerados sinônimos, mas, apesar de existir uma ligação entre eles, possuem significados diferentes.

Portabilidade na computação significa possuir todas as ferramentas e funcionalidades necessárias ao usuário para a realização de determinadas tarefas de forma compacta para que possa ser movido com facilidade e conforto de um lugar para outro (GALLIANO, 2007). Em outras palavras, para um dispositivo ser portátil é preciso ser pequeno, leve, durável, operacional em diversas condições e possuir autonomia de energia (AMADO, 2002).

A portabilidade traz diversas limitações ao equipamento por causa da necessidade de manter um tamanho confortável para o transporte, entre elas memória, capacidade de processamento, capacidade de armazenamento, consumo de energia e interface com o usuário.

O termo mobilidade na computação vai além da portabilidade, significa acesso à informação de qualquer lugar e a qualquer momento. Logo, para possuir mobilidade é preciso ser portátil, mas a mobilidade vai além.

A mobilidade leva em consideração também as necessidades dos usuários que precisam se conectar à rede a partir de diversas localizações e em movimento para acessar informações. Esse cenário é consideravelmente mais complexo pois envolve considerar diversos aspectos como a mudança automática entre as redes conectadas pelo dispositivo, além de possíveis desconexões.

Em um dispositivo móvel seu endereço de rede muda conforme o movimento, sua localização passa a influir em parâmetros de configuração e respostas a consultas (AMADO, 2002).

Um ponto ainda mais crítico está relacionado a aplicações que precisam de conexão constante com a Internet para funcionar e que têm que estar disponíveis todo o tempo. Nesses casos, para se garantir a mobilidade completa da aplicação é necessário que nos momentos de desconexão a aplicação gerencie as ações do usuário e repasse as informações pela Internet quando recuperar a conexão.

No entanto, é importante observar que nem todas as aplicações precisam garantir tal nível de mobilidade. Existe toda uma área de estudo sobre computação móvel que é focada em aproveitar toda sua capacidade.

## **2.5. Banco de Dados em Dispositivos Móveis**

O desenvolvimento dos componentes utilizados nos dispositivos móveis e a evolução das tecnologias de redes móveis têm permitido o surgimento de diversas novas aplicações móveis para auxiliar em atividades que antes eram possíveis apenas por meio de computadores convencionais. Algumas dessas novas aplicações que vêm se tornando possível nesses dispositivos e têm um grande potencial são as aplicações relacionadas a sistemas de bancos de dados.

Atualmente, já existem aplicativos que envolvem consulta, visualização e gerenciamento de dados oriundos de sistemas de bancos de dados para várias plataformas. Esses podem ser divididas em três grupos: banco de dados embutidos, ferramentas de acesso a bancos de dados remotos e bancos de dados móveis.

Banco de dados embutido é um Sistema de Gerenciamento de Banco de Dados (SGBD) que é totalmente integrado com uma aplicação que requer acesso a dados armazenados. Ou seja, os bancos de dados embutidos implementam todo o SGBD e armazenam localmente os dados.

Por isso, normalmente são mais simples, implementando menos regras e restrições para manter-se compacto e rápido.

São exemplos de sistemas de bancos de dados embutidos SQLite<sup>10</sup>, HSQLDB<sup>11</sup>, SQL Server Compact<sup>12</sup>, Berkeley DB<sup>13</sup> e Firebird Embedded<sup>14</sup>.

Ferramentas de acesso a banco de dados remotos são aplicações que têm como objetivo permitir a conexão com um SGBD remoto em qualquer lugar e possivelmente em movimento, trazendo mais simplicidade e certo nível de mobilidade para as atividades de gerenciamento e consumo de dados do banco de dados.

Porém, como foi mencionado anteriormente, existem várias dificuldades quando se trata de mobilidade, como tornar possível que o usuário tenha acesso a todas as funcionalidades do sistema independente do estado de conexão atual do dispositivo, podendo estar em qualquer rede ou em nenhuma.

Permitir toda essa mobilidade não é o objetivo dessas ferramentas de acesso a bancos de dados remotos. As aplicações com esse objetivo são chamadas de bancos de dados móveis.

Banco de dados móvel é um ramo de estudo da computação móvel que tenta extrair todo o potencial da mobilidade no gerenciamento de banco de dados.

Para alcançar o objetivo dessas aplicações tornam-se necessárias mudanças na arquitetura, gerenciamento e nos mecanismos de garantia de consistência dos dados (GALLIANO, 2007).

Um entre vários possíveis cenários para o uso de bancos de dados móveis inclui servidores fixos de bancos de dados e um ou mais dispositivos móveis dotados de um SGBD capaz de gerenciar as requisições do usuário independentemente dos servidores estarem acessíveis. Para isso a aplicação precisa possuir parte ou todos os dados do banco de dados armazenados localmente, e, eventualmente, quando conectado sincronizar a informação com os servidores.

---

<sup>10</sup> <https://www.sqlite.org/>

<sup>11</sup> <http://hsqldb.org/>

<sup>12</sup> <https://msdn.microsoft.com/pt-br/data/ff687142.aspx>

<sup>13</sup> <https://www.oracle.com/database/berkeley-db/index.html>

<sup>14</sup> <http://www.firebirdsql.org/>

## 2.6. Considerações Finais

Atualmente os dispositivos móveis já estão bastante desenvolvidos em questão de hardware e software, possuindo alguns componentes de hardware comparáveis a computadores *desktop* e uma grande quantidade de aplicativos disponíveis para diversas áreas e funções.

Porém, devido à necessidade de portabilidade, esses dispositivos trazem consigo limitações que dificultam o desenvolvimento de aplicações em diversos aspectos. As principais limitações encontradas são: dimensões da tela, meios de entrada de dados e gerenciamento de energia.

As limitações de dimensões e dispositivos de entrada foram atenuadas consideravelmente pela evolução da tecnologia *touch screen*, porém ainda são as duas principais dificuldades no desenvolvimento de interfaces para esses dispositivos.

Existem três (3) tipos diferentes de aplicativos de gerenciamento de dados oriundos de bancos de dados, cada um deles diferindo na capacidade de mobilidade e localidade dos dados. São eles: banco de dados embutidos, ferramentas de acesso a bancos de dados remotos e bancos de dados móveis. A escolha de qual dos tipos de aplicativo usar depende da necessidade apresentada em cada situação.

Este trabalho será focado nas ferramentas de acesso a bancos de dados remotos, suas funcionalidades essenciais, interface e usabilidade, de modo a facilitar o uso dessas aplicações contornando as limitações apresentadas pelos dispositivos móveis.

O próximo capítulo apresentará um estudo sobre usabilidade, seus conceitos, sua importância e formas de avaliação.

### 3. Usabilidade

No contexto da computação, a usabilidade é um conceito chave na área de estudo conhecida como Interação Humano-Computador (IHC). O termo é usado para definir a facilidade e simplicidade com que o usuário pode utilizar um sistema para executar uma determinada tarefa.

O termo Interação Humano-Computador foi adotado em meados dos anos 80 para descrever o novo campo de estudo que estava surgindo. O termo foi escolhido para deixar claro que o foco de interesse é mais amplo que somente o *design* de interfaces “amigáveis ao usuário” e abrange todos os aspectos relacionados com a interação entre usuários e computadores (ROCHA; BARANAUSKAS, 2003).

Segundo NIELSEN (1993), o termo “amigável ao usuário” não era apropriado, pois um sistema não precisa ser amigável com o usuário, precisa resolver da maneira mais simples possível as necessidades do usuário. Além disso, uma interface amigável a um grupo de usuários pode ser entediante a outro grupo.

A IHC pode ser definida como uma área do conhecimento que se dedica ao *design*, à avaliação e à implementação de sistemas computacionais interativos para uso humano e ao estudo dos principais fenômenos que os cercam (HEWETT et al., 1992).

Esse campo de conhecimento possui influência direta de diversas áreas, como, por exemplo, ergonomia, psicologia, sociologia, antropologia, *design*, engenharias e ciência da computação. É esse referencial teórico que fornece importantes subsídios para compreender quais os processos que estão envolvidos na interação física e cognitiva do ser humano com todo o sistema computacional (BETIOL, 2004).

O conceito de usabilidade é visto na IHC como um dos critérios que definem a aceitabilidade de um sistema, sendo essencial em todo projeto de sistema computacional direcionado ao uso humano (NIELSEN, 1993; SHACKEL, 1991).

A usabilidade está intimamente relacionada à interface dos sistemas computacionais, pois é por meio da interface que se dá o processo de interação do usuário com o sistema. A interface tem o objetivo de fornecer ao usuário todos os elementos necessários para a comunicação com o software. Ou seja, quanto maior for a usabilidade da interface, mais fácil será a comunicação.

Os estudos relacionados à interação do usuário com o produto de software têm crescido consideravelmente nas últimas décadas. Um dos principais motivos desse crescimento foi a

percepção por parte dos desenvolvedores de sistemas de que a satisfação subjetiva do usuário era um fator determinante para o sucesso e aceitação de produtos de software. A partir desse momento o usuário passou a ser considerado parte fundamental no projeto de software e a indústria de desenvolvimento de sistemas vem investindo cada vez mais em avaliações de usabilidade para melhorar a aceitação de seus produtos (FERREIRA, 2007).

### 3.1. Definições de Usabilidade

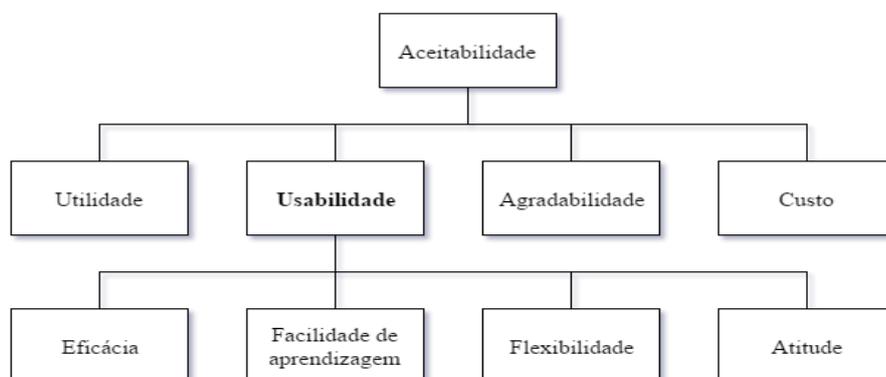
Existem várias definições para usabilidade no contexto de computação na literatura, porém todas definem o mesmo conceito, possuindo como diferença apenas algumas características e critérios básicos para avaliação de usabilidade.

Alguns dos autores sobre usabilidade na literatura são SHACKEL (1991), NIELSEN (1993) e JORDAN (1998). Serão descritas as definições de usabilidade desses autores.

#### 3.1.1. Shackel

Segundo SHACKEL (1991), quando um usuário ou cliente vai escolher um sistema eles não analisam apenas a usabilidade, e sim todo um conjunto de aspectos. Esse conjunto é composto de utilidade, usabilidade, agradabilidade e custo do produto. Esses aspectos juntos são considerados a aceitabilidade do sistema (Figura 3.1).

Figura 3.1 - Aspectos de aceitabilidade de um sistema segundo Shackel.



Fonte: SHACKEL (1991).

Utilidade se refere à capacidade do sistema de realizar as tarefas para as quais foi projetado. Já usabilidade trata da facilidade com que o usuário vai conseguir utilizar o sistema.

E, por sua vez, a agradabilidade diz respeito à satisfação subjetiva do usuário em relação ao sistema.

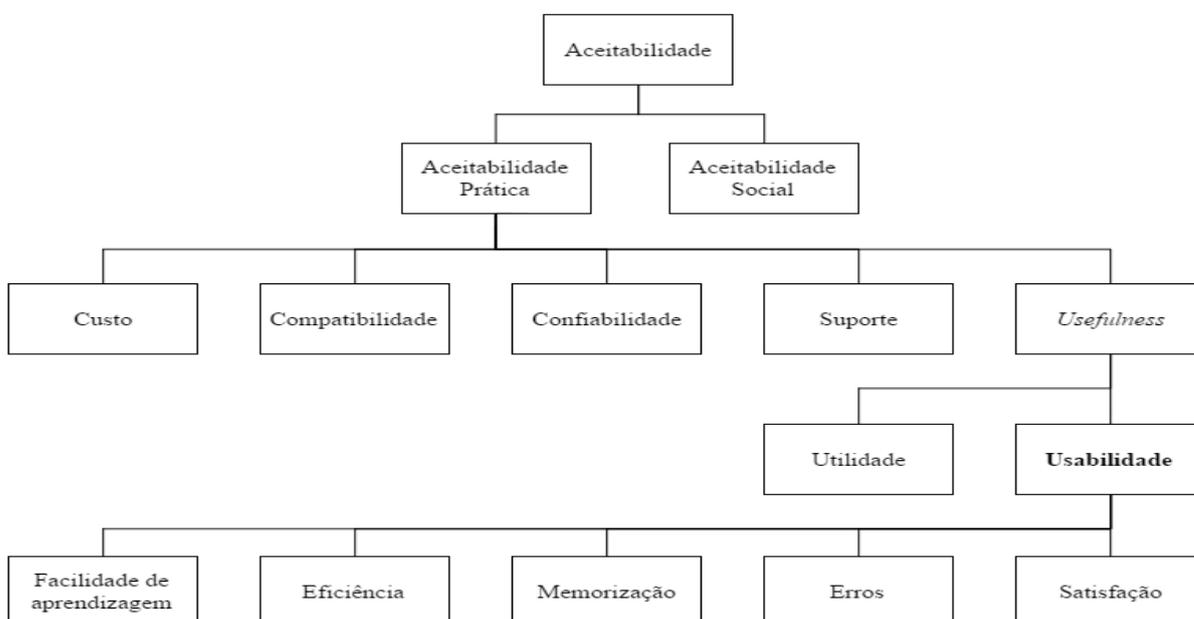
De acordo com SHACKEL (1991), usabilidade pode ser definida e medida de acordo com os seguintes critérios:

- Eficácia - Avalia se o sistema atinge determinados níveis de desempenho. Cada tarefa e sistema pode ter uma necessidade diferente. Pode ser avaliada em função do número de tarefas realizadas num determinado período de tempo ou em função da quantidade de erros;
- Facilidade de aprendizagem - É avaliada de acordo com o tempo necessário para um usuário aprender a usar o sistema, levando em consideração a frequência de uso;
- Flexibilidade - Avalia a adaptação do usuário a tarefas e ambientes diferentes dos que já tem experiência; e
- Atitude - Avalia o custo investido pelo usuário em termos de cansaço, frustração, desconforto e esforço pessoal para utilizar o sistema.

### 3.1.2. Nielsen

Semelhante a SHACKEL (1991), para NIELSEN (1993), usabilidade é um dos aspectos que definem um conceito mais abrangente, a aceitabilidade de sistema (Figura 3.2).

Figura 3.2 - Aspectos de aceitabilidade de um sistema segundo Nielsen.



Fonte: NIELSEN (1993).

Segundo NIELSEN (1993), a aceitabilidade de um sistema pode ser definida como a capacidade de um sistema computacional de atingir todos os requerimentos e necessidades dos clientes e de todos os usuários que irão interagir com o sistema. Além disso, a aceitabilidade de um sistema é composta diretamente dos conceitos de aceitabilidade social e aceitabilidade prática.

A aceitabilidade social diz respeito aos usuários acreditarem que o sistema pode efetivamente melhorar o desempenho de determinada atividade, em vez de dificultar ou inserir complicações desnecessárias. Ou seja, para um produto ser socialmente aceito ele precisa resolver um problema que existe ou facilitar em algum aspecto a execução de determinada tarefa que seja reconhecida pelo usuário.

Uma vez que se atinge a aceitabilidade social, o produto precisa tratar da aceitabilidade prática, que, por sua vez, envolve aspectos como custo, compatibilidade, confiabilidade, suporte e *Usefulness*.

*Usefulness* é definido pelo autor como sendo uma combinação de utilidade e usabilidade, e se refere à capacidade do sistema de poder ser usado para atingir um determinado objetivo.

Utilidade trata da funcionalidade do sistema, ou seja, se o sistema é capaz de fazer o que foi projetado para fazer.

Já a usabilidade diz respeito à qualidade com que as funcionalidades podem ser utilizadas. Ela aplica-se a todos os aspectos do sistema com os quais um usuário pode interagir, inclusive procedimentos como instalação e manutenção.

A usabilidade é definida em função de múltiplos componentes e é tradicionalmente associada com os seguintes cinco atributos de usabilidade:

- Facilidade de aprendizagem - O sistema deve ser de fácil aprendizado, de forma que o usuário possa rapidamente começar a utilizar. Segundo NIELSEN (1993), esse é o atributo mais importante da usabilidade já que a primeira experiência que o usuário tem com o sistema é aprender a utilizá-lo;
- Eficiência - O sistema deve ser eficiente na sua utilização, de forma que uma vez que o usuário aprenda a utilizá-lo, tenha um elevado nível de produtividade;
- Memorização - O uso do sistema deve ser lembrado com facilidade, de forma que depois de um tempo sem utilizá-lo o usuário não precise reaprender. Esse aspecto diz respeito a usuários casuais e a sistemas que são mais utilizados em períodos específicos;

- Erros - O sistema deve ter baixa taxa de erros, e quando ocorrer erros o sistema deve facilmente se recuperar deles. Não devem ocorrer erros catastróficos, por exemplo, erros que levem o sistema a um estado inconsistente, ou que causem perda de dados; e
- Satisfação - O sistema deve ser agradável, de forma que os usuários fiquem subjetivamente satisfeitos ao usá-lo.

### 3.1.3. Jordan

JORDAN (1998) se baseia na definição de usabilidade da International Organization for Standardization (ISO)<sup>15</sup>, para a qual usabilidade trata da eficácia, eficiência e satisfação que um grupo específico de usuários pode usar um sistema para atingir determinados objetivos em determinado ambiente.

Eficácia se refere à realização de determinada tarefa com sucesso. A eficácia na execução de uma tarefa depende da tarefa em si. Determinadas tarefas são consideradas realizadas com sucesso simplesmente pela não ocorrência de erros, já outras, como em sistemas críticos, existem por exemplo requisitos de tempo, quando a eficácia depende do tempo de execução da tarefa.

Já eficiência diz respeito à quantidade de esforço necessário para atingir determinado objetivo. Quanto menos esforço for necessário para realizar uma tarefa, melhor será a eficiência do sistema. A eficiência pode ser medida por meio da quantidade de tempo necessária ou o número de erros que o usuário comete para executar uma tarefa.

Satisfação se refere ao nível de conforto que o usuário sente ao utilizar um sistema para atingir seus objetivos. Esse aspecto é o mais subjetivo entre os três (3) mencionados, por isso pode ser mais difícil de medir. Entretanto, isso não quer dizer que ele é menos importante. Na verdade, em muitas situações esse pode ser o aspecto mais importante da usabilidade para um usuário considerar usar o sistema. Por exemplo, em sistemas de uso opcional e de lazer, nos quais o uso não é obrigatório, os usuários podem deixar de utilizar um sistema funcional se não achar seu uso confortável ou agradável.

JORDAN (1998) enfatiza que a usabilidade deve ser medida sempre considerando o grupo de usuários para o qual o produto é direcionado, pois um sistema com boa usabilidade

---

<sup>15</sup> <http://www.iso.org/>

para uma pessoa pode possuir má usabilidade para outras. Por isso, é muito importante conhecer o grupo de usuários específico do sistema e suas características.

Além disso, JORDAN (1998) põe em destaque a experiência do usuário. Ele explica que a usabilidade do produto pode variar de acordo com a experiência que o usuário tem com o sistema e quanto mais se usa o produto mais se tem experiência ao utilizá-lo. O desempenho de uma tarefa que parecia difícil a princípio, pode ficar muito fácil depois de algumas execuções.

Para refletir a importância da experiência do usuário no uso do sistema foi desenvolvido um modelo baseado em cinco componentes para definir usabilidade. Esse modelo foi desenvolvido inicialmente por JORDAN et al. (1991) e estendido por JORDAN (1994). Os cinco componentes são intuitividade, facilidade de aprendizagem, desempenho de usuários experientes (JORDAN et al., 1991), potencial do sistema e reusabilidade (JORDAN, 1994). Segue abaixo a definição de cada um deles:

- Intuitividade - Se refere ao custo na execução de uma tarefa do sistema por um usuário pela primeira vez. Quanto menor o esforço para realizar a tarefa, maior a intuitividade. O custo pode ser medido em função do tempo de execução ou número de erros. Esse aspecto é especialmente importante para sistemas de uso eventual, como por exemplo extintores de incêndio;
- Facilidade de aprendizagem - Trata do esforço necessário para um usuário atingir certo nível de competência na execução das tarefas que já foram desempenhadas por ele. Quanto menor o tempo de experiência e treinamento necessário para o usuário se tornar experiente na execução de determinada tarefa, maior será a facilidade de aprendizagem;
- Desempenho de usuários experientes - Refere-se ao nível de desempenho com que um usuário com experiência em determinada tarefa consegue realizá-la. É o componente mais importante quando os requisitos do sistema priorizam produtividade a longo prazo à velocidade de aprendizado;
- Potencial do sistema - Representa o nível máximo de desempenho que pode ser atingido no uso do sistema para realizar determinada função. É o limite superior do componente desempenho de usuários experientes; e
- Reusabilidade - Refere-se à possível redução do desempenho na execução de determinada tarefa após o usuário passar certo tempo sem executar a mesma.

### 3.2. Avaliação de Usabilidade

Avaliação de usabilidade consiste na aplicação de processos que têm como objetivo por meio da análise das interfaces avaliar a funcionalidade do sistema, avaliar o efeito da interface no usuário e identificar problemas de *design* (ROCHA; BARANAUSKAS, 2003). A avaliação desses aspectos é essencial em qualquer sistema para garantir que o produto está de acordo com a necessidade e requisitos do usuário, e que o sistema atenderá às suas expectativas.

A avaliação da funcionalidade do sistema verifica se o produto está de acordo com os requisitos funcionais do usuário, ou seja, se a interface permite a execução de todas as funcionalidades necessárias e se essas estão funcionando corretamente.

O *design* das interfaces deve também permitir ao usuário executar as tarefas do modo mais fácil e eficiente possível. Isso significa que apenas ser possível a realização das tarefas a partir da interface não é o suficiente, é necessário ainda avaliar quão bem o usuário vai conseguir utilizar a interface para a realização das tarefas disponíveis.

Além de analisar a funcionalidade e a usabilidade do sistema, os métodos de avaliação de usabilidade têm como objetivo identificar os aspectos da interface que possam resultar em confusão e levar o usuário a possíveis erros ou resultados inesperados.

A avaliação de usabilidade é essencial no desenvolvimento de qualquer sistema e deve ser realizada várias vezes durante o processo de criação do sistema desde o protótipo até as versões finais, pois quanto mais cedo problemas de *design* forem encontrados mais fácil e barato será para resolvê-los.

Existem diversos métodos de avaliação de usabilidade consolidados que podem ser utilizados no decorrer de um projeto. Essas técnicas podem ser combinadas e adaptadas para atender à necessidade de cada situação, como recursos disponíveis, tempo e fase do projeto (ROCHA; BARANAUSKAS, 2003).

Os métodos de avaliação podem ser agrupados em duas classes de acordo com o envolvimento ou não de usuários reais, respectivamente, métodos de avaliação empíricos e métodos de avaliação analíticos.

Métodos de avaliação empíricos englobam os métodos nos quais o usuário está diretamente ligado. Esses métodos são os mais tradicionais. No entanto, usuários reais são difíceis e caros para serem recrutados em todas as fases do desenvolvimento do sistema. Por isso, normalmente, esses métodos são utilizados mais nas fases finais do desenvolvimento,

enquanto os métodos de avaliação analíticos podem facilmente ser utilizados em todas as fases do desenvolvimento (ROCHA; BARANAUSKAS, 2003).

Muitos estudos sugerem que combinando métodos dos dois tipos se obtém um melhor resultado do que utilizando apenas um deles, pois é comum que métodos empíricos encontrem alguns problemas, enquanto métodos analíticos detectariam outros.

### **3.2.1. Métodos de Avaliação Analíticos**

Os métodos de avaliação analíticos ou por inspeção são métodos que permitem ao avaliador analisar e identificar problemas de usabilidade sem o envolvimento direto do usuário real. Por isso, esses métodos são mais flexíveis e baratos, podendo ser usados em praticamente todas as fases do desenvolvimento do sistema, como especificações de interfaces, protótipos ou em sistemas em etapas finais. Esses métodos são baseados em *checklists* e itens de verificação (ROCHA; BARANAUSKAS, 2003).

Métodos analíticos têm como objetivos identificar problemas de usabilidade em interfaces de usuário, priorizar os problemas encontrados de acordo com a gravidade do problema e o custo da correção, selecionar o maior número de problemas possíveis com base em suas prioridades e reprojeter a interface de forma a corrigir os problemas selecionados.

Problema de usabilidade é um conceito abstrato, mas pode ser definido como qualquer aspecto da interface no qual uma mudança pode melhorar alguma medida de usabilidade (ROCHA; BARANAUSKAS, 2003).

Os métodos analíticos não requerem muito esforço de quem vai utilizá-los e são flexíveis, podendo ser utilizados com qualquer metodologia de desenvolvimento de software e em qualquer fase do desenvolvimento. Os resultados são rápidos e fornecem concretas evidências de quais aspectos da interface devem ser aperfeiçoados.

Avaliação heurística e percurso cognitivo são os métodos analíticos mais utilizados e não exigem uma grande experiência ou longo treinamento para que possam ser utilizados (FILARDI, 2007; ROCHA; BARANAUSKAS, 2003).

### 3.2.1.1. Avaliação Heurística

O método de avaliação heurística foi desenvolvido por NIELSEN e MOLICH (1990) com base em suas experiências na área de engenharia de usabilidade. Esse método é um dos métodos analíticos, e como tal, não envolve diretamente o usuário final. A avaliação heurística tem como objetivo identificar problemas de usabilidade com base em um conjunto de heurísticas e retornar uma lista dos problemas encontrados. Heurísticas são princípios de usabilidade definidos por especialistas e profissionais experientes da área de IHC.

Esse método de avaliação é considerado um método fácil de ser aprendido, rápido de ser aplicado e possui custo consideravelmente menor que a maioria dos outros métodos. Normalmente é realizado por um pequeno grupo de avaliadores, geralmente de três (3) a cinco (5) especialistas. Os avaliadores idealmente devem possuir conhecimento no domínio do sistema, caso o sistema pertença a um domínio específico. Se isso não for possível, é necessário auxiliar o avaliador para que ele seja capaz de usar o sistema. Esse auxílio pode ser por meio de cenários de uso ou da presença de um especialista no domínio durante a avaliação (ROCHA; BARANAUSKAS, 2003).

A avaliação heurística inicialmente é realizada individualmente por cada avaliador, percorrendo a interface várias vezes analisando os componentes à procura de problemas de usabilidade de acordo com o conjunto de heurísticas utilizado. Os problemas encontrados devem ser listados sendo associados com as heurísticas que foram desrespeitadas.

Depois da etapa inicial, as listas de problemas de usabilidade encontrados por cada avaliador devem ser consolidadas em uma lista única. Nessa fase os avaliadores têm acesso aos relatórios dos outros avaliadores e podem debater sobre os problemas encontrados.

A etapa final consiste na seleção dos problemas que devem ser corrigidos. Essa seleção deve levar em consideração a gravidade do problema, o prazo do projeto e os custos associados à correção.

O conjunto de heurísticas mais mencionado na literatura e provavelmente o mais conhecido e utilizado é o conjunto de heurísticas de NIELSEN (1994). Essas heurísticas foram definidas por NIELSEN e MOLICH (1990) e depois reformuladas por NIELSEN (1994). A seguir são apresentadas as heurísticas de NIELSEN (1994):

- Visibilidade do *status* do sistema - O sistema deve manter os usuários informados sobre o *status* de sistema, fornecendo o *feedback* apropriado de forma rápida;

- Compatibilidade do sistema com o mundo real - O sistema deve se comunicar na linguagem do usuário, usando conceitos e termos familiares ao usuário real do sistema. As informações devem ser apresentadas em ordem natural e lógica;
- Controle do usuário e liberdade - Usuários podem cometer erros escolhendo funções que não pretendiam. Deve sempre existir uma “saída de emergência” clara para sair do estado indesejado de forma fácil sem a necessidade de um extenso diálogo;
- Consistência e padrões - O sistema deve manter um padrão para que o usuário não precise adivinhar que diferentes palavras, situações ou ações significam a mesma coisa.
- Prevenção de erros - Melhor do que apresentar boas mensagens de erro é prevenir a ocorrência desses erros;
- Reconhecimento ao invés de lembrança - A interface deve manter os objetos, ações e opções sempre visíveis. O usuário não deve precisar lembrar de informações de uma tela para outra no decorrer da execução de uma tarefa. Instruções de uso do sistema devem sempre estar presentes e ser facilmente acessadas;
- Flexibilidade e eficiência de uso - Ofereça aceleradores e caminhos alternativos para aumentar a eficiência no uso do sistema por usuários experientes. O sistema deve atender bem tanto aos usuários novatos quanto aos experientes;
- Estética e *design* minimalista - A interface não deve apresentar informações irrelevantes ou raramente necessárias. Informações extras competem com as informações relevantes e diminuem sua visibilidade;
- Ajudar o usuário a reconhecer, diagnosticar e corrigir erros - Mensagens de erro devem utilizar de linguagem clara, sem o uso de códigos, e indicar precisamente o problema e possíveis soluções; e
- *Help* e documentação - Embora seja melhor que o sistema seja capaz de ser usado sem documentação, pode ser necessário prover *help* e documentação. Esta deve ser de fácil uso, focada nas tarefas do usuário e não ser muito grande.

### **3.2.1.2. Percurso Cognitivo**

Percurso cognitivo é um método de avaliação de usabilidade analítico proposto por LEWIS et al. (1990) e POLSON et al. (1992) que tem como principal objetivo avaliar a facilidade de aprendizagem do sistema por meio da exploração. O foco na facilidade de aprendizagem foi motivado por estudos que mostraram que muitos usuários preferem aprender

a usar software por exploração, adquirindo conhecimento sobre novas funções apenas quando for necessário, ao invés de investir tempo em treinamento ou leitura de materiais de apoio (WHARTON et al., 1994).

Por causa do foco na facilidade de aprendizado, o percurso cognitivo não deve ser utilizado como único método para avaliar a usabilidade de um sistema, pois pode levar a uma interface com problemas em outros aspectos de usabilidade. Por exemplo, o sistema pode ser muito focado no aprendizado de usuários iniciantes e deixar de lado aspectos de eficiência que poderiam ser utilizados por usuários mais experientes.

O percurso cognitivo analisa os passos necessários para a realização de determinada tarefa com o objetivo de identificar problemas de usabilidade que dificultem a aprendizagem por exploração. Exemplos de aspectos de *design* que podem atrapalhar o aprendizado são problemas relacionados à linguagem utilizada pelo sistema, à visualização de botões, menus e opções e a mensagens pouco esclarecedoras.

O percurso cognitivo assim como a avaliação heurística não envolve usuários. Esse método pode ser aplicado individualmente pelo próprio projetista ou em grupo, que pode ser formado por membros da equipe do projeto, pessoal de *marketing* e especialistas em interfaces.

O método de avaliação percurso cognitivo possui duas fases, a fase preparatória e a fase de análise. Na fase preparatória são definidas informações cruciais para a execução do método: as tarefas que serão analisadas, sequência de ações para cada tarefa, descrição dos usuários reais e a interface que será analisada. Na fase de análise os avaliadores percorrerem a sequência de ações necessárias para a execução de cada tarefa definida levando em consideração o usuário final e devem se fazer uma série de perguntas, buscando encontrar problemas de usabilidade que podem ocorrer na interação entre o usuário real e a interface. Encontrando ou não problemas, se deve descrever em detalhes como seria essa interação, para que as histórias de sucesso fiquem registradas e possam ser validadas e as de fracasso utilizadas para reprojeter a interface. As perguntas para se guiar na fase de análise podem mudar de acordo com a necessidade e situação, mas o conjunto de perguntas mais utilizados é o seguinte:

- Os usuários saberão qual a ação correta para atingir o resultado desejado?
  - Será que o usuário sabe qual a próxima ação a ser executada para realizar a tarefa?
- Os usuários perceberão que a ação correta está disponível?
  - Refere-se à visibilidade das informações. Ou seja, se a ação a ser tomada estará facilmente visível na interface.
- Os usuários irão associar a ação correta com o resultado desejado?

- Refere-se à interpretação dos objetos da interface. Será que vai estar evidente para o usuário o que determinado objeto (e.g. um botão) significa na interface e qual o resultado da interação com esse objeto?
- Se a ação correta for executada os usuários perceberão que foi feito avanço em relação à tarefa desejada?
  - Refere-se ao *feedback* apropriado depois da execução da ação correta.

Depois da realização da fase de análise as informações devem ser consolidadas e as histórias de sucesso validadas, caso o processo seja realizado por um grupo. Com base no relatório de histórias de uso detalhadas pode ser realizado um *redesign* no sistema para corrigir os problemas encontrados.

### 3.2.2. Métodos de Avaliação Empíricos

Os métodos de avaliação empíricos ou testes de usabilidade são métodos que se caracterizam pelo envolvimento direto dos usuários. Esses métodos permitem ao avaliador analisar a interação de usuários reais ou representativos com o sistema durante a execução de tarefas cuidadosamente preparadas, tornando possível identificar problemas reais de usabilidade e mensurar a usabilidade do sistema. É comum também utilizar questionários e entrevistas para obter *feedback* e sugestões dos usuários após a realização dos testes de usabilidade.

A utilização dos métodos empíricos requer a existência de uma implementação real do sistema. Essa implementação pode ser um protótipo interativo sem funcionalidades implementadas, um protótipo com um conjunto básico de tarefas implementadas ou uma implementação completa. Porém, normalmente, por causa do maior custo dos métodos empíricos, os testes de usabilidade são realizados em fases do desenvolvimento nas quais já existe uma versão do sistema implementada com um subconjunto de tarefas disponíveis, para poder coletar a maior quantidade de informações possível. Pelo mesmo motivo, é comum a aplicação de métodos de avaliação analíticos antes de métodos empíricos, para que parte dos problemas de usabilidade sejam detectados e corrigidos, impedindo que problemas mais graves atrapalhem a identificação de outros possíveis aspectos mais difíceis de ser observados com métodos analíticos.

Métodos empíricos coletam dois tipos de dados: quantitativos e qualitativos. Os dados quantitativos são os dados coletados por meio de medições de desempenho dos usuários durante a execução de tarefas do sistema. Essas medições podem ser realizadas com diversas métricas diferentes como, por exemplo, número de erros cometidos, tempo gasto para realização de determinada tarefa e o tempo gasto pelos usuários recorrendo a documentações. Por sua vez, os dados qualitativos são coletados por meio das anotações e observações realizadas durante a execução do teste. São esses dados que permitem identificar os aspectos da interface que levaram os usuários a cometer erros, o que atrapalhou seu desempenho ou gerou dúvidas. Ou seja, é a partir dos dados qualitativos que os avaliadores interpretam os resultados para identificar problemas e recomendar soluções.

BARBOSA e SILVA (2010) recomendam a execução de uma série de atividades para obter resultados mais eficazes na realização de teste de usabilidade (Quadro 3.1).

Quadro 3.1 - Aplicação do teste de usabilidade.

<b>Atividade</b>	<b>Tarefa</b>
Preparação	<ul style="list-style-type: none"> <li>• Definir tarefas que serão executadas no teste;</li> <li>• Definir o perfil dos participantes e recrutá-los;</li> <li>• Preparar materiais utilizados no teste;</li> <li>• Executar um teste piloto.</li> </ul>
Coleta de dados	<ul style="list-style-type: none"> <li>• Observar e registrar a performance e a opinião dos participantes durante os testes.</li> </ul>
Interpretação dos resultados	<ul style="list-style-type: none"> <li>• Reunir, contabilizar e sumarizar os dados coletados dos participantes.</li> </ul>
Relato dos resultados	<ul style="list-style-type: none"> <li>• Relatar a performance e a opinião dos participantes.</li> </ul>

Fonte: BARBOSA e SILVA (2010).

A atividade de preparação possui quatro (4) subatividades: definição das tarefas que farão parte do teste, definição do perfil e recrutamento dos participantes, preparação do material utilizado tanto pelos avaliadores quanto pelos usuários e execução de um teste piloto.

As tarefas realizadas durante um teste devem ser as principais funções do sistema e tarefas típicas dos usuários. Além disso, devem cobrir as partes mais significativas da interface. A descrição de cada tarefa deve ser curta, clara e representar situações reais de uso.

Depois da seleção e definição das tarefas devem ser recrutados usuários que irão participar do teste. É importante que os participantes sejam usuários reais ou usuários que representem bem os usuários típicos do sistema. Os participantes devem corresponder ao perfil dos usuários reais, possuindo idade, nível educacional, experiência com computadores e conhecimento do domínio das tarefas similar ao público alvo.

Testes de usabilidade podem ser realizados em ambiente controlado ou em ambiente real de trabalho. Em caso de testes em ambiente controlado, os equipamentos que serão utilizados pelos participantes e pelos avaliadores para coletar os dados devem ser preparados e previamente testados. Em testes realizados em ambiente real de uso, os usuários já possuirão os equipamentos necessários, sendo necessário apenas preparar e testar o equipamento utilizado para registrar e coletar dados.

Depois de preparar o ambiente de teste, deve ser realizado um teste piloto que pode ser realizado com membros da própria equipe de avaliadores para verificar o funcionamento do sistema e dos equipamentos e garantir que o teste ocorrerá de acordo com o planejado.

Após a etapa de preparação, a próxima atividade a ser executada é a de coleta de dados. Esta é composta da observação da interação dos usuários com a interface e possíveis questionários e entrevistas. A coleta de dados pode ser realizada por meio de anotações realizadas pelos avaliadores sobre o uso do sistema pelos usuários e por meio de gravação da realização dos testes. A anotação é mais lenta e pode deixar passar alguns detalhes, enquanto a gravação em vídeo pode ser analisada posteriormente com maior calma. O ideal é anotar apenas observações importantes durante os testes, e deixar para analisar os detalhes posteriormente com o vídeo produzido.

A próxima atividade é a interpretação e consolidação dos dados coletados. Nessa fase os dados são analisados para gerar resultados quantitativos e qualitativos. Dependendo da fase do projeto e do objetivo do teste de usabilidade, o foco pode ser obter dados quantitativos ou qualitativos. Os dados quantitativos possuem maior foco em fases finais do desenvolvimento para comparar a usabilidade com concorrentes e verificar se o sistema atingiu as metas de usabilidade. O foco nos dados qualitativos ocorre quando o objetivo principal do teste é identificar problemas, a origem desses problemas e elaborar recomendações de solução.

A última atividade do teste de usabilidade é a construção de um relatório dos resultados. Segundo BARBOSA e SILVA (2010), esse relatório deve conter:

- Os objetivos da avaliação;
- Breve descrição dos métodos de teste de usabilidade;
- O perfil dos avaliadores e dos participantes;
- As tarefas executadas pelos participantes; e
- Lista dos problemas encontrados, indicando para cada problema:
  - Onde ocorreu o erro;
  - Descrição e justificativa;
  - Os fatores de usabilidade prejudicados; e
  - Sugestões de solução.

### **3.3. Considerações Finais**

A usabilidade é um conceito essencial que deve ser considerado em qualquer sistema ou produto que exija interação com o usuário. Em muitos casos, a usabilidade pode ser a diferença entre o sucesso e o fracasso de um sistema, pois é um critério chave na aceitabilidade do produto. Principalmente no cenário atual, no qual existe concorrência para praticamente qualquer tipo de sistema em qualquer plataforma.

A usabilidade está diretamente relacionada ao usuário, por isso é essencial ter em mente o público alvo e suas características quando realizar a avaliação de usabilidade.

Existem diversos métodos para se avaliar a usabilidade de um sistema, cada um com suas vantagens e desvantagens, cabendo ao grupo de desenvolvedores avaliar qual dos métodos se adequa melhor aos seus objetivos e recursos.

Os métodos de usabilidade podem ser divididos em dois grupos de acordo com o envolvimento direto ou não de usuários, respectivamente, métodos de avaliação empíricos e métodos de avaliação analíticos.

Este capítulo teve como objetivo apresentar uma visão geral da área de estudo usabilidade e apresentar os principais métodos de avaliação de usabilidade. Com base nesse referencial teórico, no próximo capítulo serão selecionados os métodos utilizados na avaliação dos aplicativos concorrentes existentes.

## 4. Análise de Aplicações Existentes

Esse capítulo apresenta aplicação de técnicas e conhecimentos adquiridos nos capítulos anteriores para analisar ferramentas de acesso a banco de dados remotos, especificamente aplicações disponíveis para a plataforma Android. Tais ferramentas foram escolhidas para avaliação por causa do maior uso de bancos de dados em servidores do que bancos de dados embutidos e por causa do maior número de aplicações desse tipo disponíveis no mercado quando comparado a aplicações de bancos de dados móveis, que envolvem um maior nível de complexidade. A plataforma Android foi selecionada por ser a plataforma mais popular e ter a maior quantidade de aplicativos disponíveis na loja.

### 4.1. Forma de Análise e Avaliação

As avaliações dos aplicativos serão realizadas de três (3) formas:

- *Checklist* de funções consideradas essenciais e importantes para a visualização, manipulação e gerenciamento de dados oriundos de bancos de dados;
- Aplicação do método de avaliação heurística para analisar a interface e identificar problemas de usabilidade; e
- Aplicação do método percurso cognitivo para analisar a interface e identificar problemas de usabilidade, principalmente problemas relacionados com o aspecto de facilidade de aprendizagem.

A decisão dos métodos de avaliação de usabilidade utilizados foi baseada em FILARDI (2007), ROCHA e BARANAUSKAS (2003) que afirmam que métodos analíticos apresentam bons resultados na identificação de problemas de usabilidade com custos de tempo e financeiros consideravelmente menores que métodos empíricos, por não envolver usuários e não necessitar de um grande grupo de avaliadores experientes. Os principais métodos analíticos utilizados e mencionados na literatura são os métodos de avaliação heurística e percurso cognitivo. Esses dois métodos não exigem dos avaliadores muita experiência e nem de um treinamento ou estudo longo para que possa ser aplicado com bons resultados.

A seguir será definido em mais detalhes como cada um dos métodos de avaliação será utilizado.

### **4.1.1. Checklist de Funções**

Esse método de avaliação tem como objetivo avaliar a funcionalidade das aplicações, verificando se possuem as principais funções desempenhadas por administradores e usuários na visualização, manipulação e gerência de dados provenientes de sistemas de bancos de dados.

Serão consideradas nesta avaliação as seguintes funções básicas:

1. Realização de consultas;
2. Realização de comandos;
3. Visualização dos dados em formato tabular;
4. Visualização de objetos do banco:
  - 4.1. Tabela;
  - 4.2. View;
  - 4.3. Procedure;
  - 4.4. Function;
  - 4.5. Trigger.
5. Alteração estrutural:
  - 5.1. Criação de tabelas;
  - 5.2. Remoção de tabelas;
  - 5.3. Adição de colunas;
  - 5.4. Remoção de colunas.
6. Alteração de dados:
  - 6.1. Inserção de dados;
  - 6.2. Atualização de dados;
  - 6.3. Remoção de dados.

### **4.1.2. Avaliação Heurística**

A avaliação heurística será aplicada a cada uma das aplicações selecionadas com o objetivo de identificar problemas na interface que prejudiquem a usabilidade em algum aspecto. Para essa avaliação serão usadas as heurísticas de NIELSEN (1994). As interfaces serão exploradas duas (2) vezes procurando problemas de usabilidade. Cada problema encontrado será descrito em detalhes e irá deixar claro quais das heurísticas foram violadas.

### 4.1.3. Percurso Cognitivo

O método percurso cognitivo será utilizado para identificar problemas de usabilidade com foco na facilidade de aprendizagem das aplicações. Esse método envolve a construção de cenários que devem ser baseados nas tarefas mais importantes do sistema. Os avaliadores devem tentar realizar passo a passo os cenários, procurando identificar aspectos da interface que confundam, dificultem ou façam usuários iniciantes na aplicação cometerem erros. Para cada passo na execução de um cenário, o avaliador deve explicar em detalhes porque o usuário realizaria essa ação, com base no perfil do usuário.

Serão considerados na aplicação desse método usuários com conhecimento sobre as atividades mais comuns de banco de dados, com alguma familiaridade com ferramentas de visualização, manipulação e gerenciamento de dados em ambientes *desktop* e familiarizados com aplicativos para dispositivos móveis.

Serão utilizados três (3) cenários: um de realização de consultas, um de inserção de dados e outro de criação de tabelas. Essas tarefas foram escolhidas por serem algumas das principais funções de uma aplicação desse tipo e representar bem as demais. Os cenários serão detalhados a seguir, porém a sequência correta de ações será descrita em alto nível, pois cada aplicação apresentará uma sequência de ações diferente para a execução das mesmas tarefas.

1. Realização de consultas:

- Cenário: Realização de consulta que retorne todos os dados de uma tabela.
- Sequência de ações descritas em alto nível:
  - Entrar com as informações de conexão;
  - Selecionar o banco;
  - Ir para a tela de escrita de consulta;
  - Escrever consulta;
  - Executar consulta.

2. Inserção de dados:

- Cenário: Inserir um registro via interface em uma tabela.
- Sequência de ações descritas em alto nível:
  - Selecionar o banco;
  - Selecionar a tabela;
  - Ir para a tela de inserção de dados;
  - Preencher os campos do registro que será inserido;

- Criar o registro.
3. Criação de tabelas:
- Cenário: Criar uma tabela em um banco.
  - Sequência de ações descritas em alto nível:
    - Selecionar o banco;
    - Ir para a tela com as tabelas existentes;
    - Ir para a tela de criação de tabelas;
    - Adicionar colunas;
    - Criar tabela.

## 4.2. Aplicações Selecionadas

As aplicações foram selecionadas de acordo com a descrição, quantidade de *downloads*, avaliações e funções disponíveis. Apesar de ser dada preferência às aplicações gratuitas por possuírem maior alcance, um dos aplicativos escolhidos não é gratuito. Esse foi escolhido por ser o mais completo encontrado, possuindo diversas funções mais avançadas focadas em gerência de banco de dados.

As aplicações escolhidas serão descritas e avaliadas individualmente conforme especificado anteriormente.

### 4.2.1. SQL Client

O SQL Client<sup>16</sup> é uma ferramenta de acesso a bancos de dados remotos que permite conexão a diferentes SGBD a partir de um *smartphone* ou *tablet* com SO Android. O SQL Client dá suporte a conexões aos SGBD Microsoft SQL Server, MySQL<sup>17</sup> e PostgreSQL<sup>18</sup>. Esse aplicativo tem como funções executar e salvar consultas, exportar resultado de consultas em arquivos CSV<sup>19</sup> e listar tabelas e *views* existentes no banco.

---

<sup>16</sup> <https://play.google.com/store/apps/details?id=de.mxapplications.sqlclient>

<sup>17</sup> <https://www.mysql.com/>

<sup>18</sup> <https://www.postgresql.org/>

<sup>19</sup> <http://www.creativyst.com/Doc/Articles/CSV/CSV01.htm>

O SQL Client é um aplicativo gratuito desenvolvido por Sebastian Dombrowski<sup>20</sup> com mais de 1.000 *downloads* e com nota de 4,6 na Google Play<sup>21</sup> numa escala de 0 a 5, sendo essa nota atribuída pela avaliação de 16 usuários.

#### 4.2.1.1. *Checklist de Funções*

Dentre as funções considerada básicas para uma ferramenta de acesso a banco de dados remoto, esse aplicativo permite apenas a realização de consultas (1), visualização de dados em formato tabular (3) e visualização de alguns objetos do banco, tabelas (4.1) e *views* (4.2). Não é possível visualizar os demais objetos de banco de dados como *procedures* (4.3), *functions* (4.5) e *triggers* (4.6).

Não é possível realizar alterações estruturais em tabelas (5) ou realizar alterações de dados (6) via interface. Também não é possível realizar comandos (2) a partir da tela de consultas. Comandos de alterações de dados e estrutura podem ser digitados, aparecem em *highlight* e podem ser executados, mas nenhuma modificação é realizada, apesar da mensagem apresentada deixar essa impressão.

#### 4.2.1.2. *Avaliação Heurística*

A aplicação do método de avaliação heurística nesse aplicativo resultou na seguinte lista de problemas:

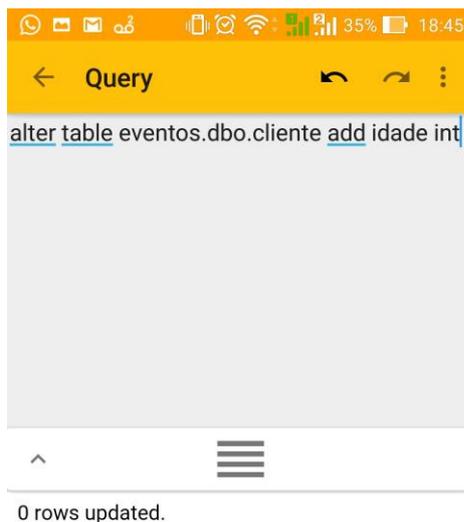
- *Feedback* inapropriado - Alteração estrutural
  - Descrição: Na tela de execução de consultas, quando são executados comandos de alteração estrutural, como adição de colunas, a mensagem resultante não deixa claro se o procedimento foi realizado com sucesso ou não. Esse tipo de comando não funciona nesse aplicativo, porém pela mensagem o usuário pode achar que o comando foi executado. O *feedback* para esse tipo de comando pode ser visto na Figura 4.1.
  - Heurística violada: Visibilidade do *status* do sistema.

---

<sup>20</sup> <https://play.google.com/store/apps/developer?id=Sebastian+Dombrowski>

<sup>21</sup> <https://play.google.com/store>

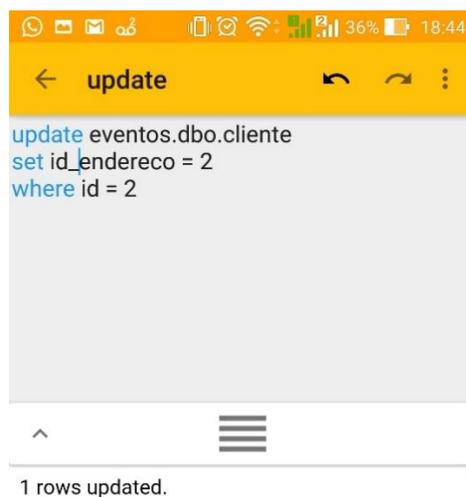
Figura 4.1 - *Feedback* inapropriado na execução de comandos de alteração estrutural.



Fonte: SQL Client (2016).

- *Feedback* inapropriado - Alteração de dados
  - Descrição: Na tela de execução de consultas, quando são executados comandos de alteração de dados, como inserção ou atualização de dados, a mensagem resultante informa ao usuário que o procedimento foi realizado com sucesso, quando na verdade o comando não alterou os dados. Esse tipo de comando não funciona nesse aplicativo, porém a mensagem informa que o comando foi executado. O *feedback* para esse tipo de comando pode ser visto na Figura 4.2.
  - Heurística violada: Visibilidade do *status* do sistema.

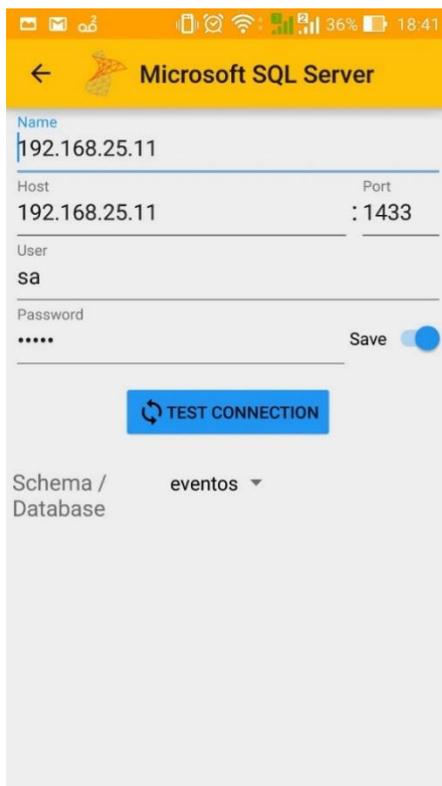
Figura 4.2 - *Feedback* inapropriado na execução de comandos de alteração de dados.



Fonte: SQL Client (2016).

- Mensagem de erro não ajuda o usuário a corrigir o problema
  - Descrição: Na tela de execução de consultas, quando uma consulta ou comando é escrita usando o nome de uma tabela, sem especificar o banco de dados e o *schema* a qual a tabela pertence uma mensagem de erro é apresentada apesar de o aplicativo estar conectado a base em que a tabela está armazenada e o *schema* da tabela ser o default. Normalmente, nessa situação não é necessário especificar o nome completo do objeto [banco de dados].[*schema*].[tabela], mas como para o aplicativo é necessário, a mensagem de erro deveria deixar isso claro, ajudando a corrigir o problema.
  - Heurística violada: Ajudar o usuário a reconhecer, diagnosticar e corrigir erros.
- Nenhum botão disponível para salvar a conexão
  - Descrição: Na tela de cadastro de conexão não existe um botão para salvar a conexão criada, sendo necessário habilitar a opção *save* e clicar no botão voltar. Isso pode ser observado na Figura 4.3.
  - Heurística violada: Compatibilidade do sistema com o mundo real.

Figura 4.3 - Tela de cadastro de conexões não possui um botão para confirmar a criação.



Fonte: SQL Client (2016).

- Símbolo utilizado para o botão de ordenação das tabelas ou *views*
  - Descrição: Nas telas de listagem de tabelas e de *views* existe um botão no canto superior direito da tela que serve para realizar ordenação das tabelas ou *views* listadas. Esse botão utiliza um símbolo que não deixa claro a sua função. Normalmente se utiliza símbolos com setas para baixo e/ou para cima para representar ordenação.
  - Heurística violada: Compatibilidade do sistema com o mundo real.
- Símbolo utilizado para o botão de filtragem das tabelas ou *views*
  - Descrição: Nas telas de listagem de tabelas e de *views* existe um botão no canto superior direito da tela que serve para realizar filtragem das tabelas ou *views* listadas. Esse botão utiliza um símbolo que não deixa claro a sua função. Normalmente se utiliza símbolos com um funil para representar filtros.
  - Heurística violada: Compatibilidade do sistema com o mundo real.
- Opção de parar a execução de um comando/consulta indisponível
  - Descrição: Na tela de execução de consultas, quando uma consulta ou comando é executado não existe a opção de parar a execução. Caso a execução demore

muito ou for identificado algum problema, o usuário tem que esperar o fim da execução da consulta ou comando.

- Heurística violada: Controle do usuário e liberdade.

### 4.2.1.3. Percurso Cognitivo

Entre os cenários criados, apenas o de realização de consultas (1) pôde ser avaliado nessa aplicação. Os demais cenários envolvem atividades não contempladas por esse aplicativo.

Segue o resultado da aplicação do método de percurso cognitivo nesse aplicativo:

- Realização de consultas
  - Clicar no botão '+':
    - **Os usuários saberão qual a ação correta para atingir o resultado desejado?**  
Não. Não existe nenhum título ou informação que identifique a tela inicial como a tela de criação e seleção de conexões e o botão '+' por si só não deixa clara a sua função.
    - **Os usuários perceberão que a ação correta está disponível?**  
Sim. Na tela inicial o botão '+' está bem visível.
    - **Os usuários irão associar a ação correta com o resultado desejado?**  
Não. Apesar do botão '+' ser associado pelo usuário à criação de novos objetos, o usuário não sabe do que se trata essa tela, logo não sabe o que o botão '+' vai criar ou adicionar.
    - **Se a ação correta for executada os usuários perceberão que foi feito avanço em relação à tarefa desejada?**  
Sim. Após clicar no botão '+' o aplicativo vai para a tela de cadastro de conexão, deixando claro que o usuário precisa cadastrar o servidor.
  - Entrar com as informações de conexão:
    - **Os usuários saberão qual a ação correta para atingir o resultado desejado?**  
Sim. Na tela de cadastro de conexão existe um formulário solicitando os dados necessários para a conexão com um banco de dados, deixando claro ao usuário que ele precisa conectar ao banco antes de realizar consultas.

- **Os usuários perceberão que a ação correta está disponível?**  
Sim. Na tela de cadastro de conexão o formulário para o cadastro de conexões está bem visível.
- **Os usuários irão associar a ação correta com o resultado desejado?**  
Sim. Os campos do formulário são informações necessárias para a conexão com um banco de dados, deixando claro para o usuário que é necessário preenche-los para cadastrar a conexão.
- **Se a ação correta for executada os usuários perceberão que foi feito avanço em relação à tarefa desejada?**  
Sim. Após preencher os campos o usuário sabe que basta salvar a conexão.
- Salvar a conexão:
  - **Os usuários saberão qual a ação correta para atingir o resultado desejado?**  
Não. Após a entrada dos dados de conexão não existe nenhum botão para conectar ou salvar a conexão, sendo necessário que o usuário volte à tela anterior após cadastrar a conexão. Isso pode ser observado na Figura 4.3.
  - **Os usuários perceberão que a ação correta está disponível?**  
Sim. O botão de voltar está bem visível na tela de cadastro de conexão.
  - **Os usuários irão associar a ação correta com o resultado desejado?**  
Não. A ação correta seria apertar no botão de voltar para salvar a conexão, porém isso não é nada comum e os usuários não iriam identificar essa funcionalidade do botão voltar.
  - **Se a ação correta for executada os usuários perceberão que foi feito avanço em relação à tarefa desejada?**  
Sim. Após apertar no botão voltar o usuário verá a conexão recém-criada na lista de conexões.
- Selecionar a conexão criada:
  - **Os usuários saberão qual a ação correta para atingir o resultado desejado?**  
Sim. Após criar a conexão os usuários saberão por experiência que precisam usá-la para se conectar ao banco.
  - **Os usuários perceberão que a ação correta está disponível?**

Sim. As conexões criadas vão estar listadas na tela, e, devido à experiência com outros aplicativos *mobile*, o usuário vai saber que pode clicar nela.

▪ **Os usuários irão associar a ação correta com o resultado desejado?**

Sim. As conexões listadas possuem o nome da conexão cadastrado quando a conexão é criada e, devido à experiência com outros aplicativos *mobile*, vão entender que se clicarem nela irão se conectar ao banco.

▪ **Se a ação correta for executada os usuários perceberão que foi feito avanço em relação à tarefa desejada?**

Sim. O aplicativo irá para a tela principal com o nome da conexão no título, deixando claro ao usuário que a conexão foi realizada.

○ Ir para a tela de escrita de consulta:

▪ **Os usuários saberão qual a ação correta para atingir o resultado desejado?**

Sim. Após se conectar, o usuário vai ver a tela com o título *SQL QUERIES*, vai ver a opção '+' e vai associar com a criação de consulta.

▪ **Os usuários perceberão que a ação correta está disponível?**

Sim. Na tela principal, na aba *SQL QUERIES* a opção '+' está bem visível.

▪ **Os usuários irão associar a ação correta com o resultado desejado?**

Sim. O usuário sabe que está na tela com o título *SQL QUERIES* e sabe, devido à experiência em aplicativos *mobile*, que o símbolo '+' é normalmente associado à criação de um novo objeto, logo, nessa situação, quer dizer criar nova *query* (consulta).

▪ **Se a ação correta for executada os usuários perceberão que foi feito avanço em relação à tarefa desejada?**

Sim. O aplicativo irá para uma tela com espaço para digitar com o nome *QUERY* no título.

○ Escrever consulta:

▪ **Os usuários saberão qual a ação correta para atingir o resultado desejado?**

Sim. O usuário tem conhecimento básico sobre a escrita de consultas.

▪ **Os usuários perceberão que a ação correta está disponível?**

Sim. Existe um campo para digitar consultas e o teclado para permitir a digitação.

- **Os usuários irão associar a ação correta com o resultado desejado?**

Sim. O usuário saberá que deve digitar sua consulta por causa do teclado e por causa do nome *QUERY* no título da tela.

- **Se a ação correta for executada os usuários perceberão que foi feito avanço em relação à tarefa desejada?**

Sim. Após escrever a consulta o usuário sabe que basta executá-la para ver o resultado.

- Executar consulta:

- **Os usuários saberão qual a ação correta para atingir o resultado desejado?**

Sim. Após escrever a consulta o usuário sabe que basta executá-la para ver o resultado.

- **Os usuários perceberão que a ação correta está disponível?**

Sim. O botão de *play* está bem visível na tela de escrita de consultas.

- **Os usuários irão associar a ação correta com o resultado desejado?**

Sim. O usuário saberá que deve clicar no botão de *play* para executar a consulta por experiência em ferramentas de conexão a bancos de dados de ambientes *desktop* que costumam utilizar o mesmo símbolo.

- **Se a ação correta for executada os usuários perceberão que foi feito avanço em relação à tarefa desejada?**

Sim. Após a realização da consulta o aplicativo irá apresentar os dados solicitados na consulta em formato tabular.

## 4.2.2. Mobile SQL Client

O Mobile SQL Client<sup>22</sup> é uma ferramenta de acesso a bancos de dados remotos que permite conexão a sistemas de bancos de dados Microsoft SQL Server a partir de um *smartphone* ou *tablet* com SO Android. Esse aplicativo tem como funções executar e salvar consultas e comandos, criar *triggers*, criar *procedures* e listar tabelas e *procedures* existentes no banco.

---

<sup>22</sup> [https://play.google.com/store/apps/details?id=com.yazilimdefteri.sqlclient&hl=pt\\_BR](https://play.google.com/store/apps/details?id=com.yazilimdefteri.sqlclient&hl=pt_BR)

O Mobile SQL Client é um aplicativo gratuito desenvolvido por yazilimdefteri<sup>23</sup> com mais de 10.000 *downloads* e com nota de 3,2 na Google Play numa escala de 0 a 5, sendo essa nota atribuída pela avaliação de 122 usuários.

#### 4.2.2.1. Checklist de Funções

Dentre as funções considerada básicas para uma ferramenta de acesso a banco de dados remoto, esse aplicativo permite a realização de consultas (1), realização de comandos (2), visualização de dados em formato tabular (3) e visualização de tabelas (4.1), *views* (4.2) e *procedures* (4.3). Não é possível visualizar por meio da interface os demais objetos de banco de dados como *functions* (4.5) e *triggers* (4.6).

Não é possível realizar alterações estruturais (5) ou alterações de dados (6) via interface. Podem ser realizados comandos de alterações de estrutura (5) ou dados (6), porém, nenhuma interface ou suporte é fornecido ao usuário pela aplicação para a execução dessas tarefas.

#### 4.2.2.2. Avaliação Heurística

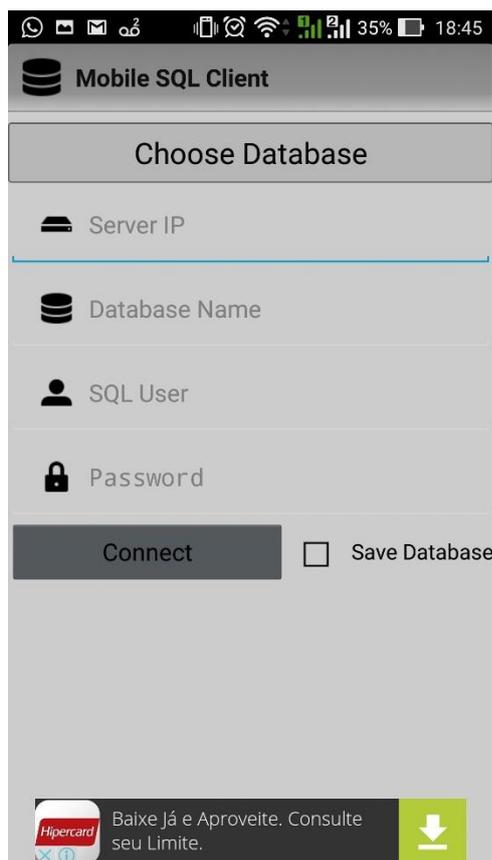
A aplicação do método de avaliação heurística nesse aplicativo resultou na seguinte lista de problemas:

- Seleção de conexão salva
  - Descrição: Na tela de seleção de conexão estão sempre presentes os campos para a criação de uma nova conexão ou edição de uma existente (Figura 4.4). Na maioria das vezes o usuário vai usar conexões salvas e não será necessário ver os detalhes dessa conexão. Os campos para criação e edição de conexões devem aparecer apenas quando necessário.
  - Heurística violada: Estética e *design* minimalista.

---

<sup>23</sup> <https://play.google.com/store/apps/developer?id=yazilimdefteri>

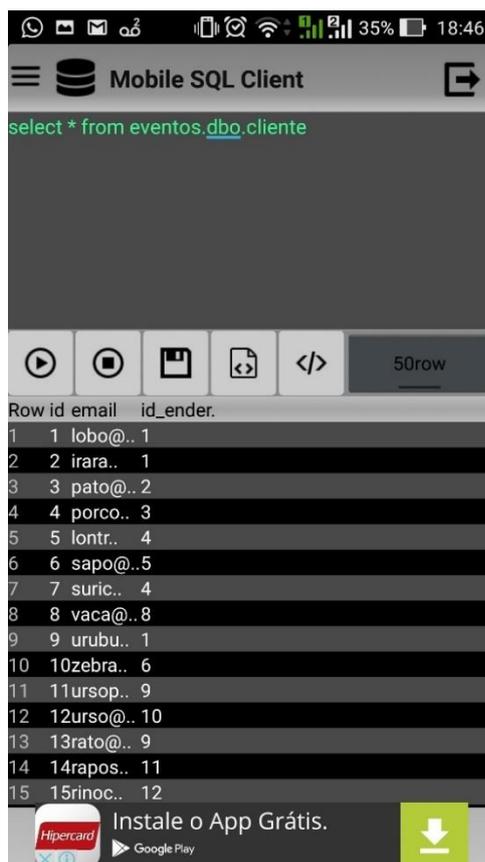
Figura 4.4 - Tela de seleção de conexões e cadastro de novas conexões.



Fonte: Mobile SQL Client (2016).

- Salvar conexão
  - Descrição: Na tela de escolha de conexão, depois de digitar as informações, existe a possibilidade de salvar a conexão. Em vez de conexão, o aplicativo chama de *database* (banco de dados), por se conectar a um banco específico em vez de se conectar a toda a instância. Quando uma conexão (ou *database*) é salva ela fica armazenada com o nome do banco de dados apenas. Por causa disso não é possível salvar conexões a diferentes bancos de dados com o mesmo nome em servidores diferentes, ou com credenciais diferentes ao mesmo banco.
  - Heurística violada: Compatibilidade do sistema com o mundo real.
- Visualização dos resultados comprometida por causa do pouco espaço e flexibilidade
  - Descrição: A tela de execução de consultas possui muita informação e nenhuma flexibilidade para permitir a melhor visualização dos resultados das consultas, deixando os valores das colunas impressados (Figura 4.5). Muitas vezes os valores das colunas não podem ser visualizados completamente na tabela.
  - Heurística violada: Estética e *design* minimalista.

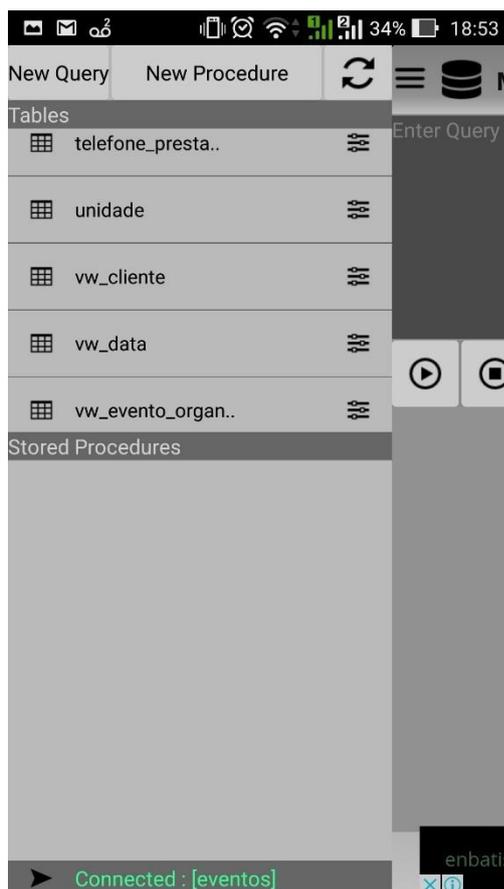
Figura 4.5 - Tela de escrita e execução de consultas.



Fonte: Mobile SQL Client (2016).

- Botões pouco utilizados ocupando espaço da tela (salvar consulta, carregar consulta e limite de 50 linhas)
  - Descrição: Na tela de execução de consultas existem vários botões disponíveis que são raramente necessários, ocupando espaço da tela (Figura 4.5). Opções como salvar consulta, carregar consulta e, principalmente, opção de limite de linhas são opções não usadas com muita frequência, podendo estar disponíveis por meio de um menu local, deixando a tela mais limpa apenas com as opções mais frequentemente utilizadas.
  - Heurística violada: Estética e *design* minimalista.
- *Views* são apresentadas na listagem de tabelas
  - Descrição: No menu lateral da aplicação são apresentadas as tabelas, *views* e procedures existentes no banco de dados. As *views* são apresentadas dentro da seção de tabelas, sem nenhuma distinção (Figura 4.6). Tabelas e *views* são objetos diferentes num banco de dados, e devem ser claramente distinguidos.
  - Heurística violada: Compatibilidade do sistema com o mundo real.

Figura 4.6 - Menu lateral com a lista de tabelas, *views* e procedures.



Fonte: Mobile SQL Client (2016).

### 4.2.2.3. Percurso Cognitivo

Entre os cenários criados, apenas o de realização de consultas (1) pôde ser avaliado nessa aplicação. Os demais cenários envolvem atividades não contempladas por esse aplicativo.

Segue o resultado da aplicação do método de percurso cognitivo nesse aplicativo:

- Realização de consultas
  - Entrar com as informações de conexão:
    - **Os usuários saberão qual a ação correta para atingir o resultado desejado?**  
Sim. Na tela inicial existe um formulário solicitando os dados necessários para a conexão com um banco de dados, deixando claro para o usuário que ele precisa conectar ao banco antes de realizar consultas.
    - **Os usuários perceberão que a ação correta está disponível?**

Sim. Na tela inicial o formulário para o cadastro de conexões e o botão para realizar a conexão estão bem visíveis.

▪ **Os usuários irão associar a ação correta com o resultado desejado?**

Sim. Os campos do formulário são informações necessárias para a conexão com um banco de dados e o botão conectar deixa claro que após inserir as informações ele deve ser pressionado.

▪ **Se a ação correta for executada os usuários perceberão que foi feito avanço em relação à tarefa desejada?**

Sim. Após preencher os campos o usuário sabe que basta salvar a conexão e usá-la.

○ Conectar ao banco de dados:

▪ **Os usuários saberão qual a ação correta para atingir o resultado desejado?**

Sim. Na tela inicial existe botão com a palavra *connect* abaixo do formulário, deixando claro que depois de preencher os campos basta clicar no botão para realizar a conexão.

▪ **Os usuários perceberão que a ação correta está disponível?**

Sim. Na tela inicial o botão *connect* está bem visível.

▪ **Os usuários irão associar a ação correta com o resultado desejado?**

Sim. O botão *connect* deixa claro que após inserir as informações sobre a conexão no formulário ele deve ser pressionado para realizar a conexão.

▪ **Se a ação correta for executada os usuários perceberão que foi feito avanço em relação à tarefa desejada?**

Sim. Após a conexão o aplicativo vai para a tela de execução de consultas, dando ao usuário certeza que a conexão foi realizada.

○ Escrever consulta:

▪ **Os usuários saberão qual a ação correta para atingir o resultado desejado?**

Sim. O usuário tem conhecimento básico sobre a escrita de consultas.

▪ **Os usuários perceberão que a ação correta está disponível?**

Sim. Existe um campo para digitar consultas e o teclado para permitir a digitação.

▪ **Os usuários irão associar a ação correta com o resultado desejado?**

Sim. O usuário saberá que deve digitar sua consulta por causa do teclado e por causa das palavras *Enter Query* no campo de digitação de consultas.

- **Se a ação correta for executada os usuários perceberão que foi feito avanço em relação à tarefa desejada?**

Sim. Após escrever a consulta o usuário sabe que basta executá-la para ver o resultado.

- Executar consulta:

- **Os usuários saberão qual a ação correta para atingir o resultado desejado?**

Sim. Após escrever a consulta o usuário sabe que basta executá-la para ver o resultado.

- **Os usuários perceberão que a ação correta está disponível?**

Sim. O botão de *play* está bem visível na tela de escrita de consultas.

- **Os usuários irão associar a ação correta com o resultado desejado?**

Sim. O usuário saberá que deve clicar no botão de *play* para executar a consulta por experiência em ferramentas de conexão a bancos de dados de ambientes *desktop* que costumam utilizar o mesmo símbolo.

- **Se a ação correta for executada os usuários perceberão que foi feito avanço em relação à tarefa desejada?**

Sim. Após a realização da consulta o aplicativo irá apresentar os dados solicitados na consulta em formato tabular.

### 4.2.3. SQL Studio Lite

O SQL Studio Lite<sup>24</sup> é uma ferramenta de acesso a bancos de dados remotos que permite conexão a sistemas de bancos de dados Microsoft SQL Server a partir de um *smartphone* ou *tablet* com SO Android. Esse aplicativo tem como funções executar e salvar consultas e comandos, exportar resultado de consultas em arquivos CSV, listar tabelas, *views*, *procedures*, *functions* e *triggers* e visualizar e alterar o código de *views*, *procedures*, *functions* e *triggers* existentes no banco.

---

<sup>24</sup> [https://play.google.com/store/apps/details?id=com.codeBuilder.sqlStudioLite&hl=pt\\_BR](https://play.google.com/store/apps/details?id=com.codeBuilder.sqlStudioLite&hl=pt_BR)

O SQL Studio Lite é um aplicativo gratuito desenvolvido por codeBuilder<sup>25</sup> com mais de 10.000 *downloads* e com nota de 3,1 na Google Play numa escala de 0 a 5, sendo essa nota atribuída pela avaliação de 146 usuários. Apesar de ser gratuito para uso, esse aplicativo possui limitações no número de linhas retornadas por consulta, permitindo apenas 10 linhas, e no número de conexões, permitindo apenas o registro de um servidor. A versão completa do aplicativo é o SQL Studio e custa R\$4,01 reais.

#### 4.2.3.1. Checklist de Funções

Dentre as funções considerada básicas para uma ferramenta de acesso a banco de dados remoto, esse aplicativo permite a realização de consultas (1), realização de comandos (2), visualização de dados em formato tabular (3) e visualização de tabelas (4.1), *views* (4.2), *procedures* (4.3), *functions* (4.4) e *triggers* (4.5).

Não é possível realizar alterações estruturais (5) ou alterações de dados (6) via interface. Podem ser realizados comandos de alterações de estrutura (5) ou dados (6), porém, nenhum suporte é fornecido ao usuário pela aplicação para a execução dessas tarefas.

#### 4.2.3.2. Avaliação Heurística

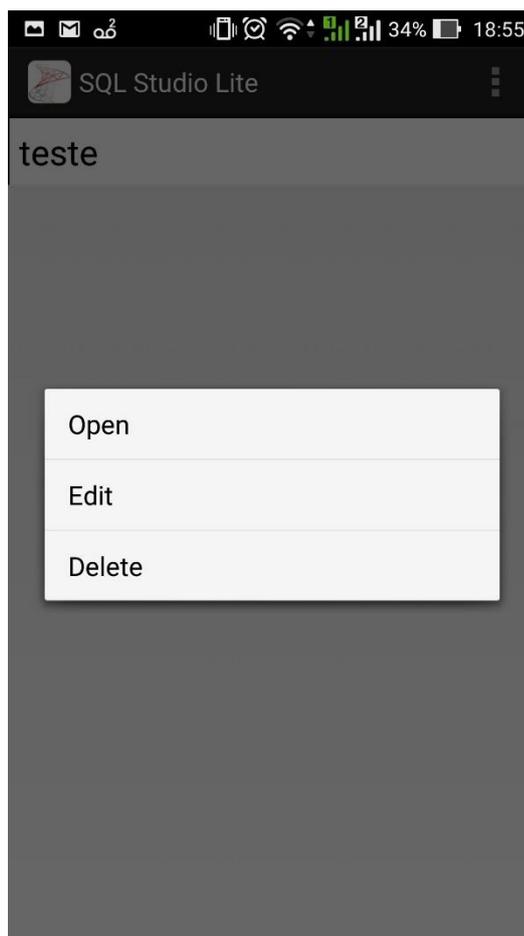
A aplicação do método de avaliação heurística nesse aplicativo resultou na seguinte lista de problemas:

- Segurar para aparecer opções na conexão
  - Descrição: Na tela de seleção de conexões, não fica claro para o usuário que existem opções de interação com cada conexão (Figura 4.7). Para abrir essas opções o usuário tem que clicar e segurar na conexão. Uma alternativa para manter a estética minimalista é cada conexão possuir um botão que deixe claro que existem opções disponíveis. Normalmente são usados para essa função símbolos compostos de três (3) pontos verticais.
  - Heurística violada: Reconhecimento ao invés de lembrança.

---

<sup>25</sup> <https://play.google.com/store/apps/developer?id=codeBuilder>

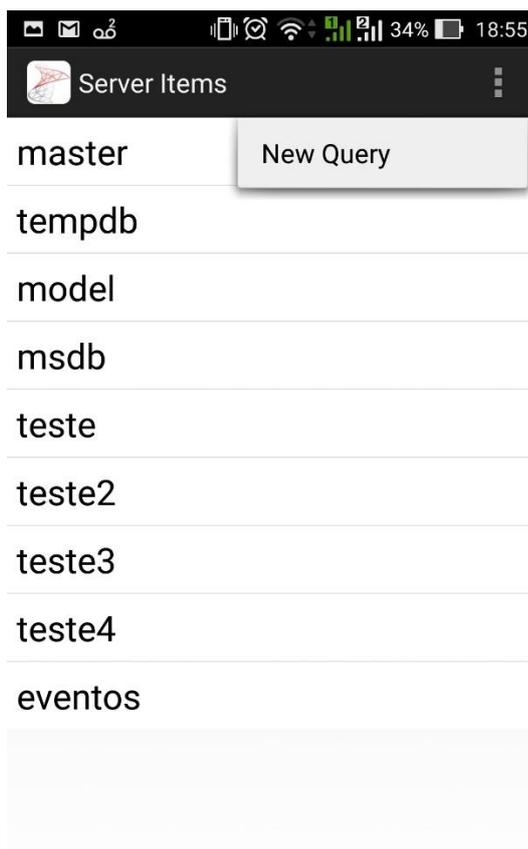
Figura 4.7 - Menu local da conexão.



Fonte: SQL Studio Lite (2016).

- Botão de executar consulta/comando não visível
  - Descrição: Na tela de execução de consultas, para executar uma consulta ou comando o usuário deve utilizar o menu local e clicar na opção de rodar consulta. Execução de consultas e comandos é a função desta tela, por isso essa opção é muito utilizada. Opções frequentemente utilizadas devem estar sempre visíveis e facilmente acessadas.
  - Heurística violada: Reconhecimento ao invés de lembrança.
- Acesso à tela de execução de consultas não visível
  - Descrição: A principal função desta aplicação é a execução de consultas e comandos, logo essa função deve estar visível e facilmente acessível a partir das principais telas. Para acessar a tela de escrita de consultas é necessário clicar no menu local da tela, como mostrado na Figura 4.8.
  - Heurística violada: Reconhecimento ao invés de lembrança.

Figura 4.8 - Opção de ir para interface de escrita de consultas dentro do menu local da tela.



Fonte: SQL Studio Lite (2016).

- *Feedback* durante a espera para a execução de uma consulta
  - Descrição: Na tela de execução de consultas, durante a execução de consultas e comandos o aplicativo não fornece nenhum *feedback* para o usuário de que a consulta ou comando está sendo executado.
  - Heurística violada: Visibilidade do *status* do sistema.
- Opção de parar a execução de um comando/consulta indisponível
  - Descrição: Na tela de execução de consultas, quando uma consulta ou comando é executado não existe a opção de parar a execução. Caso a execução demore muito ou for identificado algum problema, o usuário tem que esperar o fim da execução da consulta ou comando.
  - Heurística violada: Controle do usuário e liberdade.

### 4.2.3.3. Percurso Cognitivo

Entre os cenários criados, apenas o de realização de consultas (1) pôde ser avaliado nessa aplicação. Os demais cenários envolvem atividades não contempladas por esse aplicativo.

Segue o resultado da aplicação do método de percurso cognitivo nesse aplicativo:

- Realização de consultas
  - Clicar na opção *Add New Server*:
    - **Os usuários saberão qual a ação correta para atingir o resultado desejado?**

Não. Não existe nenhum título ou informação que identifique a tela inicial como a tela de criação e seleção de conexões.
    - **Os usuários perceberão que a ação correta está disponível?**

Sim, mas com alguma dificuldade. Para criar uma nova conexão é necessário clicar no menu local para depois clicar em *Add New Server*, logo essa opção não está com uma boa visibilidade.
    - **Os usuários irão associar a ação correta com o resultado desejado?**

Sim. Após encontrar a opção *Add New Server* o usuário vai entender que essa opção é utilizada para criar uma nova conexão.
    - **Se a ação correta for executada os usuários perceberão que foi feito avanço em relação à tarefa desejada?**

Sim. Após clicar no botão *Add New Server* o aplicativo vai para a tela de cadastro de conexão, deixando claro que o usuário precisa cadastrar o servidor.
  - Entrar com as informações de conexão:
    - **Os usuários saberão qual a ação correta para atingir o resultado desejado?**

Sim. Na tela de cadastro de conexão existe um formulário solicitando os dados necessários para a conexão com um banco de dados, deixando claro ao usuário que ele precisa conectar ao banco antes de realizar consultas.
    - **Os usuários perceberão que a ação correta está disponível?**

Sim. Na tela de cadastro de conexão o formulário para o cadastro de conexões está bem visível.

- **Os usuários irão associar a ação correta com o resultado desejado?**  
Sim. Os campos do formulário são informações necessárias para a conexão com um banco de dados, deixando claro para o usuário que é necessário preenche-los para cadastrar a conexão.
- **Se a ação correta for executada os usuários perceberão que foi feito avanço em relação à tarefa desejada?**  
Sim. Após preencher os campos o usuário sabe que basta salvar a conexão.
- Salvar a conexão:
  - **Os usuários saberão qual a ação correta para atingir o resultado desejado?**  
Sim. Na tela de cadastro de conexão existe botão com a palavra *save* abaixo do formulário, deixando claro que depois de preencher os campos basta clicar no botão para salvar a conexão.
  - **Os usuários perceberão que a ação correta está disponível?**  
Sim. O botão *save* está bem visível na tela de cadastro de conexão.
  - **Os usuários irão associar a ação correta com o resultado desejado?**  
Sim. O botão *save* deixa claro que após inserir as informações sobre a conexão no formulário ele deve ser pressionado para salvar a conexão.
  - **Se a ação correta for executada os usuários perceberão que foi feito avanço em relação à tarefa desejada?**  
Sim. Após apertar no botão *save* o usuário verá a conexão recém-criada na lista de conexões.
- Selecionar a conexão criada:
  - **Os usuários saberão qual a ação correta para atingir o resultado desejado?**  
Sim. Após criar a conexão os usuários saberão por experiência que precisam usá-la para se conectar ao banco.
  - **Os usuários perceberão que a ação correta está disponível?**  
Sim. As conexões criadas vão estar listadas na tela, e, devido à experiência com outros aplicativos *mobile*, o usuário vai saber que pode clicar nela.
  - **Os usuários irão associar a ação correta com o resultado desejado?**

Sim. As conexões listadas possuem o nome da conexão cadastrado quando a conexão é criada e, devido à experiência com outros aplicativos *mobile*, vão entender que se clicarem nela irão se conectar ao banco.

- **Se a ação correta for executada os usuários perceberão que foi feito avanço em relação à tarefa desejada?**

Sim. O aplicativo irá para outra tela com a lista de bancos de dados existentes, deixando claro ao usuário que a conexão foi realizada.

- Ir para a tela de escrita de consulta:

- **Os usuários saberão qual a ação correta para atingir o resultado desejado?**

Não. Após se conectar, o usuário não vai ver nenhuma opção ou informação visível que o ajude a descobrir como acessar a tela de escrita de consultas.

- **Os usuários perceberão que a ação correta está disponível?**

Sim, mas com alguma dificuldade. Para ir para a tela de escrita de consultas é necessário clicar no menu local para depois clicar em *New Query*, logo essa opção não está com uma boa visibilidade.

- **Os usuários irão associar a ação correta com o resultado desejado?**

Sim. Após encontrar a opção *New Query* o usuário vai entender que essa opção é utilizada para ir para a tela de escrita de consultas.

- **Se a ação correta for executada os usuários perceberão que foi feito avanço em relação à tarefa desejada?**

Sim. O aplicativo irá para uma tela com espaço para digitar com o nome *SQL Query* no título.

- Escrever consulta:

- **Os usuários saberão qual a ação correta para atingir o resultado desejado?**

Sim. O usuário tem conhecimento básico sobre a escrita de consultas.

- **Os usuários perceberão que a ação correta está disponível?**

Sim. Existe um campo para digitar consultas e o teclado para a digitação.

- **Os usuários irão associar a ação correta com o resultado desejado?**

Sim. O usuário saberá que deve digitar sua consulta por causa do teclado e por causa do nome *SQL Query* no título da tela.

- **Se a ação correta for executada os usuários perceberão que foi feito avanço em relação à tarefa desejada?**  
Sim. Após escrever a consulta o usuário sabe que basta executá-la para ver o resultado.
- Executar consulta:
  - **Os usuários saberão qual a ação correta para atingir o resultado desejado?**  
Sim. Após escrever a consulta o usuário sabe que basta executá-la para ver o resultado por experiência.
  - **Os usuários perceberão que a ação correta está disponível?**  
Sim, mas com alguma dificuldade. Para executar a consulta é necessário clicar no menu local para depois clicar em *Run Query*, logo essa opção não está com uma boa visibilidade.
  - **Os usuários irão associar a ação correta com o resultado desejado?**  
Sim. Após encontrar a opção *Run Query* o usuário vai entender que essa opção é utilizada para executar a consulta digitada.
  - **Se a ação correta for executada os usuários perceberão que foi feito avanço em relação à tarefa desejada?**  
Sim. Após a realização da consulta o aplicativo irá apresentar os dados solicitados na consulta em formato tabular.

#### 4.2.4. Mobile SQL Server Manager

O Mobile SQL Server Manager<sup>26</sup> é uma ferramenta de acesso a bancos de dados remotos que permite conexão a instancias Microsoft SQL Server a partir de um *smarphone* ou *tablet* com SO Android. Esse aplicativo tem como funções executar consultas e comandos, listar tabelas, *views*, *procedures* e *functions*, editar *logins*, usuários, *views*, *procedures* e *functions*, criar e remover bancos de dados, *logins*, usuários, tabelas, *views*, *procedures* e *functions*, realizar *backup* e *restore* de bancos de dados, gerenciar *jobs* e listar informações do log do SQL Server.

---

<sup>26</sup> <https://play.google.com/store/apps/details?id=com.mobilesigma.mobilesqlservermanager>

O Mobile SQL Server Manager é um aplicativo desenvolvido por Bruce Mobini<sup>27</sup> com mais de 100 *downloads* e com nota de 4,3 na Google Play numa escala de 0 a 5, sendo essa nota atribuída pela avaliação de 7 usuários. Esse aplicativo pode ser adquirido pelo valor de R\$12,71 reais.

#### 4.2.4.1. Checklist de Funções

Dentre as funções considerada básicas para uma ferramenta de acesso a banco de dados remoto, esse aplicativo permite a realização de consultas (1), realização de comandos (2), visualização de dados em formato tabular (3), visualização de tabelas (4.1), *views* (4.2), *procedures* (4.3) e *functions* (4.4) e criação e remoção de tabelas (5.1 e 5.2).

Apesar de ser um dos aplicativos desse tipo mais completo para Android e possuir diversas funções avançadas de gerenciamento de dados, esse aplicativo não possui diversas funções consideradas básicas. Não é possível realizar alterações estruturais exceto criação e remoção de tabelas (5.3 e 5.4) ou alterações de dados (6) via interface. Podem ser realizados comandos de alterações de estrutura (5) ou dados (6), porém, nenhum suporte é fornecido ao usuário pela aplicação para a execução dessas tarefas.

#### 4.2.4.2. Avaliação Heurística

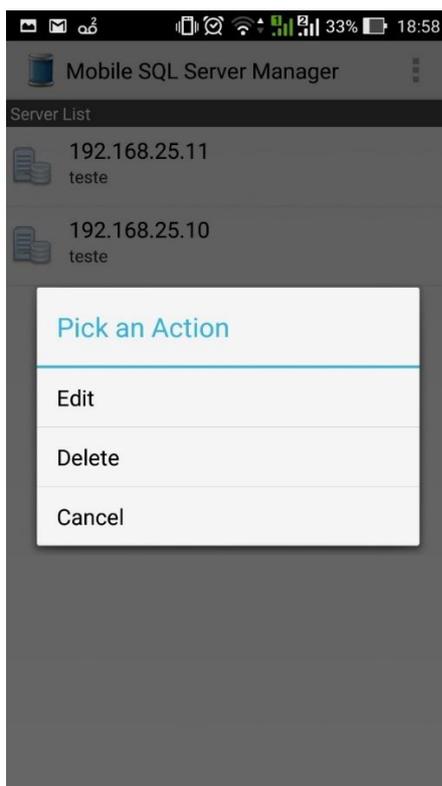
A aplicação do método de avaliação heurística nesse aplicativo resultou na seguinte lista de problemas:

- Segurar para aparecer opções
  - Descrição: Em praticamente todas as telas do aplicativo, não fica claro para o usuário que existem opções de interação com cada objeto (e.g. conexões, bancos de dados, tabelas), isso pode ser observado na Figura 4.9. Para abrir essas opções o usuário tem que clicar e segurar na conexão. Como alternativa, para manter a estética minimalista, cada objeto pode possuir um botão que deixe claro que existem opções disponíveis. Normalmente são usados para essa função símbolos compostos de três (3) pontos verticais.
  - Heurística violada: Reconhecimento ao invés de lembrança.

---

<sup>27</sup> <https://play.google.com/store/apps/developer?id=Bruce+Mobini>

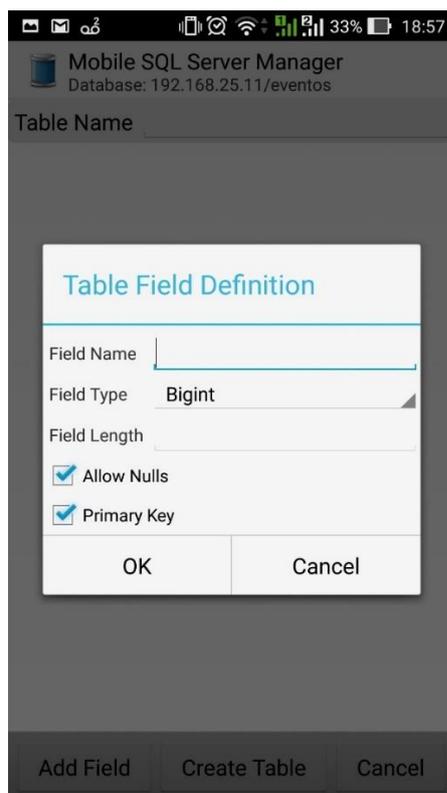
Figura 4.9 - Menu local da conexão.



Fonte: Mobile SQL Server Manager (2016).

- *Feedback* de falha na conexão
  - Descrição: Após escolher uma conexão existente, aparece a lista de possíveis objetos e ações do banco separados por pasta. Caso a conexão selecionada esteja com problema, a mensagem erro só aparece quando o usuário tenta entrar em alguma dessas pastas. Essa mensagem de erro deve ser apresentada no momento em que o usuário tenta conectar à instância.
  - Heurística violada: Visibilidade do *status* do sistema.
- Adição de nova coluna em tabela que está sendo criada
  - Descrição: Na tela de adição de coluna a uma tabela que está sendo criada, o aplicativo permite que o usuário marque opções conflitantes (*primary key* e *allow null*), como pode ser visto na Figura 4.10. O aplicativo apresenta erro quando o usuário tenta adicionar a coluna ao invés de impedir que o erro ocorra.
  - Heurística violada: Prevenção de erros.

Figura 4.10 - Tela de adição de coluna.



Fonte: Mobile SQL Server Manager (2016).

- Acesso a tela de criação de tabelas e bancos de dados não visível
  - Descrição: A criação de tabelas é uma função importante em ferramentas de acesso a bancos de dados, logo essa função deve estar visível e facilmente acessível a partir da tela de tabelas existentes.
  - Heurística violada: Reconhecimento ao invés de lembrança.
- Acesso a tela de execução de consultas não visível
  - Descrição: A execução de consultas e comandos é uma função importante em ferramentas de acesso a bancos de dados, logo essa função deve estar visível e facilmente acessível a partir das principais telas.
  - Heurística violada: Reconhecimento ao invés de lembrança.
- Opção de parar a execução de um comando/consulta indisponível
  - Descrição: Na tela de execução de consultas, quando uma consulta ou comando é executado não existe a opção de parar a execução. Caso a execução demore muito ou for identificado algum problema, o usuário tem que esperar o fim da execução da consulta ou comando.
  - Heurística violada: Controle do usuário e liberdade.

#### 4.2.4.3. Percurso cognitivo

Entre os cenários criados, apenas o de realização de consultas (1) e o de criação de tabela (3) puderam ser avaliados nessa aplicação. O cenário de inserção de dados (2) tem como base uma tarefa não contemplada por esse aplicativo.

Segue o resultado da aplicação do método de percurso cognitivo nesse aplicativo:

- Realização de consultas
  - Clicar na opção *Add Server*:
    - **Os usuários saberão qual a ação correta para atingir o resultado desejado?**  
Sim. O título *Server List* identifica a tela inicial como a tela de criação e seleção de conexões, deixando claro para o usuário que ele precisa se conectar antes de realizar consultas.
    - **Os usuários perceberão que a ação correta está disponível?**  
Sim, mas com alguma dificuldade. Para criar uma nova conexão é necessário clicar no menu local para depois clicar em *Add Server*, logo essa opção não está com uma boa visibilidade.
    - **Os usuários irão associar a ação correta com o resultado desejado?**  
Sim. Após encontrar a opção *Add Server* o usuário vai entender que essa opção é utilizada para criar uma nova conexão.
    - **Se a ação correta for executada os usuários perceberão que foi feito avanço em relação à tarefa desejada?**  
Sim. Após clicar no botão *Add Server* o aplicativo vai para a tela de cadastro de conexão, deixando claro que é necessário cadastrar o servidor.
  - Entrar com as informações de conexão:
    - **Os usuários saberão qual a ação correta para atingir o resultado desejado?**  
Sim. Na tela de cadastro de conexão existe um formulário solicitando os dados necessários para a conexão com um banco de dados, deixando claro para o usuário que ele precisa conectar ao banco antes de realizar consultas.
    - **Os usuários perceberão que a ação correta está disponível?**

Sim. Na tela de cadastro de conexão o formulário para o cadastro de conexões está bem visível.

▪ **Os usuários irão associar a ação correta com o resultado desejado?**

Sim. Os campos do formulário são informações necessárias para a conexão com um banco de dados, deixando claro para o usuário que é necessário preenche-los para cadastrar a conexão.

▪ **Se a ação correta for executada os usuários perceberão que foi feito avanço em relação à tarefa desejada?**

Sim. Após preencher os campos o usuário sabe que basta salvar ou usar a conexão.

○ Salvar a conexão:

▪ **Os usuários saberão qual a ação correta para atingir o resultado desejado?**

Não. Na tela de cadastro de conexão não existe opção visível para informar ao usuário se ele vai usar os dados preenchidos para se conectar ou salvar a conexão.

▪ **Os usuários perceberão que a ação correta está disponível?**

Sim, mas com alguma dificuldade. Para salvar a conexão é necessário clicar no menu local para depois clicar em *save*, logo essa opção não está com uma boa visibilidade.

▪ **Os usuários irão associar a ação correta com o resultado desejado?**

Sim. Após encontrar a opção *save* o usuário vai entender que essa opção é utilizada para salvar a conexão.

▪ **Se a ação correta for executada os usuários perceberão que foi feito avanço em relação à tarefa desejada?**

Sim. Após apertar no botão *save* o usuário verá a conexão recém-criada na lista de conexões.

○ Selecionar a conexão criada:

▪ **Os usuários saberão qual a ação correta para atingir o resultado desejado?**

Sim. Após criar a conexão os usuários saberão por experiência que precisam usá-la para se conectar ao banco.

▪ **Os usuários perceberão que a ação correta está disponível?**

Sim. As conexões criadas vão estar listadas na tela, e, devido à experiência com outros aplicativos *mobile*, o usuário vai saber que pode clicar nela.

- **Os usuários irão associar a ação correta com o resultado desejado?**

Sim. As conexões listadas possuem o nome da conexão cadastrado quando a conexão é criada e, devido à experiência com outros aplicativos *mobile*, vão entender que se clicarem nela irão se conectar ao banco.

- **Se a ação correta for executada os usuários perceberão que foi feito avanço em relação à tarefa desejada?**

Sim. O aplicativo irá para outra tela com um menu dos tipos de objetos de bancos de dados existentes, deixando claro que a conexão foi realizada.

- Clicar na pasta *Users Databases*:

- **Os usuários saberão qual a ação correta para atingir o resultado desejado?**

Não. Após se conectar o usuário vai procurar como ir para a tela de escrita de consultas, porém para fazer isso é necessário antes clicar na pasta *Users Databases* para selecionar a base na qual as consultas serão realizadas. O usuário não vai perceber isso.

- **Os usuários perceberão que a ação correta está disponível?**

Sim. A opção *Users Databases* está bem visível na tela.

- **Os usuários irão associar a ação correta com o resultado desejado?**

Sim. O usuário vai entender que ao clicar na pasta *Users Databases* o aplicativo vai mostrar os bancos de dados existentes.

- **Se a ação correta for executada os usuários perceberão que foi feito avanço em relação à tarefa desejada?**

Não. O usuário vai estar procurando a tela de escrita de consultas e não vai entender que foi necessário ir para a tela de *Users Databases*.

- Ir para a tela de escrita de consulta:

- **Os usuários saberão qual a ação correta para atingir o resultado desejado?**

Não. Após entrar na tela de *Users Databases* o usuário não vai ver nenhuma opção ou informação que o ajude a descobrir como acessar a tela de escrita de consultas.

- **Os usuários perceberão que a ação correta está disponível?**  
Não. Para ir para a tela de escrita de consultas é necessário clicar e segurar em um banco de dado para aparecer uma lista de opções e depois clicar em *Query*. Não há nenhuma informação na tela que deixe claro que existem opções de interação com o objeto.
  - **Os usuários irão associar a ação correta com o resultado desejado?**  
Sim. Caso o usuário encontre a opção dentro do menu de opções do banco de dados ele vai associar a opção *Query* à tela de escrita de consultas.
  - **Se a ação correta for executada os usuários perceberão que foi feito avanço em relação à tarefa desejada?**  
Sim. O aplicativo irá para uma tela com espaço para digitar e com um botão *Execute*.
- Escrever consulta:
- **Os usuários saberão qual a ação correta para atingir o resultado desejado?**  
Sim. O usuário tem conhecimento básico sobre a escrita de consultas.
  - **Os usuários perceberão que a ação correta está disponível?**  
Sim. Existe um campo para digitar consultas e o teclado para a digitação.
  - **Os usuários irão associar a ação correta com o resultado desejado?**  
Sim. O usuário saberá que deve digitar sua consulta por causa do teclado e por causa do botão *Execute*.
  - **Se a ação correta for executada os usuários perceberão que foi feito avanço em relação à tarefa desejada?**  
Sim. Após escrever a consulta o usuário sabe que basta executá-la para ver o resultado.
- Executar consulta:
- **Os usuários saberão qual a ação correta para atingir o resultado desejado?**  
Sim. Após escrever a consulta o usuário sabe que basta executá-la para ver o resultado.
  - **Os usuários perceberão que a ação correta está disponível?**  
Sim. O botão *Execute* está bem visível na tela.
  - **Os usuários irão associar a ação correta com o resultado desejado?**

Sim. O usuário vai entender o botão *Execute* é utilizado para executar a consulta digitada.

- **Se a ação correta for executada os usuários perceberão que foi feito avanço em relação à tarefa desejada?**

Sim. Após a realização da consulta o aplicativo irá apresentar os dados solicitados na consulta em formato tabular.

- Criação de tabela

- Selecionar a conexão:

- **Os usuários saberão qual a ação correta para atingir o resultado desejado?**

Sim. Quando ver a lista de conexões disponíveis na tela com o título *Server List* o usuário vai saber que precisa usar uma das conexões.

- **Os usuários perceberão que a ação correta está disponível?**

Sim. As conexões criadas vão estar listadas na tela, e, devido à experiência com outros aplicativos *mobile*, o usuário vai saber que pode clicar nela.

- **Os usuários irão associar a ação correta com o resultado desejado?**

Sim. As conexões listadas possuem o nome da conexão cadastrado quando a conexão é criada e, devido à experiência com outros aplicativos *mobile*, vão entender que se clicarem nela irão se conectar ao banco.

- **Se a ação correta for executada os usuários perceberão que foi feito avanço em relação à tarefa desejada?**

Sim. O aplicativo irá para outra tela com um menu com os tipos de objetos de bancos de dados existentes, deixando claro ao usuário que a conexão foi realizada.

- Clicar na pasta *Users Databases*:

- **Os usuários saberão qual a ação correta para atingir o resultado desejado?**

Sim. Após se conectar e ver o menu o usuário vai entender que precisa ir para a tela de bancos de dados para escolher um para criar a tabela.

- **Os usuários perceberão que a ação correta está disponível?**

Sim. A opção *Users Databases* está bem visível na tela.

- **Os usuários irão associar a ação correta com o resultado desejado?**

Sim. O usuário vai entender que ao clicar na pasta *Users Databases* o aplicativo vai mostrar os bancos de dados existentes.

- **Se a ação correta for executada os usuários perceberão que foi feito avanço em relação à tarefa desejada?**

Sim. O aplicativo irá para uma tela com a lista de bancos de dados.

- Ir para a tela de tabelas do banco de dados:

- **Os usuários saberão qual a ação correta para atingir o resultado desejado?**

Sim. Após entrar na tela de *Users Databases* o usuário vai saber, por experiência, que o próximo passo é ir para a tela de tabelas do banco para criar uma nova tabela.

- **Os usuários perceberão que a ação correta está disponível?**

Não. Para ir para a tela de tabelas de um banco de dados é necessário clicar e segurar no banco escolhido na lista para aparecer uma lista de opções e depois clicar em *Tables*. Não há nenhuma informação na tela que deixe claro que existem opções de interação com o objeto.

- **Os usuários irão associar a ação correta com o resultado desejado?**

Sim. Caso o usuário encontre a opção dentro do menu de opções do banco de dados ele vai associar a opção *Tables* à tela de tabelas.

- **Se a ação correta for executada os usuários perceberão que foi feito avanço em relação à tarefa desejada?**

Sim. O aplicativo irá para uma tela com a lista de tabelas do banco.

- Ir para a tela de criação de tabelas:

- **Os usuários saberão qual a ação correta para atingir o resultado desejado?**

Sim. Após entrar na tela com a lista de tabelas, o usuário vai saber por experiência que o próximo passo é ir para a tela de criação de tabelas.

- **Os usuários perceberão que a ação correta está disponível?**

Não. Para ir para a tela de criação de tabelas é necessário clicar e segurar em uma tabela existente na lista para aparecer uma lista de opções e depois clicar em *New Table*. Não há nenhuma informação na tela que deixe claro que existem opções de interação com o objeto.

- **Os usuários irão associar a ação correta com o resultado desejado?**

Sim. Caso o usuário encontre a opção dentro do menu de opções da tabela, ele vai associar a opção *New Table* à tela de criação de tabelas.

- **Se a ação correta for executada os usuários perceberão que foi feito avanço em relação à tarefa desejada?**

Sim. O aplicativo irá para uma tela solicitando o nome da tabela e com um botão *Create Table*.

- Adicionar coluna:

- **Os usuários saberão qual a ação correta para atingir o resultado desejado?**

Sim. Após entrar na tela de criação de tabelas, o usuário vai saber por experiência que o próximo passo é criar as colunas da tabela.

- **Os usuários perceberão que a ação correta está disponível?**

Sim. O botão *Add Field* está bem visível.

- **Os usuários irão associar a ação correta com o resultado desejado?**

Sim. O botão *Add Field* na tela de criação de tabelas deixa claro para o usuário que se refere à criação de um campo (coluna).

- **Se a ação correta for executada os usuários perceberão que foi feito avanço em relação à tarefa desejada?**

Sim. O aplicativo irá para uma tela solicitando as informações da coluna que será criada.

- Preencher formulário:

- **Os usuários saberão qual a ação correta para atingir o resultado desejado?**

Sim. Após entrar na tela de criação de coluna, o usuário vai ver o formulário solicitando as informações da coluna que será criada.

- **Os usuários perceberão que a ação correta está disponível?**

Sim. O formulário está bem visível na tela.

- **Os usuários irão associar a ação correta com o resultado desejado?**

Sim. Os campos do formulário vão deixar claro que se trata de informações referentes à criação de coluna.

- **Se a ação correta for executada os usuários perceberão que foi feito avanço em relação à tarefa desejada?**

Sim. Após preencher o formulário, o usuário sabe por experiência que basta confirmar a criação da coluna.

- Confirmar criação da coluna:
  - **Os usuários saberão qual a ação correta para atingir o resultado desejado?**

Sim. Após preencher o formulário, o usuário sabe por experiência que basta confirmar a criação da coluna.
  - **Os usuários perceberão que a ação correta está disponível?**

Sim. O botão *OK* está bem visível na tela.
  - **Os usuários irão associar a ação correta com o resultado desejado?**

Sim. O usuário vai saber por experiência que após preencher o formulário o botão *OK* vai criar a coluna.
  - **Se a ação correta for executada os usuários perceberão que foi feito avanço em relação à tarefa desejada?**

Sim. O aplicativo irá mostrar na tela de criação de tabela a coluna adicionada.
- Criar tabela:
  - **Os usuários saberão qual a ação correta para atingir o resultado desejado?**

Sim. Após adicionar as colunas, o usuário vai saber que basta criar a tabela.
  - **Os usuários perceberão que a ação correta está disponível?**

Sim. O botão *Create Table* está bem visível na tela.
  - **Os usuários irão associar a ação correta com o resultado desejado?**

Sim. O botão *Create Table* diz exatamente o que ele faz.
  - **Se a ação correta for executada os usuários perceberão que foi feito avanço em relação à tarefa desejada?**

Sim. O aplicativo voltará para a tela com a lista de tabelas e a tabela recém-criada estará na lista.

### 4.2.5. RemoDB

O RemoDB<sup>28</sup> é uma ferramenta de acesso a bancos de dados remotos que permite conexão a diferentes SGBD a partir de um *smartphone* ou *tablet* com SO Android. O RemoDB dá suporte a conexões aos SGBD Microsoft SQL Server, MySQL, PostgreSQL e Sap Sybase ASE<sup>29</sup>. Esse aplicativo tem como funções executar e salvar consultas e comandos, exportar resultado de consultas em arquivos CSV, listar tabelas e *views*, apagar tabelas e *views* e inserir e atualizar dados.

O RemoDB é um aplicativo gratuito desenvolvido por Christian Castaldi<sup>30</sup> com mais de 1.000 *downloads* e com nota de 4,1 na Google Play numa escala de 0 a 5, sendo essa nota atribuída pela avaliação de 28 usuários. Apesar de ser gratuito para uso, esse aplicativo possui limitações. Apenas na versão completa podem ser realizadas exportação, inserção e atualização de dados. A versão completa pode ser liberada por R\$ 9,42 reais.

#### 4.2.5.1. Checklist de Funções

Dentre as funções considerada básicas para uma ferramenta de acesso à banco de dados remoto, esse aplicativo permite a realização de consultas (1), realização de comandos (2), visualização de dados em formato tabular (3), visualização de tabelas (4.1) e *views* (4.2) e inserção e atualização de dados (6.1 e 6.2). Uma limitação encontrada é que o aplicativo só encontra tabelas e *views* presente no *schema* padrão (para o caso de bancos de dados Microsoft SQL Server) e consequentemente só permite atualização e inserção de linhas nessas tabelas.

Não é possível realizar alterações estruturais (5) ou remoção de dados (6.3) via interface. Podem ser realizados comandos para desempenhar essas funções, porém, nenhuma interface ou suporte é fornecido ao usuário pela aplicação.

---

<sup>28</sup> <https://play.google.com/store/apps/details?id=com.kriskast.remotedb>

<sup>29</sup> <http://go.sap.com/product/data-mgmt/sybase-ase.html>

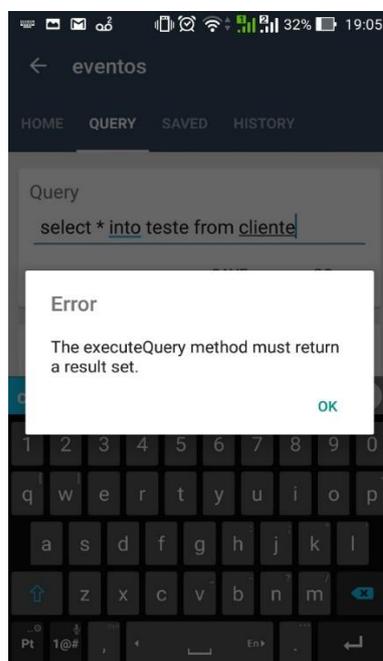
<sup>30</sup> <https://play.google.com/store/apps/developer?id=Christian+Castaldi>

### 4.2.5.2. Avaliação Heurística

A aplicação do método de avaliação heurística nesse aplicativo resultou na seguinte lista de problemas:

- Falta de *feedback* na execução de comandos
  - Descrição: Na tela de execução de consultas, quando comandos são executados (e.g. inserção de dados, adição de colunas em tabelas) com sucesso, nenhum *feedback* é apresentado ao usuário.
  - Heurística violada: Visibilidade do *status* do sistema.
- Falta de *feedback* na inserção de dados
  - Descrição: Quando registros são inseridos com sucesso em uma tabela por meio do botão *Insert new Row*, nenhum *feedback* é apresentado ao usuário.
  - Heurística violada: Visibilidade do *status* do sistema.
- *Feedback* inapropriado na execução de comandos
  - Descrição: Na tela de execução de consultas, quando alguns comandos são executados (e.g. comandos do tipo *select into*) uma mensagem de erro é apresentada, porém o comando é executado corretamente (Figura 4.11).
  - Heurística violada: Visibilidade do *status* do sistema.

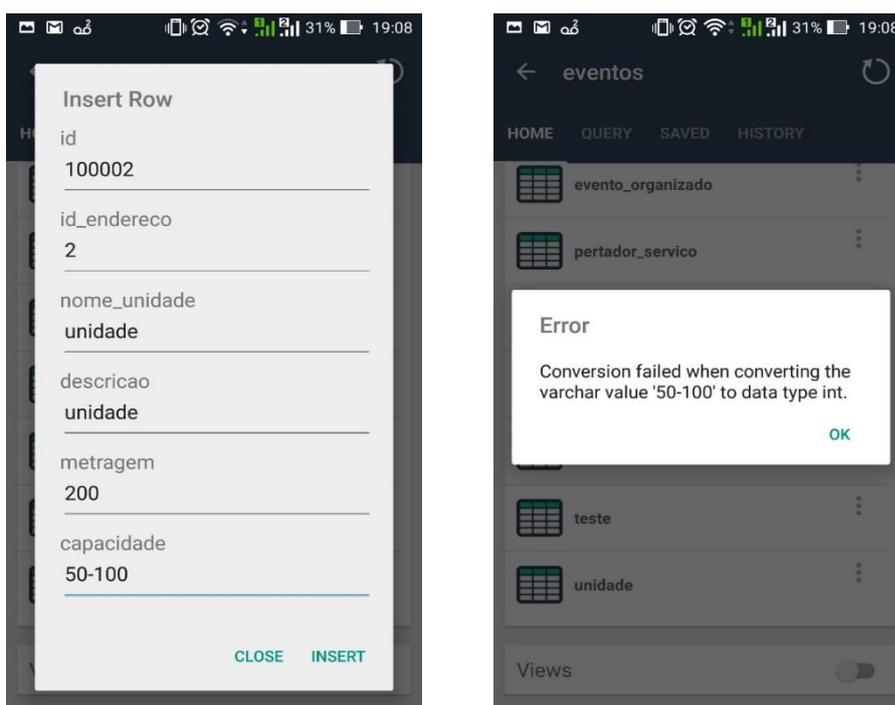
Figura 4.11 - Mensagem de erro apresentada na execução de um comando *select into*.



Fonte: RemoDB (2016).

- Falta de informação e auxílio na inserção de registros
  - Descrição: No formulário de inserção de dados não há nenhuma informação sobre os tipos dos campos da tabela ou limitação do teclado com base no tipo do campo. Sem essa informação os usuários podem achar que um campo código, por exemplo, é do tipo *VARCHAR* quando na verdade é do tipo *INT*, resultando em um erro que poderia ser evitado (Figura 4.12).
  - Heurística violada: Prevenção de erros.

Figura 4.12 - Erro apresentado ao tentar cadastrar um tipo *VARCHAR* em um campo *INT*.



Fonte: RemoDB (2016).

### 4.2.5.3. Percurso Cognitivo

Entre os cenários criados, apenas o de realização de consultas (1) e o de inserção de dados (2) puderam ser avaliados nessa aplicação. O cenário de criação de tabela (3) tem como base uma tarefa não contemplada por esse aplicativo.

Segue o resultado da aplicação do método de percurso cognitivo nesse aplicativo:

- Realização de consultas
  - Clicar no botão '+':

- **Os usuários saberão qual a ação correta para atingir o resultado desejado?**

Não. Não existe nenhum título ou informação que identifique a tela inicial como a tela de criação e seleção de conexões e o botão '+' por si só não deixa claro a sua função.

- **Os usuários perceberão que a ação correta está disponível?**

Sim. Na tela inicial o botão '+' está bem visível.

- **Os usuários irão associar a ação correta com o resultado desejado?**

Não. Apesar do botão '+' ser associado pelo usuário à criação de novos objetos, o usuário não sabe do que se trata essa tela, logo não sabe o que o botão '+' vai criar ou adicionar.

- **Se a ação correta for executada os usuários perceberão que foi feito avanço em relação à tarefa desejada?**

Sim. Após clicar no botão '+' o aplicativo vai para a tela de cadastro de conexão, deixando claro que o usuário precisa cadastrar o servidor.

- Entrar com as informações de conexão:

- **Os usuários saberão qual a ação correta para atingir o resultado desejado?**

Sim. Na tela de cadastro de conexão existe um formulário solicitando os dados necessários para a conexão com um banco de dados, deixando claro para o usuário que ele precisa conectar ao banco antes de realizar consultas.

- **Os usuários perceberão que a ação correta está disponível?**

Sim. Na tela de cadastro de conexão o formulário para o cadastro de conexões está bem visível.

- **Os usuários irão associar a ação correta com o resultado desejado?**

Sim. Os campos do formulário são informações necessárias para a conexão com um banco de dados, deixando claro para o usuário que é necessário preenche-los para cadastrar a conexão.

- **Se a ação correta for executada os usuários perceberão que foi feito avanço em relação à tarefa desejada?**

Sim. Após preencher os campos o usuário sabe que basta salvar a conexão.

- Salvar a conexão:

- **Os usuários saberão qual a ação correta para atingir o resultado desejado?**

Sim. Na tela de cadastro de conexão existe botão com a palavra *DONE* abaixo do formulário, deixando claro que depois de preencher os campos basta clicar no botão para realizar o cadastro da conexão.

- **Os usuários perceberão que a ação correta está disponível?**

Sim. O botão *DONE* está bem visível na tela de cadastro de conexão.

- **Os usuários irão associar a ação correta com o resultado desejado?**

Sim. O botão *DONE* deixa claro que após inserir as informações sobre a conexão no formulário ele deve ser pressionado para cadastrar a conexão.

- **Se a ação correta for executada os usuários perceberão que foi feito avanço em relação à tarefa desejada?**

Sim. Após apertar no botão *DONE* o usuário verá a conexão recém-criada na lista de conexões.

- Selecionar a conexão criada:

- **Os usuários saberão qual a ação correta para atingir o resultado desejado?**

Sim. Após criar a conexão os usuários saberão por experiência que precisam usá-la para se conectar ao banco.

- **Os usuários perceberão que a ação correta está disponível?**

Sim. As conexões criadas vão estar listadas na tela, e, devido à experiência com outros aplicativos *mobile*, o usuário vai saber que pode clicar nela.

- **Os usuários irão associar a ação correta com o resultado desejado?**

Sim. As conexões listadas possuem o nome da conexão cadastrado quando a conexão é criada e, devido à experiência com outros aplicativos *mobile*, vão entender que se clicarem nela irão se conectar ao banco.

- **Se a ação correta for executada os usuários perceberão que foi feito avanço em relação à tarefa desejada?**

Sim. O aplicativo irá para a tela principal com o nome da conexão no título, deixando claro ao usuário que a conexão foi realizada.

- Ir para a tela de escrita de consulta:

- **Os usuários saberão qual a ação correta para atingir o resultado desejado?**

Sim. Após se conectar o usuário vai estar na tela principal do aplicativo que possui várias abas, sendo uma delas a aba *QUERY*. Fica evidente para o usuário que a próxima ação é ir para essa aba.

- **Os usuários perceberão que a ação correta está disponível?**

Sim. Na tela principal, a aba *QUERY* está bem visível.

- **Os usuários irão associar a ação correta com o resultado desejado?**

Sim. O usuário vai ver a aba *QUERY* e vai entender que deve clicar para ir para a tela de escrita de consultas.

- **Se a ação correta for executada os usuários perceberão que foi feito avanço em relação à tarefa desejada?**

Sim. O aplicativo irá para uma tela com espaço para digitar com o nome *QUERY* na aba selecionada.

- Escrever consulta:

- **Os usuários saberão qual a ação correta para atingir o resultado desejado?**

Sim. O usuário tem conhecimento básico sobre a escrita de consultas.

- **Os usuários perceberão que a ação correta está disponível?**

Sim. Existe um campo para digitar consultas e o teclado para permitir a digitação.

- **Os usuários irão associar a ação correta com o resultado desejado?**

Sim. O usuário saberá que deve digitar sua consulta por causa do teclado e por causa do nome *QUERY* no título da tela.

- **Se a ação correta for executada os usuários perceberão que foi feito avanço em relação à tarefa desejada?**

Sim. Após escrever a consulta o usuário sabe que basta executá-la para ver o resultado.

- Executar consulta:

- **Os usuários saberão qual a ação correta para atingir o resultado desejado?**

Sim. Após escrever a consulta o usuário sabe que basta executá-la para ver o resultado.

- **Os usuários perceberão que a ação correta está disponível?**

Sim. O botão *GO* está bem visível na tela de escrita de consultas.

- **Os usuários irão associar a ação correta com o resultado desejado?**

Sim. O usuário vai entender o botão *GO* é utilizado para executar a consulta digitada.

- **Se a ação correta for executada os usuários perceberão que foi feito avanço em relação à tarefa desejada?**

Sim. Após a realização da consulta o aplicativo irá apresentar os dados solicitados na consulta em formato tabular.

- Inserção de dados

- Selecionar a conexão:

- **Os usuários saberão qual a ação correta para atingir o resultado desejado?**

Sim. Após criar a conexão os usuários saberão por experiência que precisam usá-la para se conectar ao banco.

- **Os usuários perceberão que a ação correta está disponível?**

Sim. As conexões criadas vão estar listadas na tela, e, devido à experiência com outros aplicativos *mobile*, o usuário vai saber que pode clicar nela.

- **Os usuários irão associar a ação correta com o resultado desejado?**

Sim. As conexões listadas possuem o nome da conexão cadastrado quando a conexão é criada e, devido à experiência com outros aplicativos *mobile*, vão entender que se clicarem nela irão se conectar ao banco.

- **Se a ação correta for executada os usuários perceberão que foi feito avanço em relação à tarefa desejada?**

Sim. O aplicativo irá para a tela principal com o nome da conexão no título, deixando claro ao usuário que a conexão foi realizada.

- Habilitar a listagem de tabelas:

- **Os usuários saberão qual a ação correta para atingir o resultado desejado?**

Sim. Na tela principal existe uma opção *Tables* com um botão tipicamente utilizado para habilitar ou mostrar informações.

- **Os usuários perceberão que a ação correta está disponível?**

Sim. O botão de habilitar está bem visível.

- **Os usuários irão associar a ação correta com o resultado desejado?**

Sim. O tipo do botão usado para mostrar as tabelas é tipicamente usado para habilitar ou mostrar informações.

- **Se a ação correta for executada os usuários perceberão que foi feito avanço em relação à tarefa desejada?**

Sim. Após clicar no botão, as tabelas do banco vão ser listadas na tela.

- Ir para a tela de inserção de dados:

- **Os usuários saberão qual a ação correta para atingir o resultado desejado?**

Sim. Após ver as tabelas listadas o usuário vai saber que precisa escolher uma tabela e ir para a tela de inserção de dados.

- **Os usuários perceberão que a ação correta está disponível?**

Sim. O botão *Insert new Row* está visível dentro do menu de opções de cada tabela. O menu de opções está bem visível.

- **Os usuários irão associar a ação correta com o resultado desejado?**

Sim. O botão *Insert new Row* diz exatamente o que ele faz.

- **Se a ação correta for executada os usuários perceberão que foi feito avanço em relação à tarefa desejada?**

Sim. Após clicar no botão *Insert new Row*, um formulário com os campos da tabela irá aparecer na tela.

- Preencher formulário:

- **Os usuários saberão qual a ação correta para atingir o resultado desejado?**

Sim. O formulário com as colunas da tabela deixa claro para o usuário que ele tem que informar os valores dos campos para o novo registro.

- **Os usuários perceberão que a ação correta está disponível?**

Sim. O formulário para a inserção de dados está bem visível na tela.

- **Os usuários irão associar a ação correta com o resultado desejado?**

Sim. O formulário com as colunas da tabela deixa claro para o usuário que ele tem que informar os valores dos campos para inserir o novo registro.

- **Se a ação correta for executada os usuários perceberão que foi feito avanço em relação à tarefa desejada?**

Sim. Após preencher o formulário, o usuário vai saber que basta clicar no botão *INSERT* para inserir o registro.

- Inserir dados:
  - **Os usuários saberão qual a ação correta para atingir o resultado desejado?**

Sim. Após preencher o formulário, o usuário sabe que o próximo passo é inserir o registro.
  - **Os usuários perceberão que a ação correta está disponível?**

Sim. O botão *INSERT* está bem visível na tela.
  - **Os usuários irão associar a ação correta com o resultado desejado?**

Sim. O botão *INSERT* deixa claro para o usuário que sua função é inserir os dados preenchidos no formulário.
  - **Se a ação correta for executada os usuários perceberão que foi feito avanço em relação à tarefa desejada?**

Não. Após clicar no botão *INSERT* nenhuma mensagem de confirmação de inserção é apresentada caso o registro tenha sido inserido com sucesso.

### 4.3. Considerações Finais

Após realizar a seleção e análise dos aplicativos existentes no mercado, fica evidente a indisponibilidade de ferramentas minimamente completas e os vários problemas encontrados pelos usuários ao tentar utilizar esse tipo de ferramenta.

Todas as aplicações encontradas possuíam apenas parte das funções consideradas básicas para a visualização, manipulação e gerenciamento de dados oriundos de bancos de dados, sendo as funções mais encontradas a execução de consultas, execução de comandos, visualização de objetos de banco e visualização de dados. As funções referentes à alteração de dados e à alteração de estrutura não foram encontradas de forma completa em nenhum dos aplicativos analisados. Entre os aplicativos encontrados durante a seleção, apenas o aplicativo RemoDB dá suporte à inserção e à atualização de dados (6.1 e 6.2), e o aplicativo Mobile SQL Server Manager à criação e à remoção de tabelas (5.1 e 5.2). As funções de remoção de dados (6.3), adição de colunas (5.3) e remoção de colunas (5.4) não podem ser realizadas via interface por nenhum dos aplicativos encontrados.

No quesito usabilidade, as aplicações selecionadas também deixam a desejar, possuindo interfaces que apresentaram vários problemas de usabilidade durante os testes. Um dos

problemas mais encontrados são relacionados ao mal, ou nenhum uso de *feedback*, seja para informar erro ou sucesso de uma operação. Outros problemas encontrados foram a má visibilidade de botões e opções frequentemente utilizados, excesso de informações e opções irrelevantes ou raramente necessárias apresentadas na mesma tela, falta de sinalização sobre a função de opções, botões e telas, falta de informação sobre o *status* do sistema e falta de controle do usuário.

Além dos problemas de usabilidade encontrados, outros pontos que dificultam o uso desses aplicativos são aspectos relacionados a limitações de dispositivos móveis, principalmente formato de entrada de dados e dimensão. A maioria das aplicações analisadas possui uma apresentação de dados em formato tabular descente, porém no aplicativo Mobile SQL Client, por exemplo, a visualização de dados em formato tabular é muito difícil, sendo necessário para ver informações completas clicar na linha e ver os dados de maneira diferenciada. A visualização de informações em formato tabular é essencial, sendo necessário contornar as limitações apresentadas pelo tamanho da tela por meio de ajustes no tamanho das colunas de acordo com o maior valor no campo, possibilidade de dar *zoom* e percorrer a tabela com gestos simples.

A entrada de dados em dispositivos móveis é outra limitação que deve ser contornada para tornar mais usual a escrita de consultas, que é uma das funções mais importantes para aplicativos desse tipo. Entre os aplicativos analisados, apenas o Mobile SQL Client possui uma forma de auxílio na escrita de consultas. Esse auxílio é fornecido por um dicionário de palavras chave que pode ser acessado e as palavras chave inseridas no campo de consulta. Apesar dessa função ser fornecida, o usuário pode apresentar problemas no seu uso, pois é necessário percorrer a lista procurando uma das palavras, essa palavra é sempre inserida no final do texto em vez de onde o cursor está atualmente, e o cursor volta para o início do campo de texto. O máximo de suporte que algumas das outras aplicações dá ao usuário na escrita de consulta é o *highlight* nas palavras chave.

O Quadro 4.1 apresenta um resumo do resultado de todas as avaliações realizadas para cada aplicativo.

Quadro 4.1 - Resumo das avaliações realizadas no capítulo.

	Funções não contempladas	Problemas de usabilidade				Limitações não contornadas
		Avaliação heurística	Percurso cognitivo			
			Consultas	Inserção de dados	Criação de tabela	
SQL Client	12	7	4	-	-	1
Mobile SQL Client	10	5	0	-	-	2
SQL Studio Lite	8	5	5	-	-	1
Mobile SQL Server Manager	7	6	6	-	2	1
RemoDB	9	5	2	1	-	1

Fonte: O Autor (2016).

No próximo capítulo serão apresentadas possíveis soluções para os problemas e limitações encontrados nas aplicações analisadas e uma aplicação que põe em prática essas soluções.

## 5. Solução Desenvolvida: SQL Manager

Este capítulo irá propor soluções para os problemas encontrados na análise realizada nos aplicativos selecionados, sejam eles problemas de usabilidade, limitações inerentes aos dispositivos móveis ou problemas nas funcionalidades apresentadas. Será apresentada também uma aplicação para a plataforma Android baseada nas soluções propostas, como prova de conceito. Os detalhes da aplicação e as ferramentas utilizadas no seu desenvolvimento serão apresentados na próxima seção.

### 5.1. Detalhes do Desenvolvimento

A aplicação desenvolvida para pôr em prática os resultados das análises e melhorias propostas foi chamada de SQL Manager. Esta foi desenvolvida para a plataforma Android, mesma plataforma das aplicações selecionadas, e dá suporte a sistemas de banco de dados Microsoft SQL Server, para o seu uso ser avaliado sobre o mesmo SGBD que as outras aplicações foram.

Foi utilizada no desenvolvimento a IDE Android Studio<sup>31</sup>, a qual atualmente é a IDE oficial para desenvolvimento Android, substituindo a IDE Eclipse<sup>32</sup> com o Eclipse ADT Plugin<sup>33</sup>, que não tem mais suporte.

Para a realização de testes da aplicação foi criado um banco de dados no Microsoft SQL Server por meio da ferramenta de gerenciamento SQL Server Management Studio<sup>34</sup>. As mensagens de erro da aplicação desenvolvida foram baseadas nas mensagens de erro dessa ferramenta, para facilitar o entendimento de usuários familiarizados com ela.

---

<sup>31</sup> <https://developer.android.com/studio/index.html>

<sup>32</sup> <https://eclipse.org/>

<sup>33</sup> <https://developer.android.com/studio/tools/sdk/eclipse-adt.html>

<sup>34</sup> <https://msdn.microsoft.com/pt-BR/library/hh213248.aspx>

## 5.2. Melhorias Implementadas

A partir das análises realizadas no capítulo anterior, pode-se perceber que os problemas mais críticos encontrados nas aplicações analisadas são: falta de funções importantes disponíveis via interface, má visibilidade de botões e opções frequentemente utilizados, excesso de informações e opções irrelevantes ou raramente necessárias, falta de sinalização sobre a função de opções, botões e telas, *feedback* inapropriado ou inexistente, falta de sinalização do *status* do sistema e problemas referentes à escrita de consultas e à visualização de informação em formato tabular. Serão apresentados no decorrer dessa seção aspectos que foram melhorados comparando com os aplicativos analisados, enquanto são apresentadas as telas do SQL Manager.

A identificação dos componentes da interface, como tela, botões e opções disponíveis são essenciais para o bom entendimento do usuário sobre as funções disponíveis e para que o sistema seja de fácil aprendizado. Os aplicativos SQL Studio Lite e o Mobile SQL Server Manager não identificavam as telas, fazendo com que fosse necessário que o usuário subentendesse qual o objetivo da tela de acordo com os objetos apresentados e como a tela foi acessada. No SQL Manager todas as telas foram sinalizadas de acordo com sua função. No título da tela fica visível o nome da conexão, banco de dados ou tabela de origem das informações, e como subtítulo é apresentada a descrição da tela, como pode ser visto na Figura 5.1.

Os botões do SQL Manager também foram bem identificados, sempre deixando claro sua função para o público alvo, seja com palavras ou símbolos conhecidos por usuários de aplicações móveis e de ferramentas de banco de dados em ambiente *desktop*. Um exemplo disso é o botão '+', que pode ser encontrado em diversas telas do aplicativo, com o qual é possível criar novos objetos (Figura 5.1). Apesar desse botão não dizer exatamente o que ele faz, sua função é conhecida por usuários de aplicações Android por ser um botão recorrente em aplicativos. Esse conhecimento associado à descrição da tela deixa clara sua função a usuários com conhecimento no domínio.

Outro aspecto melhorado foi a visibilidade das opções importantes disponíveis. Alguns aplicativos, como o SQL Studio Lite e o Mobile SQL Server Manager, colocavam funções importantes acessíveis apenas por meio do menu da tela ou do menu de objetos da tela. Um exemplo disso é o acesso à tela de escrita de consultas. No SQL Manager essa função fica disponível no lado direito superior das telas em que essa função pode ser acessada (Figura 5.1).

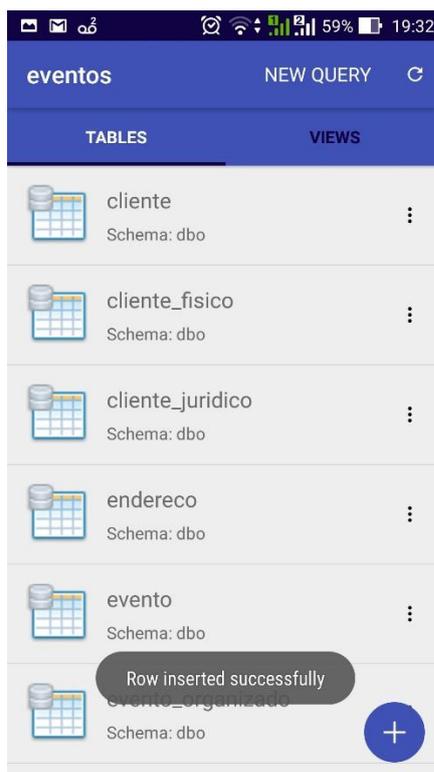
Figura 5.1 - Tela de seleção de bancos de dados existentes na conexão.



Fonte: O Autor (2016).

A visibilidade do status do sistema também é muito importante. Por exemplo, o usuário precisa ser informado quando a aplicação está executando uma tarefa na qual é necessário aguardar a sua finalização, como o carregamento de informações externas (e.g. lista de bancos de dados, lista de tabelas) ou a execução de consultas e comandos. O SQL Manager apresenta uma tela informando que uma operação está sendo realizada sempre que necessário, mantendo o usuário ciente do que está acontecendo.

Dar um *feedback* apropriado ao usuário também é necessário em diversas situações, seja para informar sobre erros, sucesso ou simplesmente sobre o *status* do sistema. Nos aplicativos analisados, em várias situações, os *feedbacks* deixaram a desejar ou simplesmente não existiram. Por exemplo, no RemoDB após a inserção de dados com sucesso nenhum *feedback* é apresentado para confirmar a inserção do registro. No aplicativo SQL Client, quando comandos de alteração de estrutura ou de dados são executados a mensagem retornada é semelhante a uma mensagem de sucesso, mas esse tipo de comando não funciona no aplicativo. O SQL Manager apresenta *feedbacks* claros em todas as situações que são necessárias, inclusive nessas três tarefas essenciais desse tipo de sistema. Na Figura 5.2 pode-se ver o *feedback* apresentado quando linhas são inseridas.

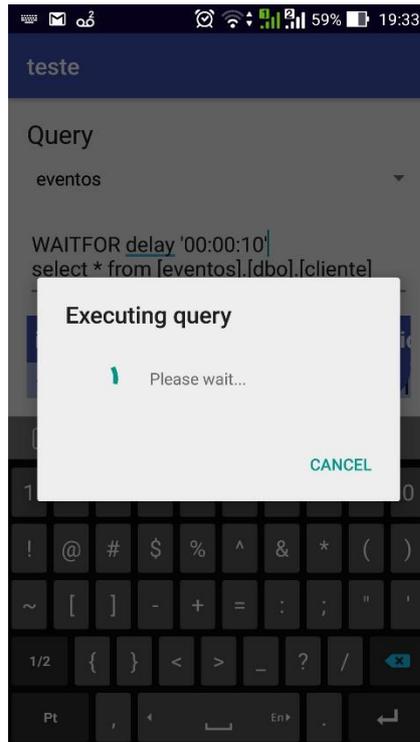
Figura 5.2 - *Feedback* apresentado após a inserção de dados com sucesso.

Fonte: O Autor (2016).

Dentre os aplicativos analisados, apenas o Mobile SQL Client possui a opção de parar uma consulta que foi iniciada. É essencial a qualquer tipo de sistema fornecer ao usuário controle, permitindo cancelar operações que foram solicitadas. É muito comum em escrita de consultas SQL que o usuário perceba que existe algum erro na consulta depois que já está executando ou que a consulta esteja demorando mais do que o esperado, por isso o usuário deve ter a possibilidade de cancelar a operação. Como pode ser visto na Figura 5.3, o SQL Manager fornece essa opção.

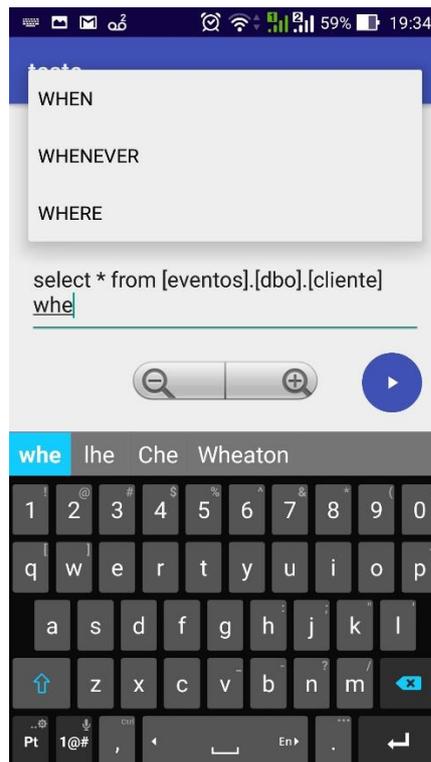
A escrita e execução de consultas é uma das principais funções desses aplicativos, e também é um dos pontos que mais sofre com as limitações dos dispositivos móveis, por isso essa função deve ser facilitada. O único aplicativo selecionado que possui alguma forma de auxílio na escrita de consultas é o Mobile SQL Client, no entanto, como mencionado no capítulo anterior, essa opção não é muito efetiva. O SQL Manager apresenta um meio mais efetivo de auxílio na escrita de consultas, apresentando uma opção de completar automaticamente. De acordo com o que está sendo digitado o sistema apresenta opções de palavras reservadas ao usuário, facilitando a escrita de consultas (Figura 5.4).

Figura 5.3 - Execução de consulta com a opção de cancelamento.



Fonte: O Autor (2016).

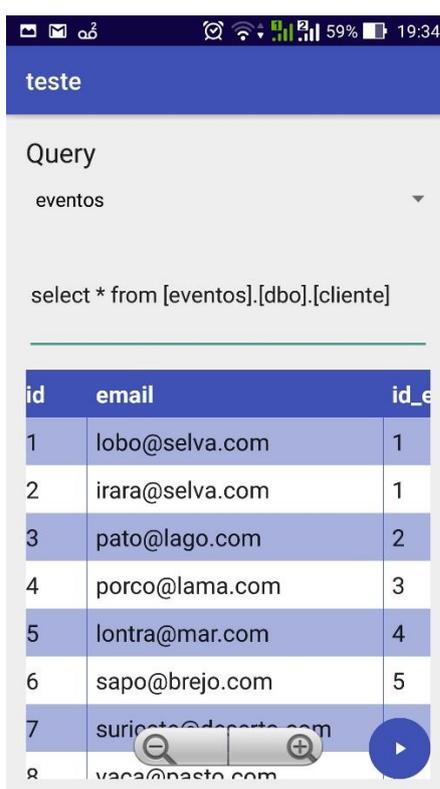
Figura 5.4 - Escrita de consultas com a opção de auto completar.



Fonte: O Autor (2016).

A visualização de dados em formato tabular também é afetada pelas restrições dos dispositivos móveis. O aplicativo Mobile SQL Client não utiliza bem o espaço da tela, e como consequência não permite ao usuário ver todas as informações na tabela, sendo necessário muitas vezes clicar na linha para ver as informações em outro formato. O SQL Manager fornece formas de facilitar a visualização da informação, permitindo por meio de gestos simples percorrer a tabela e dar *zoom*, sempre apresentando todas as informações. Pode ser visto na Figura 5.5 os botões de *zoom* na tela de visualização de dados.

Figura 5.5 - Botão de *zoom* disponível na tela de visualização de dados.



Fonte: O Autor (2016).

O SQL Manager realizou melhoria também na tarefa de inserção e modificação de dados. Com exceção do Remo DB, nenhum aplicativo apresenta a possibilidade de inserir e modificar informações das tabelas via interface. No RemoDB isso é possível, mas nenhuma informação sobre os tipos das colunas é fornecida ao usuário, o que pode levar a erros. No SQL Manager nas telas de inserção e atualização de dados, além de informar o nome da coluna e o tipo, o método de entrada de dados muda de acordo com o tipo, facilitando a inserção ou atualização de dados e evitando erros. Na Figura 5.6 pode-se observar a inserção em campos

do tipo *date*. Além disso, não é permitida entrada de dados nas colunas que possuem características que não permitem a inserção de informação, como colunas calculadas.

Figura 5.6 - Cadastro de colunas do tipo *date*.



Fonte: O Autor (2016).

A criação de tabelas é outra tarefa que apenas um dos aplicativos selecionados dá suporte via interface. O SQL Manager dá suporte a essa tarefa e realizou melhorias nela em comparação com o Mobile SQL Server Manager. Uma das melhorias foi a adição de mais opções na criação de colunas, permitindo criar colunas calculada, *primary key*, *unique key*, *identity* e *nullable*. A outra melhoria realizada foi a prevenção de erros, impedindo o usuário de tentar criar uma coluna que tenha propriedades exclusivas juntas, por exemplo, *primary key* e *nullable*. Quando a propriedade *nullable* é selecionada a opção *primary key* é desabilitada.

Apesar de ser uma das principais tarefas desse tipo de ferramenta, nenhuma das aplicações analisadas dá suporte a modificação de tabelas. O SQL Manager permite a modificação de tabelas via interface, possibilitando a remoção e criação de colunas em tabelas existentes.

### 5.3. Checklist de Funções

Dentre as funções consideradas importantes para uma ferramenta de acesso a bancos de dados remotos o SQL Manager dá suporte a: realização de consultas (1), realização de comandos (2), visualização de dados em formato tabular (3), visualização de tabelas (4.1), visualização de *views* (4.2), criação de tabelas (5.1), remoção de tabelas (5.2), adição de colunas (5.3), remoção de colunas (5.4), inserção de dados (6.1), atualização de dados (6.2) e remoção de dados (6.3).

Apenas a visualização via interface de *procedures* (4.3), *functions* (4.4) e *triggers* (4.5) não foi implementada. Foi dada prioridade à conclusão das principais tarefas realizadas em um banco de dados via interface, de maneira a facilitar o uso da ferramenta, evitar a escrita de consultas para realizar tarefas que podem ser feitas via interface, otimizar a escrita de consultas e otimizar a visualização de dados em formato tabular.

### 5.4. Avaliação Heurística

Como foi realizada uma análise nos aplicativos selecionados e o SQL Manager foi desenvolvido para resolver os problemas encontrados focando na usabilidade, muitos dos problemas de usabilidade foram resolvidos no próprio desenvolvimento. No entanto, avaliando o aplicativo desenvolvido ainda foi encontrado um problema de usabilidade por meio da técnica de avaliação heurística. O problema está descrito abaixo:

- Botão de opções do objeto da lista oculto pelo botão de adição de objetos
  - Descrição: Quando a lista de objetos (e.g. conexões, bancos de dados, tabelas) está grande a ponto de precisar de *scroll* o último elemento fica com o botão de menu atrás do botão de adição de objeto. O menu do item ainda pode ser acessado a partir da opção segurar e clicar.
  - Heurística violada: Reconhecimento ao invés de lembrança.

## 5.5. Percurso Cognitivo

Segue o resultado da aplicação do método de percurso cognitivo no SQL Manager:

- Realização de consultas
  - Clicar no botão '+':
    - **Os usuários saberão qual a ação correta para atingir o resultado desejado?**  
Sim. O título *Connection List* associado com o botão '+' deixa claro que é necessário criar uma nova conexão para acessar o banco de dados.
    - **Os usuários perceberão que a ação correta está disponível?**  
Sim. Na tela de conexões o botão '+' está bem visível.
    - **Os usuários irão associar a ação correta com o resultado desejado?**  
Sim. O usuário sabe que está na tela com de conexões e sabe, devido à experiência em aplicativos *mobile*, que o símbolo '+' é normalmente associado a criação de um novo objeto.
    - **Se a ação correta for executada os usuários perceberão que foi feito avanço em relação à tarefa desejada?**  
Sim. Após clicar no botão '+' o aplicativo vai para a tela de cadastro de conexão, deixando claro que o usuário precisa cadastrar o servidor.
  - Entrar com as informações de conexão:
    - **Os usuários saberão qual a ação correta para atingir o resultado desejado?**  
Sim. Na tela de cadastro de conexão existe um formulário solicitando os dados necessários para a conexão com um banco de dados, deixando claro ao usuário que ele precisa conectar ao banco antes de realizar consultas.
    - **Os usuários perceberão que a ação correta está disponível?**  
Sim. Na tela de cadastro de conexão o formulário para o cadastro de conexões está bem visível.
    - **Os usuários irão associar a ação correta com o resultado desejado?**  
Sim. Os campos do formulário são informações necessárias para a conexão com um banco de dados, deixando claro para o usuário que é necessário preenche-los para cadastrar a conexão.

- **Se a ação correta for executada os usuários perceberão que foi feito avanço em relação à tarefa desejada?**

Sim. Após preencher os campos o usuário sabe que basta salvar a conexão.
- Salvar a conexão:
  - **Os usuários saberão qual a ação correta para atingir o resultado desejado?**

Sim. Na tela de cadastro de conexão existe botão com a palavra *save* abaixo do formulário, deixando claro que depois de preencher os campos basta clicar no botão para salvar a conexão.
  - **Os usuários perceberão que a ação correta está disponível?**

Sim. O botão *save* está bem visível na tela de cadastro de conexão.
  - **Os usuários irão associar a ação correta com o resultado desejado?**

Sim. O botão *save* deixa claro que após inserir as informações sobre a conexão no formulário ele deve ser pressionado para salvar a conexão.
  - **Se a ação correta for executada os usuários perceberão que foi feito avanço em relação à tarefa desejada?**

Sim. Após apertar no botão *save* o usuário verá a conexão recém-criada na lista de conexões.
- Selecionar a conexão criada:
  - **Os usuários saberão qual a ação correta para atingir o resultado desejado?**

Sim. Após criar a conexão os usuários saberão por experiência que precisam usá-la para se conectar ao banco.
  - **Os usuários perceberão que a ação correta está disponível?**

Sim. As conexões criadas vão estar listadas na tela, e, devido à experiência com outros aplicativos *mobile*, o usuário vai saber que pode clicar nela.
  - **Os usuários irão associar a ação correta com o resultado desejado?**

Sim. As conexões listadas possuem o nome da conexão cadastrado quando a conexão é criada e, devido à experiência com outros aplicativos *mobile*, vão entender que se clicarem nela irão se conectar ao banco.
  - **Se a ação correta for executada os usuários perceberão que foi feito avanço em relação à tarefa desejada?**

Sim. O aplicativo irá para outra tela com a lista de bancos de dados existentes, deixando claro ao usuário que a conexão foi realizada.

- Ir para a tela de escrita de consulta:
  - **Os usuários saberão qual a ação correta para atingir o resultado desejado?**  
Sim. Após se conectar, o usuário vai ver a opção *NEW QUERY* bem visível no canto superior direito da tela.
  - **Os usuários perceberão que a ação correta está disponível?**  
Sim. Após se conectar, o usuário vai ver a opção *NEW QUERY* bem visível no canto superior direito da tela.
  - **Os usuários irão associar a ação correta com o resultado desejado?**  
Sim. O botão *NEW QUERY* deixa claro sua função.
  - **Se a ação correta for executada os usuários perceberão que foi feito avanço em relação à tarefa desejada?**  
Sim. O aplicativo irá para uma tela com espaço para digitar com o nome *Query* no título.
- Escrever consulta:
  - **Os usuários saberão qual a ação correta para atingir o resultado desejado?**  
Sim. O usuário tem conhecimento básico sobre a escrita de consultas.
  - **Os usuários perceberão que a ação correta está disponível?**  
Sim. Existe um campo para digitar consultas e o teclado para a digitação.
  - **Os usuários irão associar a ação correta com o resultado desejado?**  
Sim. O usuário saberá que deve digitar sua consulta por causa do teclado e por causa do nome *SQL Query* no título da tela.
  - **Se a ação correta for executada os usuários perceberão que foi feito avanço em relação à tarefa desejada?**  
Sim. Após escrever a consulta o usuário sabe que basta executá-la para ver o resultado.
- Executar consulta:
  - **Os usuários saberão qual a ação correta para atingir o resultado desejado?**  
Sim. Após escrever a consulta o usuário sabe que basta executá-la para ver o resultado por experiência.

- **Os usuários perceberão que a ação correta está disponível?**  
Sim. O botão de *play* está bem visível.
- **Os usuários irão associar a ação correta com o resultado desejado?**  
Sim. O botão de *play* deixa claro sua função.
- **Se a ação correta for executada os usuários perceberão que foi feito avanço em relação à tarefa desejada?**  
Sim. Após a realização da consulta o aplicativo irá apresentar os dados solicitados na consulta em formato tabular.
- Inserção de dados
  - Selecionar a conexão:
    - **Os usuários saberão qual a ação correta para atingir o resultado desejado?**  
Sim. O título *Connection List* e a lista com o nome das conexões vão deixar claro que o usuário precisa escolher uma conexão.
    - **Os usuários perceberão que a ação correta está disponível?**  
Sim. As conexões criadas vão estar listadas na tela, e, devido à experiência com outros aplicativos *mobile*, o usuário vai saber que pode clicar nela.
    - **Os usuários irão associar a ação correta com o resultado desejado?**  
Sim. As conexões listadas possuem o nome da conexão cadastrado quando a conexão é criada e, devido à experiência com outros aplicativos *mobile*, vão entender que se clicarem nela irão se conectar ao banco.
    - **Se a ação correta for executada os usuários perceberão que foi feito avanço em relação à tarefa desejada?**  
Sim. O aplicativo irá para a tela de seleção de bancos com o nome da conexão no título, deixando claro ao usuário que a conexão foi realizada.
  - Selecionar o banco de dados:
    - **Os usuários saberão qual a ação correta para atingir o resultado desejado?**  
Sim. O título *Database List* informa que a lista se trata dos bancos existentes no servidor e o usuário saberá que tem que escolher uma base.
    - **Os usuários perceberão que a ação correta está disponível?**

Sim. Os bancos de dados existentes vão estar listados na tela, e, devido à experiência com outros aplicativos *mobile*, o usuário vai saber que pode clicar nele.

▪ **Os usuários irão associar a ação correta com o resultado desejado?**

Sim. Os bancos de dados existentes vão estar listados na tela, e, devido à experiência com outros aplicativos *mobile*, vão entender que se clicarem nele irão se conectar ao banco.

▪ **Se a ação correta for executada os usuários perceberão que foi feito avanço em relação à tarefa desejada?**

Sim. Após clicar no banco de dados, o aplicativo irá para a tela principal com as tabelas existentes listadas.

○ Ir para a tela de inserção de dados:

▪ **Os usuários saberão qual a ação correta para atingir o resultado desejado?**

Sim. Após ver as tabelas listadas o usuário vai saber que precisa escolher uma tabela e ir para a tela de inserção de dados.

▪ **Os usuários perceberão que a ação correta está disponível?**

Sim. O botão *Insert Row* está visível dentro do menu de opções de cada tabela. O menu de opções está bem visível.

▪ **Os usuários irão associar a ação correta com o resultado desejado?**

Sim. O botão *Insert Row* diz exatamente o que ele faz.

▪ **Se a ação correta for executada os usuários perceberão que foi feito avanço em relação à tarefa desejada?**

Sim. Após clicar no botão *Insert Row*, um formulário com os campos da tabela irá aparecer na tela.

○ Preencher formulário:

▪ **Os usuários saberão qual a ação correta para atingir o resultado desejado?**

Sim. O formulário com as colunas da tabela deixa claro para o usuário que ele tem que informar os valores dos campos para o novo registro.

▪ **Os usuários perceberão que a ação correta está disponível?**

Sim. O formulário para a inserção de dados está bem visível na tela.

▪ **Os usuários irão associar a ação correta com o resultado desejado?**

Sim. O formulário com as colunas da tabela deixa claro para o usuário que ele tem que informar os valores dos campos para inserir o novo registro.

- **Se a ação correta for executada os usuários perceberão que foi feito avanço em relação à tarefa desejada?**

Sim. Após preencher o formulário, o usuário vai saber que basta clicar no botão *INSERT* para inserir o registro.

- Inserir dados:

- **Os usuários saberão qual a ação correta para atingir o resultado desejado?**

Sim. Após preencher o formulário, o usuário sabe que o próximo passo é inserir o registro.

- **Os usuários perceberão que a ação correta está disponível?**

Sim. O botão *INSERT* está bem visível na tela.

- **Os usuários irão associar a ação correta com o resultado desejado?**

Sim. O botão *INSERT* deixa claro para o usuário que sua função é inserir os dados preenchidos no formulário.

- **Se a ação correta for executada os usuários perceberão que foi feito avanço em relação à tarefa desejada?**

Sim. Após clicar no botão *INSERT*, caso o cadastro tenha sido realizado com sucesso uma mensagem irá confirmar a inserção do registro.

- Criação de tabela

- Selecionar a conexão:

- **Os usuários saberão qual a ação correta para atingir o resultado desejado?**

Sim. O título *Connection List* e a lista com o nome das conexões vão deixar claro que o usuário precisa escolher uma conexão.

- **Os usuários perceberão que a ação correta está disponível?**

Sim. As conexões criadas vão estar listadas na tela, e, devido à experiência com outros aplicativos *mobile*, o usuário vai saber que pode clicar nela.

- **Os usuários irão associar a ação correta com o resultado desejado?**

Sim. As conexões listadas possuem o nome da conexão cadastrado quando a conexão é criada e, devido à experiência com outros aplicativos *mobile*, vão entender que se clicarem nela irão se conectar ao banco.

- **Se a ação correta for executada os usuários perceberão que foi feito avanço em relação à tarefa desejada?**

Sim. O aplicativo irá para a tela de seleção de bancos com o nome da conexão no título, deixando claro ao usuário que a conexão foi realizada.

- Selecionar o banco de dados:

- **Os usuários saberão qual a ação correta para atingir o resultado desejado?**

Sim. O título *Database List* informa que a lista se trata dos bancos existentes no servidor e o usuário saberá que tem que escolher uma base.

- **Os usuários perceberão que a ação correta está disponível?**

Sim. Os bancos de dados existentes vão estar listados na tela, e, devido à experiência com outros aplicativos *mobile*, o usuário vai saber que pode clicar nele.

- **Os usuários irão associar a ação correta com o resultado desejado?**

Sim. Os bancos de dados existentes vão estar listados na tela, e, devido à experiência com outros aplicativos *mobile*, vão entender que se clicarem nele irão se conectar ao banco.

- **Se a ação correta for executada os usuários perceberão que foi feito avanço em relação à tarefa desejada?**

Sim. Após clicar no banco de dados, o aplicativo irá para a tela principal com as tabelas existentes listadas.

- Ir para a tela de criação de tabelas:

- **Os usuários saberão qual a ação correta para atingir o resultado desejado?**

Sim. Após entrar na tela com a lista de tabelas, o usuário vai saber por experiência que o próximo passo é ir para a tela de criação de tabelas.

- **Os usuários perceberão que a ação correta está disponível?**

Sim. Na tela de tabelas o botão '+' está bem visível.

- **Os usuários irão associar a ação correta com o resultado desejado?**

Sim. O usuário sabe que está na tela de tabelas e sabe, devido à experiência em aplicativos mobile, que o símbolo ‘+’ é normalmente associado a criação de um novo objeto.

- **Se a ação correta for executada os usuários perceberão que foi feito avanço em relação à tarefa desejada?**

Sim. O aplicativo irá para uma tela com o título *New Table*.

- Adicionar coluna:

- **Os usuários saberão qual a ação correta para atingir o resultado desejado?**

Sim. Após entrar na tela de criação de tabelas, o usuário vai saber por experiência que o próximo passo é criar as colunas da tabela.

- **Os usuários perceberão que a ação correta está disponível?**

Sim. O botão ‘+’ está bem visível.

- **Os usuários irão associar a ação correta com o resultado desejado?**

Sim. A tela tem *Column List* como subtítulo deixando claro que o usuário precisa cadastrar as colunas da tabela e o botão ‘+’ vai ser associado a criação de colunas devido à experiência em aplicativos *mobile*.

- **Se a ação correta for executada os usuários perceberão que foi feito avanço em relação à tarefa desejada?**

Sim. O aplicativo irá para uma tela solicitando as informações da coluna que será criada.

- Preencher formulário:

- **Os usuários saberão qual a ação correta para atingir o resultado desejado?**

Sim. Após entrar na tela de criação de coluna, o usuário vai ver o formulário solicitando as informações da coluna que será criada.

- **Os usuários perceberão que a ação correta está disponível?**

Sim. O formulário está bem visível na tela.

- **Os usuários irão associar a ação correta com o resultado desejado?**

Sim. Os campos do formulário vão deixar claro que se trata de informações referentes à criação de coluna.

- **Se a ação correta for executada os usuários perceberão que foi feito avanço em relação à tarefa desejada?**

Sim. Após preencher o formulário, o usuário sabe por experiência que basta confirmar a criação da coluna.

- Confirmar criação da coluna:
  - **Os usuários saberão qual a ação correta para atingir o resultado desejado?**

Sim. Após preencher o formulário, o usuário sabe por experiência que basta confirmar a criação da coluna.
  - **Os usuários perceberão que a ação correta está disponível?**

Sim. O botão *ADD COLUMN* está bem visível na tela.
  - **Os usuários irão associar a ação correta com o resultado desejado?**

Sim. O usuário vai saber por experiência que após preencher o formulário o botão *ADD COLUMN* vai criar a coluna.
  - **Se a ação correta for executada os usuários perceberão que foi feito avanço em relação à tarefa desejada?**

Sim. O aplicativo irá mostrar na tela de criação de tabela a coluna adicionada.
- Criar tabela:
  - **Os usuários saberão qual a ação correta para atingir o resultado desejado?**

Sim. Após adicionar as colunas, o usuário vai saber que basta criar a tabela.
  - **Os usuários perceberão que a ação correta está disponível?**

Sim. O botão *CREATE TABLE* está bem visível na tela.
  - **Os usuários irão associar a ação correta com o resultado desejado?**

Sim. O botão *CREATE TABLE* diz exatamente o que ele faz.
  - **Se a ação correta for executada os usuários perceberão que foi feito avanço em relação à tarefa desejada?**

Sim. O aplicativo voltará para a tela com a lista de tabelas e a tabela recém-criada estará na lista.

## 5.6. Considerações Finais

O SQL Manager apresentou diversas melhorias quando comparado a aplicativos disponíveis no mercado. Os principais objetivos do aplicativo foram melhorar a usabilidade da interação, aumentar o número de tarefas implementadas via interface e contornar as limitações de dispositivos móveis, otimizando a escrita de consultas e a visualização de dados em formato tabular. Praticamente todas as funções consideradas básicas foram implementadas e o sistema apresentou uma boa usabilidade nos testes realizados. O Quadro 5.1 apresenta um resumo da análise realizada no SQL Manager e nos demais aplicativos avaliados.

Quadro 5.1 - Resumo de análise realizada em todos os aplicativos.

	Funções não contempladas	Problemas de usabilidade				Limitações não contornadas
		Avaliação heurística	Percurso cognitivo			
			Consultas	Inserção de dados	Criação de tabela	
SQL Client	12	7	4	-	-	1
Mobile SQL Client	10	5	0	-	-	2
SQL Studio Lite	8	5	5	-	-	1
Mobile SQL Server Manager	7	6	6	-	2	1
RemoDB	9	5	2	1	-	1
SQL Manager	3	1	0	0	0	0

Fonte: O Autor (2016).

O próximo capítulo apresentará uma visão geral do trabalho desenvolvido, limitações do SQL Manager e sugestões de trabalhos futuros.

## 6. Considerações Finais

Apesar da capacidade atual dos dispositivos móveis e da mobilidade que uma ferramenta de acesso a banco de dados *mobile* pode proporcionar, o uso de aplicativos para essa função ainda é escasso. O motivo disso são as limitações referentes a dispositivos móveis combinadas com a má usabilidade e incompletude dos produtos disponíveis no mercado.

As principais restrições dos dispositivos móveis vêm da necessidade de portabilidade e mobilidade, que força a redução do tamanho dos dispositivos, e conseqüentemente a redução dos seus componentes e restrição dos dispositivos de entrada. Essas restrições foram bastante atenuadas com a popularização da tecnologia *touch screen*, a qual permite o melhor uso da tela, apresentando botões, opções e métodos de entrada lógicos quando necessário em vez de físicos que ocupam espaço sempre. Com essa tecnologia a seu favor, os desenvolvedores precisam otimizar o espaço da tela e criar formas de contornar as limitações existentes.

O estudo da usabilidade e dos métodos de avaliação é essencial na tarefa de criar telas que sejam compreendidas pelo usuário, fáceis de usar, intuitivas e que apresentem boa visibilidade. Com esses conceitos em mente e aplicando as avaliações adequadas, muitos problemas que dificultariam o uso e a compreensão de um sistema podem ser evitados.

Esse trabalho utilizou conceitos e métodos de usabilidade para tornar viável o uso de aplicações de acesso a banco de dados por meio de dispositivos móveis, contornando as limitações dos dispositivos móveis e os problemas apresentados por outros aplicativos. Como resultado, foi apresentada uma aplicação com melhoria em vários aspectos quando comparada com as aplicações selecionadas. Apesar disso, a aplicação desenvolvida possui limitações e pontos de melhoria, como a adição de novas tarefas, funcionalidades e opções mais avançadas.

### 6.1. Limitações

Apesar de ser mais completo que os aplicativos analisados e implementar a maioria das funções consideradas básicas, o SQL Manager possui limitações e poderia ser mais completo, abrangendo um maior número de tarefas, inclusive tarefas de gerenciamento de banco de dados mais avançadas.

A principal limitação é a falta de visualização via interface de alguns objetos de bancos de dados, como *procedures*, *functions* e *views*, tarefas que foram consideradas básicas nesse trabalho. Outra limitação apresentada é a falta de gerenciamento de restrições (e.g. *foreign key*,

*unique key, default, check*) que, apesar de não serem consideradas básicas nesse trabalho, são tarefas importantes e muito realizadas, principalmente por administradores de bancos de dados.

## 6.2. Trabalhos Futuros

Diversas contribuições podem ser realizadas em trabalhos futuros. Entre elas, a avaliação de usabilidade da aplicação desenvolvida utilizando métodos empíricos. A avaliação com o público alvo permite: i) gerar dados estatísticos importantes na comparação com as demais aplicações; ii) encontrar problemas que não foram identificados com os métodos analíticos; e iii) receber *feedback* dos usuários finais.

Outra possível contribuição é o tratamento de dados multimídia, tanto na visualização quanto na inserção de dados. O SQL Manager não dá suporte à visualização de dados multimídias (e.g. fotos, vídeos áudios), apresentando apenas o valor do campo binário. A visualização de dados multimídia e a inserção de dados encontrados no próprio dispositivo é algo que não existe na maioria das ferramentas de banco de dados em ambiente *desktop*, mas que representaria uma vantagem atrativa para esse tipo de aplicativo.

Ainda com o objetivo de trazer mais facilidade e flexibilidade para os usuários, podem ser adicionadas funcionalidades úteis em aplicativos desse tipo. Por exemplo, pode existir uma funcionalidade que permita ao usuário ver e alterar o comando que seria executado em uma tarefa realizada via interface, semelhante a como pode ser realizado no SQL Server Management Studio. Isso traz maior flexibilidade na execução de tarefas, especialmente para usuários mais experientes. Outra função interessante seria permitir o salvamento de consultas e comandos executados, permitindo seu reaproveitamento.

E com o objetivo de criar ferramentas mais completas, pode-se ainda adicionar tarefas de gerenciamento de banco de dados como, por exemplo, realização de *backups, restores*, gerenciamento de usuários, *logins, jobs* e outros objetos de banco de dados.

## Referências

- AMADO, P. G. F. **Bancos de dados móveis: visão geral, desafios e soluções atuais**. Universidade Federal de Pernambuco, 2002.
- BARBOSA, S. D. J.; SILVA, B. S. DA. **Interação humano-computador**. Elsevier, 2010.
- BETIOL, A. H. **Avaliação de usabilidade para os computadores de mão: um estudo comparativo entre três abordagens para ensaios de interação**. Universidade Federal de Santa Catarina, 2004.
- BREWSTER, S. A.; CRYER, P. G. **Maximising screen-space on mobile computing devices**. CHI'99 extended abstracts on Human factors in computing systems. ACM, 1999.
- CEREJO, L. The elements of the mobile user experience. **Smashing Magazine**, 2012.
- FERREIRA, D. DE S. **Abordagem híbrida para avaliação da usabilidade de dispositivos móveis**. Universidade Federal de Campina Grande, 2007.
- FIGUEIREDO, C. M. S.; NAKAMURA, E. Computação móvel: Novas oportunidades e novos desafios. **T&C Amazônia**, v. 1, n. 2, 2003.
- FILARDI, A. L. **Análise e avaliação de técnicas de interação humano-computador para sistemas de recuperação de imagens por conteúdo baseadas em um estudo de caso**. Instituto de Ciências Matemáticas e de Computação, 2007.
- GALLIANO, E. **Bancos de dados móveis**. Universidade Estadual de Londrina, 2007.
- HEWETT, T. T. et al. **ACM SIGCHI curricula for human-computer interaction**. ACM, 1992.
- JORDAN, P. W. et al. **Guessability, learnability, and experienced user performance**. (Diaper, D. & Hamond, N., Ed.) People and Computers VI: Usability Now! (Proceedings of HCI' 91). Cambridge University Press, 1991.
- JORDAN, P. W. What is usability? **Contemporary ergonomics**, p. 454, 1994.
- JORDAN, P. W. **An introduction to usability**. CRC Press, 1998.
- LEWIS, C. et al. **Testing a walkthrough methodology for theory-based design of walk-up-and-use interfaces**. Proceedings of the SIGCHI conference on Human factors in computing systems. ACM, 1990.

- NIELSEN, J. **Usability engineering**. Cambridge, MA: Academic Press Inc, 1993.
- NIELSEN, J. Heuristic Evaluation. **Usability Inspection Methods**, p. 25–62, 1994.
- NIELSEN, J.; MOLICH, R. **Heuristic evaluation of user interfaces**. Proceedings of the SIGCHI conference on Human factors in computing systems. Seattle, WA: ACM, 1990.
- POLSON, P. G. et al. Cognitive walkthroughs: a method for theory-based evaluation of user interfaces. **International Journal of man-machine studies**, v. 36, n. 5, p. 741–773, 1992.
- PRAYAGA, S. **Mobile user interface and usability guide**. Disponível em: <<https://pshyama.wordpress.com/2010/06/17/mobile-user-interface-and-usability-guide/>>. Acesso em: 21 jun. 2016.
- ROCHA, H. V. DA; BARANAUSKAS, M. C. C. **Design e avaliação de interfaces humano-computador**. Campinas, SP: Unicamp, 2003.
- SHACKEL, B. Usability-context, framework, definition, design and evaluation. **Human factors for informatics usability**, p. 21–37, 1991.
- URRIZA, J. M. et al. **Economia de energia em dispositivos móveis**. VI Workshop de Comunicação sem Fio e Computação Móvel. 2004.
- WHARTON, C. et al. The cognitive walkthrough: a practitioner's guide. **Usability Inspection Methods**, 1994.