

a cometer erros ou ficar confuso. Nesse caso, a opinião subjetiva da pessoa que recebeu a atribuição de avaliar pode ser decisiva.

Além do desafio de trabalhar com tecnologias em constante mudança, cada método usado para avaliar a usabilidade nos dispositivos móveis atuais possui suas vantagens e desvantagens. Alguns são difíceis de se aplicar e outros são dependentes de opiniões subjetivas dos usuários ou do uso de instrumentos [26].

Em uma pesquisa conduzida para uma dissertação de mestrado, Neto [22] mostrou que as heurísticas de Nielsen são deficientes em identificar problemas de usabilidade de aplicações de dispositivos móveis. O autor fez um período de experimentação de um mês em 4 aplicativos para a plataforma Android e encontrou vários problemas de usabilidade, sendo que alguns não se encaixavam em nenhuma das heurísticas de Nielsen. Isso mostrou que novas heurísticas seriam necessárias e, através de alguns ciclos de brainstormings com outros 5 especialistas e testes com usuários, as validou e refinou. Como resultado, temos uma versão aprimorada das heurísticas de Nielsen para dispositivos móveis, a qual está relacionada a seguir:

1. **Bom aproveitamento do espaço da tela, de acordo com a orientação** - Independentemente da orientação do dispositivo, o design deve ser realizado de forma que os itens não fiquem muito distantes, nem muito juntos. Elementos relacionados devem estar próximos e os sem relacionamento devem estar mais afastados. Interfaces não devem estar carregadas com muitos elementos.
2. **Consistência e padrões da interface** - A aplicação deve manter os componentes no mesmo lugar e na mesma configuração ao longo de toda a interação, para facilitar a aprendizagem. Funcionalidades análogas devem possuir interações análogas, por meio de atividades parecidas. As características de cada componente (seu tamanho, fonte, cor, etc.) devem permanecer os mesmos em

toda a aplicação.

3. **Visibilidade e acesso fácil a toda informação existente** - Todas as informações devem ser visíveis e legíveis, tanto em retrato quanto em paisagem. O usuário não deve se esforçar para encontrar ou entender qualquer informação sendo transmitida. Isso também vale para mídias, que devem de ser vistas ou executadas na íntegra. Os elementos da interface devem possuir contraste e elementos de um mesmo grupo de informações devem ter alinhamento adequado.
4. **Adequação entre o componente e sua funcionalidade** - O usuário deve saber exatamente o que ele deve colocar como entrada a um componente, sem que haja ambiguidades ou dúvidas. Metáforas de funcionalidades devem ser compreendidas sem dificuldades.
5. **Adequação de mensagem à funcionalidade e ao usuário** - A aplicação deve falar a linguagem do usuário e as instruções para executar as funcionalidades devem ser claras e objetivas. A leitura deve ser natural e a linguagem não deve ser invasiva no sentido de obrigar o usuário a fazer algo.
6. **Prevenção de erros e retomada rápida ao último estado estável** - O sistema deve ser capaz de se antecipar a uma situação que leve a algum erro por parte do usuário com base em alguma atividade já realizada pelo usuário. Quando um erro ocorrer, a aplicação deve avisar o usuário prontamente e retornar ao último estado estável. Em casos em que o retorno ao último estado seja difícil, o sistema pode transferir o controle para o usuário, para que este decida o que fazer (para onde ir).
7. **Facilidade de entrada de dados** - A forma com que o usuário fornece os dados pode se basear em tecnologias assistivas, mas a aplicação deve sempre mostrar com legibilidade o que está sendo digitado, para que o usuário tenha total controle da situação. O usuário deve conseguir fornecer os dados

requeridos de forma prática.

8. **Facilidade de acesso às funcionalidades** - As funcionalidades principais da aplicação devem ser realizadas com maior facilidade possível, preferencialmente em apenas uma interação. Nenhuma funcionalidade deve ser difícil de encontrar na interface da aplicação. Todos os componentes de entrada devem ser assimilados com facilidade.
9. **Feedback imediato e fácil de ser notado** - O feedback deve ser fácil de ser notado, para que não haja dúvidas de que a operação foi realizada ou está em andamento. Atualizações locais na página devem ser priorizadas, para evitar recarregamento e perda do ponto em que o usuário estava. Mensagens que aparecem muitas vezes devem ter opção de serem ocultadas pelo usuário. Barras de progresso demoradas devem permitir que o usuário continue executando outras atividades. Feedbacks positivos devem ser visíveis, mas não exigir interação redundante com o usuário, para não estressá-lo.
10. **Aplicação deve ser autocontida** - A aplicação deve conter todas as funcionalidades e informações embutidas nela, sem a necessidade de encaminhar o usuário para programas externos.
11. **Ajuda e documentação** - O aplicativo deve possuir opção de Ajuda para especificar os problemas comuns e as formas de solucioná-los. Os assuntos considerados nessa opção devem ser fáceis de serem encontrados.
12. **Minimização da carga de Memória do usuário** - Aplicações devem permitir que o usuário obtenha a informação de que precisa com facilidade, sem exigir que o usuário memorize passos anteriores para completar uma atividade.
13. **Personalização** - A interface deve permitir opções de configurações pessoais dos usuários, como aumentos do tamanho das letras, disposição de elementos na tela, etc.