Roteiro para fazer testes

Acessado a página do Development Android

[Testar a acessibilidade do seu app  |  Desenvolvedores Android  |  Android Developers](https://developer.android.com/guide/topics/ui/accessibility/testing?hl=pt-br)

Os testes de acessibilidade viabilizam encontrar problemas de usabilidade de acessibilidade que durante o desenvolvimento do aplicativo algumas vezes não é possível identificar.

Mesmo ferramentas como o Lint tem a característica de realizarem analises estáticas cobrindo relativamente aspectos que o usuário durante o uso do aplicativo não será beneficiado.

Portanto é necessário recorrer a outros recursos para tornar o aplicativo mais poderoso e versátil. Dessa forma todos os usuários serão beneficiados, incluindo idosos e portadores de deficiências.

Há várias estratégias de testes entre as mais conhecidas temos:

* **Teste manual:** Quando o App é utilizado com o uso dos serviços de acessibilidade do Android. Exemplo: TalkBack, Acesso com interruptor, **BrailleBack.**
* **Testes com ferramentas de análise:** Pode ser usado ferramentas para descobrir meios de melhorar a acessibilidade do app. Exemplo: Scanner.
* **Teste automatizado:** Pode ser usado o teste de acessibilidade no Espresso e no Robolectric.
* **Testes de usuários:** Pode ser usado feedback de pessoas reais que interagem com seu app.

Resultado do teste com Talkback

Resultado do teste com o interruptor

Resultado do teste com o Scanner

O app Scanner analisa a tela e baseado em padrões de acessibilidade fornece sugestões específicas depois de analisar marcadores de conteúdo, itens clicáveis, contraste e muito mais.

No 1º teste feito efetuado com o Scanner ligado foram coletadas 77 imagens e 77 arquivos textos.

<https://www.youtube.com/watch?v=riAGnGv5aAs>

Vídeo orientando sobre as diversas ferramentas e configuração para realizar testes de validação de acessibilidade.

<https://www.lambda3.com.br/2022/08/guia-de-cores-para-acessibilidade/>

Site falando sobre o uso adequado de cores para estimular a acessibilidade.

No referido é os tópicos abordados serão acessibilidade das cores, contraste, contraste tem papel limitado e ferramentas.

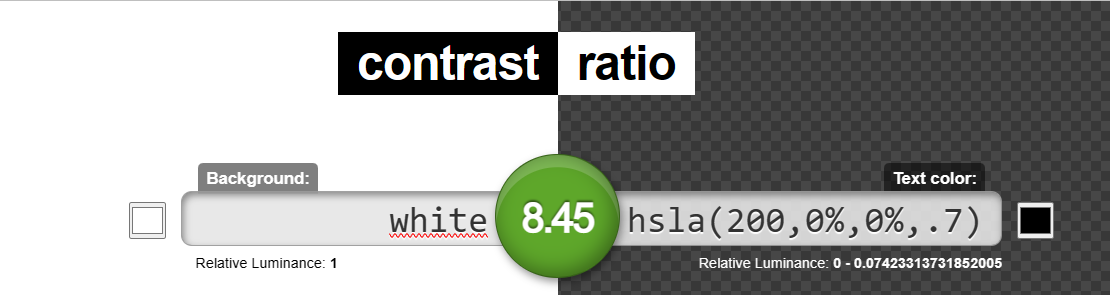
Necessário é ter uma direção de design inclusivo (e empatia). Acessibilidade é a usabilidade para pessoas que interagem com produtos de formas diferentes. É importante ter uma compreensão clara das principais categorias de deficiências, limitações ou restriçoes para oferecer recursos de acessibilidade.

* **Deficiências visuais** – cegueira e baixa visão, daltonismo…
* **Deficiências auditivas** – surdez e baixa audição, zumbido…
* **Problemas motores** – tremores nas mãos, deformidades físicas ou amputações…
* **Distúrbios cognitivos** – dislexia, demência ou privação de sono…

É importante entender que todos enfrentam algum tipo de deficiência permanente, temporária ou situacional em sua vida. Um exemplo interessante para entender esse conceito é: ter apenas um braço é uma condição permanente, ter um braço engessado é temporário e segurar um bebê em um braço é situacional – mas em cada caso você está restrito a completar tarefas com um braço.

[Contrast Ratio: Easily calculate color contrast ratios. Passing WCAG was never this easy! (contrast-ratio.com)](https://contrast-ratio.com/)

Site para fazer teste de cores de fundo com cores de primeiro plano que indica o nível de constrate



**Relatório de pré-lançamento no Google Play**

Google Play executa testes de acessibilidade

Tamanho da área de toque

Elementos interativos no seu app que têm uma área focalizável ou um [tamanho da área de toque](https://developer.android.com/guide/topics/ui/accessibility/apps?hl=pt-br#touch-targets) menor que o recomendado.

Baixo contraste

Instâncias em que o par de cores usado para um elemento de texto e o plano de fundo atrás desse elemento têm uma [proporção de contraste de cor](https://developer.android.com/guide/topics/ui/accessibility/apps?hl=pt-br#color-contrast) menor que o recomendado.

Marcação de conteúdo

Elementos da IU que não têm um [marcador que descreva a finalidade de um determinado elemento](https://developer.android.com/guide/topics/ui/accessibility/apps?hl=pt-br#label-elements).

Implementação

Atributos designados a elementos da IU que dificultam a interpretação correta dos elementos pelos serviços de acessibilidade do sistema. Entre os exemplos estão a definição de uma descrição para um [rótulo View editável](https://support.google.com/accessibility/android/answer/6378120?hl=pt-br) e o uso de uma [ordem de apresentação](https://support.google.com/accessibility/android/answer/7664232?hl=pt-br) de um elemento que não corresponda à disposição lógica dos elementos.

Verificando guia de Acessibilidade

<https://www.sidi.org.br/guiadeacessibilidade/index.html#requisitos>

Analise do aplicativo AUM

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Requisito | Tipo | Descrição | Detalhes | Atendido | Resultado do Teste | Justificativa |
| R01 | Desejável | Os componentes e informações da interface devem contribuir diretamente para a funcionalidade da aplicação. | Não utilizar componentes com propósito apenas estético pois isso aumenta a quantidade de informações para o usuário com deficiência visual memorizar quando explora e interage com a interface, aumentando a carga cognitiva. | Sim | Removido Tela de Boas vindas |  |
| R02 | Mandatório | Os componentes da interface devem ser entendidos sem a utilização de cores. | Não utilizar apenas cores para identificar ou indicar a funcionalidade de componentes. Tanto o entendimento quanto a interação de usuários, sobretudo com perda parcial da visão ou daltônicos, podem ser prejudicados. | Parcial | A partir da captura de tela verifica-se que apenas o uso da paleta de cores não é eficiente para daltônicos. <https://carlos-andrade-aum.blogspot.com/p/testes-realizados-atraves-do-aplicativo.html> | A paleta de cores para identificar medicamentos |
| R03 | Mandatório | Os componentes da interface devem utilizar cores com alto contraste em relação ao plano de fundo. | A falta de alto contraste entre os componentes pode fazer usuários com perda parcial da visão ignorá-los. O contraste pode ser verificado por várias ferramentas, como a <http://gmazzocato.altervista.org/colorwheel/wheel.php> | Sim | Testes realizados pelo aplicativo Scanner  <https://carlos-andrade-aum.blogspot.com/p/realizados-atraves-do-aplicativo.html> |  |
| R04 | Desejável | A interface sempre deve oferecer a opção para a orientação vertical da tela. | Usuários com deficiência visual utilizam o aparelho posicionado na vertical em uma das mãos e interagem com a outra. Desta forma, a melhor posição do telefone é na vertical e, portanto, os aplicativos devem oferecer essa opção. | Sim | Testes realizados pelo aplicativo Scanner  <https://carlos-andrade-aum.blogspot.com/p/realizados-atraves-do-aplicativo.html> |  |
| R05 | Mandatório | A aplicação deve notificar o usuário antes de alterar a orientação da tela, caso seja apropriado modificá-la. | Não informar a mudança na orientação da tela pode atrapalhar a interação, pois o usuário com deficiência visual precisará explorar a tela novamente por causa da mudança de posicionamento dos componentes. Além disso, há a possibilidade de acionar componentes erroneamente, o que pode causar até a perda de dados. | Não | Não previsto | O Aplicativo deve ser executado apenas na orientação vertical. Embora funcione na horizontal |
| R06 | Mandatório | A aplicação deve usar linguagem adequada ao contexto do usuário. | Não utilizar termos técnicos ou em outra língua, que não façam parte do conhecimento popular do público-alvo da aplicação. | Sim | Testes realizados pelo aplicativo Scanner  <https://carlos-andrade-aum.blogspot.com/p/realizados-atraves-do-aplicativo.html> |  |
| R07 | Mandatório | Todos os componentes da interface devem possuir rótulos. | Certifique-se que todos os elementos visuais (incluindo componentes não-textuais) da interface da aplicação estejam rotulados na implementação do código, para que os componentes sejam devidamente identificados pelo usuário com deficiência visual. Exemplo: Ler "home\_activity" ao invés do correto, "Tela inicial". | Sim | Testes realizados pelo aplicativo Scanner  <https://carlos-andrade-aum.blogspot.com/p/realizados-atraves-do-aplicativo.html> |  |
| R08 | Mandatório | Os rótulos devem descrever a funcionalidade ou significado e o estado dos componentes da interface de forma clara, sucinta e completa. | Garantir que a descrição de conteúdo legendado expresse de forma clara, sucinta e completa a funcionalidade ou significado dos elementos da tela. Ex.: "S-Planner – Agenda", "Botão Voltar", "Campo ‘Inserir seu nome'". | Parcialmente | Testes realizados pelo aplicativo Scanner  <https://carlos-andrade-aum.blogspot.com/p/realizados-atraves-do-aplicativo.html> | As listas suspensas e a lista da tela principal não estão informando os itens restantes na tela em áudio. |
| R09 | Mandatório | A aplicação deve fornecer as informações textuais ou numéricas contextualizadas para o conversor de texto para voz. | As leituras devem ser feitas levando em consideração o contexto da informação, conduzindo o usuário a assimilá-la, de forma fácil e rápida. Exemplo: Para um número de telefone, ler os números um por vez e não por extenso - Ler Nove, Oito, Oito... ao invés de Novencentos e oitenta e oito milhões... para 988764000". | Não Testado |  | Depende de fazer um teste com o leitor de telas e verificar o resultado de widget por widget |
| R10 | Mandatório | Os componentes não-textuais devem ser nomeados de forma que sejam compreendidos independente do contexto. | Os componentes não-textuais aqui abordados são aqueles representados geralmente por alguma imagem característica, como um "+" para adicionar algo ou uma "casa" para indicar a tela inicial da aplicação. Caso o componente possua alguma especificidade, não indicá-la pode deixar o usuário confuso sobre a funcionalidade que representa. Exemplo: Ao invés de "Botão Salvar", utilizar "Botão Salvar Usuário" (em uma tela de cadastro de usuário) e "Botão Salvar Cartão" (em uma tela de cadastro de cartão). | Sim | Testes realizados pelo aplicativo Scanner  <https://carlos-andrade-aum.blogspot.com/p/realizados-atraves-do-aplicativo.html> |  |
| R11 | Mandatório | O rótulo dos componentes com itens sequenciais e/ou paginados deve informar o intervalo que está sendo mostrado e o número total de itens. | Informar e identificar a ordenação das informações e componentes listados e/ou sequenciais, colabora para que o usuário não deixe de percebê-los durante a interação. Exemplo: "Página 01 de 03", "Cinco itens listados", "Item 1 de 5". | Sim | Testes realizados pelo aplicativo Scanner  <https://carlos-andrade-aum.blogspot.com/p/realizados-atraves-do-aplicativo.html> |  |
| R12 | Mandatório | Todas as imagens e figuras da interface da aplicação devem possuir uma audiodescrição. | A audiodescrição é utilizada para descrever o conteúdo de uma imagem, como se fosse uma narração. Para isso, deve ser definido o texto que será lido pelo leitor de telas no código da aplicação. Esse recurso só deve ser implementado para as imagens do próprio aplicativo, excluindo assim arquivos recebidos de outras aplicações. | Não aplicado |  | O Aplicativo não possui imagens |
| R13 | Mandatório | A área de toque dos componentes da interface deve ter tamanho e espaçamento que facilitem a interação do usuário. | ara garantir o equilíbrio entre densidade e usabilidade, a Google recomenda que componentes de toque devem ter no mínimo 48 x 48 dp (Density-independent Pixels - uma unidade abstrata baseada na densidade física da tela. Essas unidades são relativas à tela de 160 dpi). Na maioria dos casos, deve existir um espaçamento de pelo menos 8dp entre eles. Esses componentes devem ter um tamanho físico de aproximadamente 9mm, sendo que o tamanho recomendado em telas de toque é entre 7 e 10 mm. | Sim | Testes realizados pelo aplicativo Scanner  <https://carlos-andrade-aum.blogspot.com/p/realizados-atraves-do-aplicativo.html> |  |
| R14 | Desejável | Os componentes da interface das funções mais importantes/utilizadas devem estar situados próximos das extremidades da tela. | Usuários com deficiência visual tendem a procurar botões nas extremidades da tela, em especial na área superior da interface. | Parcial | Testes realizados pelo aplicativo Scanner  <https://carlos-andrade-aum.blogspot.com/p/realizados-atraves-do-aplicativo.html> | A lista de medicamentos cadastrados e o cadastro de horário utilizam um botão na parte inferior da tela. |
| R15 | Desejável | Cada tela da interface deve conter a menor quantidade de componentes de interação possível. | Quanto mais componentes na tela, mais tempo de exploração será necessário para que o usuário com deficiência visual assimile a tela. Além disso, há o risco do usuário não perceber algum componente. | Parcial | Testes realizados pelo aplicativo Scanner  <https://carlos-andrade-aum.blogspot.com/p/realizados-atraves-do-aplicativo.html> | O cadastro de Medicamento possui muito componentes em razão da inclusão do horário. |
| R16 | Desejável | Listas suspensas devem estar situadas na metade superior da tela. | Quando a lista suspensa está localizada no fim da tela, nem todos os itens são exibidos. Além disso a "janela" aberta com os itens pode ficar muito pequena. Desta forma, a aplicação dos gestos necessários para acessá-los (rolar a tela e gesto tab) necessita de muita precisão, o que traz bastante dificuldade para pessoas que não enxergam. | Não |  | O componente Spinner não apresentou esse comportamento de forma padrão. |
| R17 | Mandatório | Componentes de interação devem ter o espaçamento mínimo em relação à borda da tela. | Componentes localizados muito próximos aos limites da tela podem levar o usuário com deficiência visual a ter dificuldade de acessá-los, ou mesmo fazer com que toque erroneamente em outros componentes. Exemplo: Posicionar determinado componente no limite inferior da tela pode fazer o usuário tocar involuntariamente nos botões físicos do dispositivo, executando ações não esperadas, como voltar para a tela anterior ao clicar no botão "Voltar". | Sim | Testes realizados pelo aplicativo Scanner  <https://carlos-andrade-aum.blogspot.com/p/realizados-atraves-do-aplicativo.html> |  |
| R18 | Desejável | Componentes de interação devem manter tamanhos similares na mesma tela. | Devido à exploração por toque, interfaces com componentes de tamanhos variados colaboram para que usuários com deficiência visual não percebam os menores, sobretudo quando entre componentes grandes. No caso, usuários com perda parcial da visão, normalmente tocam em componentes grandes, que chamam mais sua atenção em detrimento dos menores. | Parcial | Testes realizados pelo aplicativo Scanner  <https://carlos-andrade-aum.blogspot.com/p/realizados-atraves-do-aplicativo.html> | A alteração de medicamento não atende ao requisito totalmente devido a existência de muitas funcionalidades na tela. |
| R19 | Desejável | A interface da aplicação com grande quantidade de informações deve ser estruturada em forma de lista ordenada. | Para facilitar a exploração dos itens deve ser usado um critério de ordenação para apresentação dos dados que melhor se adeque ao contexto da tela. Exemplo: Ordem alfabética, numérica, crescente/decrescente, recente/antigo, etc. | Parcial | Testes realizados pelo aplicativo Scanner  <https://carlos-andrade-aum.blogspot.com/p/realizados-atraves-do-aplicativo.html> | O cadastro de horário poderia ser chamado por um botão após do medicamento evitando o problema. No detalhe do medicamento as abas de horário/estoque e utilização poderia estar totalmente separada do resto. |
| R20 | Desejável | Componentes de formulários devem ser distribuídos um item por linha, evitando o uso de múltiplas colunas numa mesma linha. | Usuários com deficiência visual, sobretudo com perda total da visão, que exploram a tela deslizando o dedo (varrendo a tela) costumam fazê-lo de cima para baixo, em linha reta. Desse modo, se uma linha tiver mais de um componente, pode ser que o usuário não identifique um deles. | Não | Testes realizados pelo aplicativo Scanner  <https://carlos-andrade-aum.blogspot.com/p/realizados-atraves-do-aplicativo.html> | Na intenção de tornar o aplicativo com a centralização de recursos resultou em um acumulo de funcionalidades no detalhe de medicamentos e na inclusão de medicamentos |
| R21 | Mandatório | A barra de ferramentas deve permanecer fixa em telas de rolagem. | Usuários com deficiência visual se valem da memória para localizar componentes. Logo ocultar a barra durante a interação pode deixá-lo confuso sobre a estrutura da tela. | Sim | Testes realizados pelo aplicativo Scanner  <https://carlos-andrade-aum.blogspot.com/p/realizados-atraves-do-aplicativo.html> |  |
| R22 | Desejável | Componentes com textos devem ser curtos e concisos, sempre que possível. | A escuta de textos longos é cansativa para os usuários com deficiência visual. Por isso, aplicações que precisam oferecer algum texto longo, devem ter ele dividido em blocos, de modo que o usuário possa lê-lo por partes e, da mesma forma, recomeçar a leitura sem que seja do início. | Sim | Testes realizados pelo aplicativo Scanner  <https://carlos-andrade-aum.blogspot.com/p/realizados-atraves-do-aplicativo.html> |  |
| R23 | Desejável | As telas da aplicação devem seguir a mesma identidade visual e manter o mesmo padrão de layout. | Manter a mesma identidade visual ao longo de toda a aplicação faz com que a aplicação seja previsível, facilitando o aprendizado do usuário. | Sim | Testes realizados pelo aplicativo Scanner  <https://carlos-andrade-aum.blogspot.com/p/realizados-atraves-do-aplicativo.html> | O Número de telas é pequeno |
| R24 | Mandatório | A localização dos componentes deve seguir padrões amplamente utilizados. | O uso de padrões ajuda os usuários com deficiência visual a localizarem os componentes reduzindo assim a carga de memorização durante a exploração, bem como acelerando o aprendizado e a interação do usuário com a aplicação. Exemplo: botão "Salvar" no canto superior direito | Não |  | O botão confirmar horário que também inclui medicamento acabou não seguindo a recomendação |
| R25 | Mandatório | Os componentes de formulários devem seguir o mesmo padrão. | A aplicação deve manter um padrão de apresentação entre o label e o componente a ser preenchido, selecionado ou marcado. O mais comum para usuários com deficiência visual é primeiro o label e logo abaixo o componente referenciado por ele. | Não |  | Na intenção de tornar o aplicativo com a centralização de recursos resultou em um acumulo de funcionalidades no detalhe de medicamentos e na inclusão de medicamentos |
| R26 | Desejável | A aplicação deve manter a tela ativa por tempo ilimitado quando o leitor de tela estiver ativo. | Manter a tela ativa facilita a interação do deficiente visual com funcionalidades que exigem um tempo de leitura maior, como execução de vídeos e/ou apresentação de textos longos. | Sim |  | Depende de configuração do Android para o descanso de tela. |
| R27 | Desejável | A aplicação deve oferecer recursos que reduzam o esforço do usuário. | Como por exemplo, a função auto complete, utilizar comandos de voz e busca por voz. | Sim |  | Basicamente depende dos recursos do próprio Android como assistente de voz |
| R28 | Mandatório | O teclado utilizado pela aplicação deve ser compatível com o contexto do campo. | A depender do tipo de dado a ser preenchido, a aplicação deve oferecer o teclado mais adequado. Exemplo: Em campos em que o valor a ser inserido seja um e-mail, oferecer um teclado com as teclas arroba (@) e ".com". | Sim |  | Basicamente depende dos recursos do teclado do Android. O aplicativo apenas sinaliza o tipo de dado e o sistema Android faz o restante. |
| R29 | Mandatório | O teclado utilizado pela aplicação deve conter teclas que permitam a navegação entre os componentes da interface. | Apresentar no teclado o botão de navegação entre os campos de um formulário facilita a navegação entre eles, e consequentemente o preenchimento dos campos. Exemplo: inserir botão "Prox." quando o próximo campo é de entrada de dados. | Sim |  | Basicamente depende dos recursos do teclado do Android. |
| R30 | Desejável | Componentes de interação devem se manter na tela independente da funcionalidade que estiver em uso. | Esconder componentes, principalmente em funcionalidades que não necessitam uma interação ativa do usuário com deficiência visual, como a execução de videos ou apresentação de textos longos, pode deixá-lo confuso sobre a estrutura da tela. Além disso uma nova exploração da tela será necessária para encontrar os componentes. | Sim | Testes realizados pelo aplicativo Scanner  <https://carlos-andrade-aum.blogspot.com/p/realizados-atraves-do-aplicativo.html> |  |
| R31 | Mandatório | O leitor de telas deve informar ao usuário todos os eventos visíveis. | Usuários com perda parcial da visão normalmente se utilizam do resquício de visão que possuem para interagir com aplicativos. Por isso, é necessário que mesmo com o uso de leitor de telas, haja estímulos visuais sobre o que estiver ocorrendo na aplicação. Exemplo: O usuário deve ser informado sobre a conclusão correta do preenchimento de um formulário. | Sim | O Aplicativo emite um áudio após a execução das funções. |  |
| R32 | Mandatório | O leitor de telas deve informar o conteúdo de um componente assim que tocado, interrompendo qualquer leitura em andamento. | Deslizar o dedo na tela é uma das formas mais comuns de explorar a interface. Dessa forma, quando a leitura de um componente é interrompida, significa que o usuário está tocando outro componente. Caso a leitura não seja interrompida, o usuário concluirá que a tela não tem outros componentes tocáveis. Isso pode fazer com que componentes importantes na tela não sejam notados. | Sim | Testes realizados pelo aplicativo Scanner  <https://carlos-andrade-aum.blogspot.com/p/realizados-atraves-do-aplicativo.html> | Os testes do Lint e scanner além dos alertas do Android identificam elementos que precisam da propriedade Content description. Quando ela é preenchida o leitor de telas a utiliza. |
| R33 | Mandatório | A aplicação deve fornecer feedback sonoro sobre todas as ações executadas pelo usuário. | É necessário informar o resultado das ações do usuário para que ele não tente executá-la novamente sem necessidade. Esses feedbacks também podem ser realizados através de sons característicos de algumas operações, por exemplo, para indicar uma confirmação. | Parcial | O aplicativo AUM está realizando esse requisito apenas para o perfil de deficiência visual porem pelo que estamos percebendo seria necessário criar uma opção quando marcada emitiria áudio de confirmação. |  |
| R34 | Mandatório | A aplicação deve fornecer feedback visual sobre todas as ações executadas pelo usuário. | Usuários com perda parcial da visão devem ser informados visualmente do resultado das ações que realiza. | Parcial | O aplicativo AUM está realizando esse requisito apenas para o perfil de deficiência visual porem pelo que estamos percebendo seria necessário criar uma opção quando marcada emitiria áudio de confirmação. |  |
| R35 | Mandatório | As telas da aplicação, exceto *popups* (pequenas janelas que se abrem por cima da tela sendo visualizada), devem disponibilizar link para a tela principal do aplicativo. | Buscar a tela inicial da aplicação é um comportamento comum para usuários com deficiência visual quando se deparam com situações adversas durante a interação, como quando perdem a referência de qual tela da aplicação está sendo mostrada. Normalmente esse link pode estar representado por um botão "Home", ou mesmo dentro de um menu de gaveta (ou sanduíche). | Não |  | O aplicativo possui um número de telas bem reduzido. Principal / Inclusao de Medicamento / Detalhe Medicamento / Horario / Usuario / Login / Usuario / Inicial / Pesquisa Anvisa / Detalhe Anvisa |
| R36 | Mandatório | As telas da aplicação devem disponibilizar o botão "Voltar" para a tela anteriormente acessada pelo usuário. | O botão "Voltar" é muito utilizado por usuários com deficiência visual quando possuem alguma dificuldade na interação com a aplicação. Ele é utilizado para que o usuário volte até uma tela que conhece, ou mesmo para cancelar uma ação para fazê-la novamente. | Sim | As capturas de tela não estão contemplando essa funcionalidade. Foi adicionado em 19/02/2023 |  |
| R37 | Desejável | As telas da aplicação devem oferecer atalhos para suas principais funcionalidades. | Usuários com deficiência visual podem ter dificuldades para encontrar os componentes referentes às funcionalidades principais da aplicação. Sendo assim, uma das formas de minimizar esse problema é disponibilizar atalhos para as funcionalidades que se espera que sejam mais utilizadas. Exemplo: FAB, menu. | Parcial |  | Na intenção de tornar o aplicativo com a centralização de recursos resultou em um acumulo de funcionalidades no detalhe de medicamentos e na inclusão de medicamentos faltou colocar o atalho no cadastro de horário |
| R38 | Desejável | A aplicação deve oferecer uma navegação sequencial entre as telas. | Plataformas móveis, como o Android, utilizam o desempilhamento de telas. Contudo, para pessoas que não enxergam, é necessário que a aplicação utilize o desempilhamento "semântico", onde a pilha é limpa quando o usuário chega à tela inicial. Caso contrário, o usuário deixa de ter esse referencial, podendo ficar confuso. Exemplo: Navegando pelo aplicativo o usuário pode passar algumas vezes pela tela inicial. Utilizando o botão "Voltar", o aplicativo só deve permitir que o usuário volte até a tela inicial, não antes dela. | Sim | Após o usuário efetuar o acesso ao aplicativo AUM há apenas um botão sair na tela principal. O botão voltar de navegação não existe e o botão físico foi anulado na tela principal. |  |
| R39 | Mandatório | A aplicação deve suportar a navegação baseada em foco. | Para explorar a tela, é comum que usuários com deficiência visual utilizem o gesto "tab". Desta forma, a aplicação deve garantir que todos os componentes possam receber o foco, bem como que a ordem dele seja de cima para baixo e da esquerda para a direita. | Não testado |  | Depende de um teste com um leitor de telas |
| R40 | Mandatório | A aplicação deve informar possíveis erros de interação ao usuário. | A aplicação deve informar erros ocorridos durante a interação, evitando repetições, frustrações ou mesmo a desistência do uso decorrente de não entender o que está ocorrendo com a aplicação.  Exemplo: Quando o usuário salvar um formulário de cadastro com campo obrigatório não preenchido, informá-lo da ação. Assim que o usuário confirmar que visualizou a notificação do erro, alterar o foco para o campo que está faltando. | Parcial | Informa o erro. | O aplicativo não seta o foco do campo que dá o erro. |