

# APLICATIVO DE VIOLÊNCIA DOMÉSTICA CRIADA COM AUXÍLIO DE INTELIGÊNCIA ARTIFICIAL

**Daiane Dias Pereira<sup>1</sup>, Sergio Muinhos Barroso Lima<sup>2</sup>**

<sup>1</sup>Instituto Federal do Sudeste de Minas Gerais, Rio Pomba, Brasil([diasdaianedd@gmail.com](mailto:diasdaianedd@gmail.com))

<sup>2</sup>Instituto Federal do Sudeste de Minas Gerais, Rio Pomba, Brasil ([sergio.lima@ifsudestemg.edu.br](mailto:sergio.lima@ifsudestemg.edu.br))

**Resumo:** Este artigo tem como objetivo trazer uma análise, levantamento de facilidades e desafios sobre o uso da inteligência artificial (IA) para o desenvolvimento de um aplicativo para auxílio de mulheres vítimas de violência doméstica, e reflexões sobre o impacto da IA no mercado de trabalho de desenvolvimento de software. Para tanto, a IA foi utilizada em todas as fases, desde a concepção das funcionalidades do aplicativo, levantamento formal de requisitos, modelagem e desenvolvimento.

**Palavras-chave:** aplicativo; inteligência artificial; violência doméstica.

## INTRODUÇÃO

Este artigo aborda o desenvolvimento de um aplicativo, denominado Escuta Segura, com o auxílio da inteligência artificial. Este aplicativo foi implementado para abordar o tema da violência doméstica, haja visto o crescimento assombroso dos casos nos últimos anos. Um estudo recente, assim como vários outros, revela que “21,4 milhões de brasileiras sofreram algum tipo de violência nos últimos 12 meses, diz pesquisa” (G1, 2025), tornando o aplicativo relevante tanto sob o ponto de vista social quanto tecnológico, uma vez que abarca tema social de grande importância e explora a IA (inteligência artificial e refere-se à capacidade de sistemas computacionais simularem a inteligência humana) como ferramenta para o desenvolvimento de software.

A violência doméstica pode se apresentar de forma física, psicológica, sexual, patrimonial e moral. Apesar de existirem leis como a lei Maria da Penha (Lei nº 11.340/2006) e a Lei do Feminicídio (Lei nº 13.104/2015), muitas mulheres ainda sofrem caladas devido ao medo de retaliação, “No último ano, apenas 25,7% das mulheres que sofreram violência recorreram a órgãos oficiais, enquanto 33,8% buscaram apoio entre familiares e amigos.” (G1, 2025), como o caso da mulher que ligou para o 190 simulando o pedido de uma pizza para receber a ajuda contra a violência de seu marido (Souza, Araújo, 2025). Situações como estas são comuns nos dias atuais, o que torna importante, dentre outras

iniciativas, de auxílio tecnológico para facilitar a comunicação desse tipo de ocorrência.

Apesar de existirem alguns aplicativos com o intuito de ajudar essas mulheres, o aplicativo proposto neste trabalho visa trazer funcionalidades extras e a inovação tecnológica de ter sido criado com o auxílio da inteligência artificial desde sua especificação à finalização. A seguir, a tabela 1 exibe um comparativo dos aplicativos existentes no mercado.

Tabela 1. Comparativos de aplicativos existentes no mercado.

| Funcionalidade    | Escuta Segura (v1.2)        | Apps Comuns (ex: Life360, Sister) | Apps Governamentais (ex: SOS Mulher) |
|-------------------|-----------------------------|-----------------------------------|--------------------------------------|
| Identidade Visual | Disfarçada (Calculadora)    | Explícita (Segurança/Mapa)        | Explícita (Governo)                  |
| Conectividade     | 100% Offline (SMS Fallback) | Dependente de Internet/4G         | Dependente de Internet               |
| Privacidade       | Sem Login / Sem Nuvem       | Requer conta, e-mail e            | Requer CPF e cadastro                |

|                      |                                |                                   |                                |
|----------------------|--------------------------------|-----------------------------------|--------------------------------|
|                      |                                | telefone                          | oficial                        |
| Acionamento          | Voz (Palavra-chave) e Botão    | Botão ou agito do celular         | Botão físico ou na tela        |
| Escuta Passiva       | Sim (Local e Automática)       | Raramente (Geralmente apenas SOS) | Não                            |
| Hierarquia de Alerta | WhatsApp > SMS > Ligação > 190 | Notificação Push > SMS            | Direto para o COPOM (Polícia)  |
| Conteúdo Educativo   | Disponível                     | Requer carregamento online        | Links externos (Web)           |
| Armazenamento        | Apenas no dispositivo (Local)  | Servidores de terceiros (Nuvem)   | Base de dados de governamental |

Como visto na tabela 1, o sistema proposto diferencia-se por seu funcionamento ser offline, assegurando a assistência devida à vítima mesmo que não tenha acesso ou conectividade à rede de internet. A solução prioriza múltiplos canais para envio de notificações de emergência (WHATSAPP, SMS, Ligação e E-mail), além de possuir uma interface camuflada sob a forma de calculadora, essa funcionalidade projetada para preservação da integridade do usuário e mitigar inspeções indesejadas de terceiros ao dispositivo.

A integração da IA posiciona o desenvolvedor de software no quarto gradiente (nível de exposição da função à inteligência artificial), indicando o forte impacto sobre essa profissão (Sakamoto, 2025), ou seja, a IA impacta fortemente o desenvolvimento de software podendo ter efeitos benéficos, como a facilitação e a agilização no desenvolvimento, mas também pode trazer efeitos negativos no mercado de trabalho como a diminuição de postos. De fato, durante o desenvolvimento deste trabalho, ficou evidente os benefícios na agilidade de desenvolvimento, o que reforça as preocupações quanto aos impactos negativos no mercado de trabalho.

A inteligência artificial mostrou, neste trabalho, ser de grande valia como auxiliar ao processo de desenvolvimento, trazendo desde a concepção, modelagem, correção de erros, geração de código, configuração de ambientes de desenvolvimento integrado (IDEs) e armazenamento de dados.

Esse novo poder dado aos desenvolvedores de software traz porém, consigo, preocupações. Um recente artigo (ROOSE, 2025) alerta para o uso da IA em substituição dos recém-formados, ou seja, os desenvolvedores juniores. Roose ainda afirma que as empresas estão tentando substituir as atividades de nível básico e contratar analistas de nível avançado. A tendência, expressa nesse artigo, é que as empresas comecem a investir na IA para serviços que demandem pouca expertise e que também deixe de investir na especialização de juniores, o que pode prejudicar a evolução profissional destes e dificultar a entrada no mercado de trabalho.

O Github Copilot da Microsoft é um dos maiores plataformas dominantes de linha de código (CDB, 2025), mas existem outras como OpenAI, Anthropic ou DeepSeek, dentre outras inúmeras IAs, tanto de uso generalista como específicas para o desenvolvimento de software. É possível desenvolver sistemas com as IAs generalistas, como o chatGPT e o DeepSeek, por exemplo, mas elas dificultam na necessidade de cópias/criações manuais de arquivos e árvores de diretórios, bem como na configuração manual de IDEs, de acordo com os testes iniciais para o desenvolvimento deste trabalho. Já as específicas, como as IAs embutidas em IDEs, facilitam esse processo. A tabela 2 mostra um comparativo com algumas das principais IAs existentes no mercado.

Tabela 2. Comparativo Inteligências artificiais.

| Nome da IA | Empresa         | Gratuito | Tipo               | Transparência |
|------------|-----------------|----------|--------------------|---------------|
| Copilot    | Microsoft       | Parcial  | Genérica           | Caixa Preta   |
| ChatGPT    | OpenAI          | Parcial  | Genérica           | Caixa Preta   |
| Gemini     | Google          | Parcial  | Genérica           | Caixa Preta   |
| Claude     | Anthropic       | Parcial  | Genérica           | Caixa Preta   |
| Llama      | Meta            | Sim      | Genérica           | Caixa Branca  |
| Perplexity | Perplexity AI   | Parcial  | Específica (Busca) | Caixa Preta   |
| Midjourney | Midjourney Inc. | Não      | Específica         | Caixa Preta   |

|             |                 |                 |                      |              |
|-------------|-----------------|-----------------|----------------------|--------------|
| DeepSeek    | DeepSeek        | Sim / Parcial   | Genérica             | Caixa Branca |
| AlphaFold 3 | Google DeepMind | Sim (Acadêmico) | Específica (Ciência) | Caixa Branca |
| Devin AI    | Cognition       | Não             | Específica (Coding)  | Caixa Preta  |
| Grok        | xAI (Elon Musk) | Não             | Genérica             | Caixa Branca |

A inteligência artificial copilot foi selecionada devido sua popularidade no momento do desenvolvimento do projeto. No entanto, inteligências como copilot, chatgpt e gemini são modelos de IA proprietária, chamados caixas pretas, no qual o código-fonte, a arquitetura e o processo de treinamento permanecem ocultos aos usuários finais, o que pode injetar vieses negativos, como homofobia, racismo, misoginia e outras inclinações já mapeadas em alguns estudos. Entretanto, a geração de código em linguagens de programação são menos afetadas por esses vieses, sendo então a escolha do copilot ter sido decisiva por sua popularidade. Já as IAs caixas brancas são transparentes e permitem consulta pública aos códigos gerados, o que as torna mais seguras para aplicações de cunho social ou aquelas mais sensíveis a vieses advindos do processo de treinamento.

O objetivo deste artigo é demonstrar os resultados obtidos referente a eficácia da IA, como ferramenta de suporte para o ciclo de vida de desenvolvimento de software.

### MATERIAL E MÉTODOS

Este artigo utilizou a abordagem qualitativa tanto para a análise de documentação e artigos sobre violência doméstica e inteligência artificial. Inicialmente foi realizado um levantamento e estudo de trabalhos relacionados, logo após definido o software a ser desenvolvido, neste momento a IA foi consultada para trazer sugestões de aplicativos que pudessem trazer relevância e agregar algum valor à comunidade. As melhores sugestões foram escolhidas pela equipe do projeto, evidenciando a importância de usar a IA como instrumento auxiliar e não como última voz.

Uma vez escolhido o software a ser desenvolvido, dada a sua importância e relevância social, a especificação do aplicativo, de forma não técnica, ou seja, como se fosse a requisição de um cliente, ou pessoa leiga em computação, foi refinada pela inteligência artificial e revisada pela equipe.

A seguir a ferramenta a ser desenvolvida deveria entrar em fase de desenvolvimento, onde seria utilizada a metodologia clássica para esta etapa: especificação, modelagem, projeto, implementação e testes. O modelo clássico de desenvolvimento foi escolhido para que fosse mais fácil e clara a aferição da qualidade dos subprodutos gerados pela IA. À cada passo, a equipe avaliou a qualidade dos subprodutos que foram gerados pela IA, como por exemplo a especificação formal dos requisitos, a modelagem e o código fonte, em termos de legibilidade, coesão e indicadores de qualidade de software (PRESMAN MAXIN, 2021). Além disso, o software deveria prover boa ergonomia e acessibilidade digital de acordo com o padrão ARIA (MDN WEB DOCS, 2026) que preconiza a acessibilidade como um dos requisitos do software.

Resumidamente, a metodologia foi assim definida:

- definição de quais plataformas de IA generativas seriam utilizadas, desde as de uso genérico, como chat GPT e o Deep Seek, por exemplo, como as de uso específico para o desenvolvimento de software, como as embutidas em IDEs, por exemplo;
- a consulta nessas plataformas de IA sobre idéias de aplicativos que seriam atrativos e relevantes no cenário social, educacional ou ambiental, por exemplo;
- o refinamento, via IA, de um texto para uma especificação de um software, escolhido pela equipe, à partir do objetivo anterior, de forma leiga, ou seja, uma demanda de software feita sem especificações técnicas;
- a elaboração de um documento de especificação de requisitos técnico, segundo modelo IEEE 830 (IEEE, 2025);
- desenvolvimento clássico de sistemas: modelagem, implementação e testes;
- verificação de ergonomia (qualidade da interface com o usuário em termos gráficos, facilidade de operação, acessibilidade, etc.) e
- reflexões sobre o processo como um todo, buscando-se por vantagens, desvantagens e impactos no mercado de trabalho.

Em cada um desses passos, os resultados gerados pela IA foram analisados quanto a sua completude, correte e qualidade.

### RESULTADOS E DISCUSSÃO

A especificação do projeto foi desenvolvido por uma inteligência artificial chamada copilot, ela se mostrou uma boa ferramenta auxiliar no detalhamento das funcionalidades do sistema e orientar no desenvolvimento técnico desde sua concepção, mas com a devida validação humana, assim como o

processo de codificação e desenvolvimento do aplicativo também foi realizado pela mesma inteligência artificial, porém com a versão integrada ao ambiente de desenvolvimento integrado (IDE) visual studio code. A principal diferença entre a extensão no visual studio code e o copilot via navegador é que o integrado consegue ter acesso direto de edição ao código fonte projetado, o que facilita o processo de desenvolvimento, uma vez que é necessário copiar e colar o código gerado por IAs de uso genérico em navegadores para dentro das IDEs, onde, de fato, esses códigos devem ser trabalhados e testados.

O objetivo esperado dessa IA era que através de pequenos comandos em forma de frases (prompts) pudesse construir um aplicativo com as funcionalidades desejadas.

Inicialmente foram definidos os possíveis temas do aplicativo que teriam impacto positivo para a comunidade, esta etapa foi definida com o crivo da equipe do projeto, autores deste trabalho.

Posteriormente, foi feita a especificação formal do aplicativo, seguindo padrão IEEE 830 (IEEE, 2025). Onde foram levantados e registrados os requisitos funcionais e não funcionais do projeto.

Em seguida foram realizados os modelos de dados: o modelo conceitual, o modelo de dados, o modelo entidade de relacionamento, o diagrama de casos de uso e o diagrama de navegação. Também foram gerados os termo de uso, política de privacidade e termo de revogação.

Até este ponto o copilot ajudou no processo, como mostra a figura 1. Para o desenvolvimento do aplicativo em si, foi necessário o uso de inteligência artificial integrado ao editor de código-fonte, mas o copilot foi um auxiliar na escolha de uma IA adequada para a continuidade do projeto, ou seja, a sua implementação. O copilot ainda ajudou no processo de criação de imagens, como fluxograma de uso e telas simuladas (figura 2), no modelo de requisito IEEE conforme as diretrizes da MDN Web Docs (2026), mas foi necessária várias correções dessas imagens, como a sugestão de correção de palavras que foram geradas com erro ortográfico, até se chegar a algo que fosse considerado satisfatório, o que leva à conclusão da necessária a intervenção humana para refinamento e obtenção do resultado final esperado, não só para o caso e de geração de imagens, mas como em todas as etapas de desenvolvimento.

Um ponto de dificuldade constatado foi que o copilot recebeu atualizações no decurso do projeto e, após isso, não disponibilizou mais os históricos de conversação, e também foi necessário lembrá-lo das conversas anteriores, repassando-se novamente todas as informações necessárias para dar a continuidade ao

processo de desenvolvimento do projeto, o que atrasou o processo de desenvolvimento, e desqualifica o copilot em modo navegador, de uso genérico, como ferramenta adequada/profissional para o desenvolvimento de software.

## Etapas do Projeto

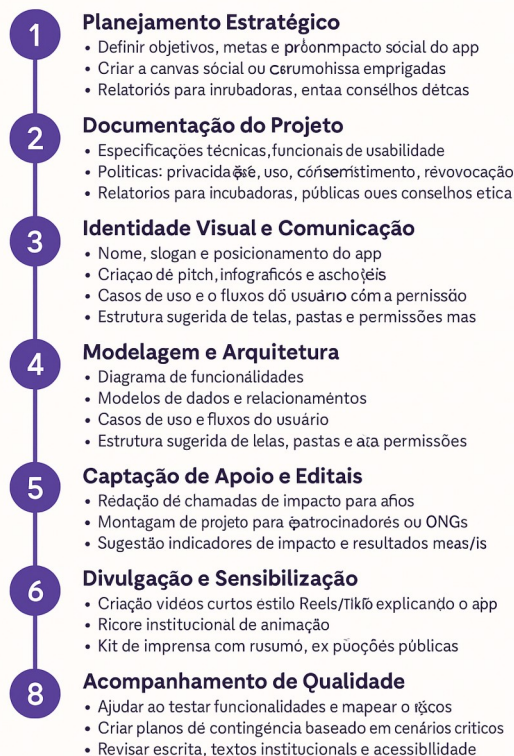


Figura 1. Etapas em que o copilot atuou em auxílio.



Figura 2. Tela Simulada do aplicativo Escuta Segura gerada pelo copilot.

No processo de criação do documento de especificação de requisitos, utilizando-se o copilot navegador, foram necessários vários refinamentos até se chegar ao documento tido como satisfatório pela equipe. A tabela 3 ilustra a quantidade de requisitos funcionais e não funcionais gerados pelo copilot à pedido e revisão da equipe. Foram solicitados a



geração de diagrama de caso de uso (figura 3) e diagrama de classes do projeto (figura 4).

Tabela 3. Quantidade de Requisitos.

| Requisitos     | Quantidades |
|----------------|-------------|
| Funcionais     | 10          |
| Não Funcionais | 07          |

Já na fase de implementação, utilizando o copilot integrado ao visual studio code, ao delegar os comandos para o Github copilot chat, com as instruções que o próprio copilot criou, ele gerou o projeto em react native, como solicitado, porém com erro na pasta android. Mas com os devidos comandos e o envio do erro ao chat integrado ao visual studio code, ele indicou o caminho para a solução, bem como o passo a passo para a execução do projeto e configuração do android studio que foi utilizado para executar o emulador do projeto.

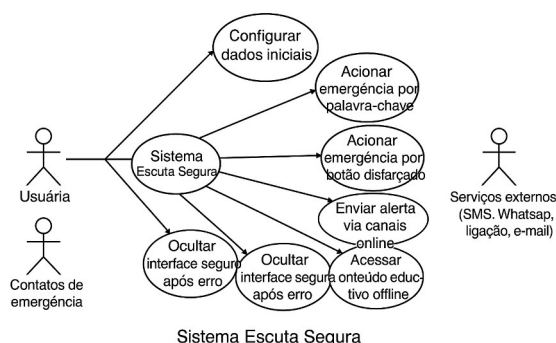


Figura 3. Diagrama de caso de uso do aplicativo Escuta Segura gerada pelo copilot.

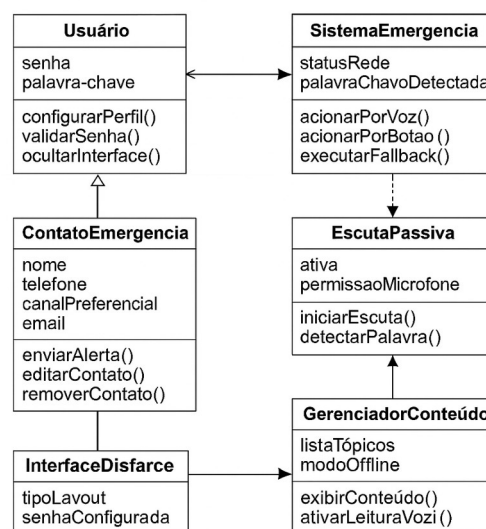


Figura 4. Diagrama de classes do aplicativo Escuta Segura gerada pelo copilot.

A IA também não criou o arquivo metro.config.js para o projeto, resultando em erros. Ao ser solicitado, o copilot informou o caminho para a resolução do problema, porém esta tentativa foi sem sucesso, foi necessário, então, apagar os arquivos do projeto e recriá-lo, comando a comando, conforme a sugestão do copilot. Após 3 tentativas ainda não tinha sido possível começar a instalação do projeto, pois as pastas android e ios não tinham sido criadas, então o powershell foi usado para apoio, mas também sem sucesso.

Posteriormente foi descoberto que uma das variáveis de ambiente não estava configurada, e que não estava com o ambiente preparado para execução da aplicação pela necessidade de instalação do java development kit (JDK). Depois disso ainda surgiram problemas para executar a aplicação, mas depois de criar um novo projeto com o expo, obteve-se sucesso na execução do projeto.

Para criação do ícone foi necessário o uso do copilot fora do vs code, pois no visual studio code não foi possível gerar uma imagem png para usar como ícone do aplicativo.

Durante a fase de implementação foi identificada uma lacuna na documentação do projeto original: a necessidade de integração da tela de disfarce (calculadora) quando os usuários já estão logados. Na implementação da configuração de palavra-chave inicial e senha, foi sugerido à IA a criação de um *onboarding*, que é um passo a passo para guiar o usuário no primeiro acesso ao aplicativo. Embora desvios do escopo inicial do projeto possam comprometer o contrato com cliente, no projeto foi tratado como mitigação de um ponto de vulnerabilidade e oportunidade de melhoria. A

metodologia de desenvolvimento ágil, entretanto, sanaria esse tipo de problema sem se configurar como uma quebra de contrato como o ocorrido na metodologia clássica, ou seja, um requisito não previsto teve que ser implementado ou adaptado de forma diferente do documentado na especificação de requisitos.

A IA do VS Code sugeriu a inserção de um mecanismo de camuflagem em todas as interfaces. Durante a implementação ocorreu erro no *onboarding*, na persistência da palavra chave para socorro num bloco try-catch na função de salvamento sugerida pela ferramenta de inteligência. Ao submeter o erro ao chat da VS, foi identificado a incompatibilidade pelo uso da biblioteca de funções CryptoJS que não estava disponível para o react native e o Expo, como solução a IA sugeriu a migração para a biblioteca expo-crypto, a substituição sanou o problema.

Ao solicitar à IA que fosse implementado o bloqueio da tela para o modo disfarce calculadora, observou-se uma falha lógica na solução proposta pelo copilot: o código fornecido não habilitou a interface de entrada para a senha configurada no *onboarding*. A solução sugerida, inicialmente, foi um detector de toque invisível, o que provou-se inviável por comprometer a usabilidade da aplicação. Após sugestões redundante e alucinações, como a criação de uma calculadora global já existente, foi necessária a intervenção humana. A sugestão da IA foi um gatilho (duplo clique na tela) para habilitar a digitação para bloqueio do modo disfarce, o que, de fato, funcionou.

Esses tipos de problemas indicam que um usuário leigo em computação teria grandes problemas em conduzir o desenvolvimento, o que era uma grande preocupação quando da popularização das IA geradoras de código fonte, ou seja, a colocação da profissão de programador em cheque devido ao eventual desenvolvimento de software por pessoas leigas em computação, o que seria catastrófico para o mercado de trabalho. Pelo contrário, o desenvolvimento deste projeto evidenciou de forma contundente que a IA precisa ser operada por um profissional da área, consciente das diversas etapas, que tenha condições técnicas de avaliar a qualidade de cada subproduto produzido pela IA. De fato, observou-se que a IA precisava de direcionamento para sua correta execução e prévio conhecimento de programação e das potencialidades dos frameworks e IDEs utilizados.

Para a escuta simulada utilizou-se inicialmente o framework (biblioteca de funções) expo-v para captura de dados, contudo, devido a limitações existentes como a impossibilidade da escuta contínua em segundo plano, e também o consumo de bateria consideravelmente alto, embora sugerido pelo copilot, a ferramenta omitiu restrições de interoperabilidade de

API's (interface de programas de aplicações que permite dois softwares conversarem entre si) nativas de reconhecimento de voz. Para refinar o envio automático da mensagem de emergência foi utilizado o react-native-voice, que resultou em conflitos de dependência que impediram o aplicativo de ser renderizado. Após intervenção assistida pelo copilot, para correção dos módulos, o sistema retomou a operabilidade desejada, embora apresentando caracteres em idioma estrangeiro, como visto na Figura 5.

Diante da necessidade da escuta passiva funcional, foi imperativo migrar o projeto para React Native CLI, descartando a arquitetura inicial (Expo) proposta pela IA.

O React Native CLI oferece maior controle e flexibilidade, permitindo a adição de bibliotecas nativas e a personalização do projeto, além de oferecer um desempenho mais rápido e melhorado. Já o Expo tem uma configuração inicial mais fácil e é mais fácil de usar para novos desenvolvedores, mas pode ter limitações em personalização.

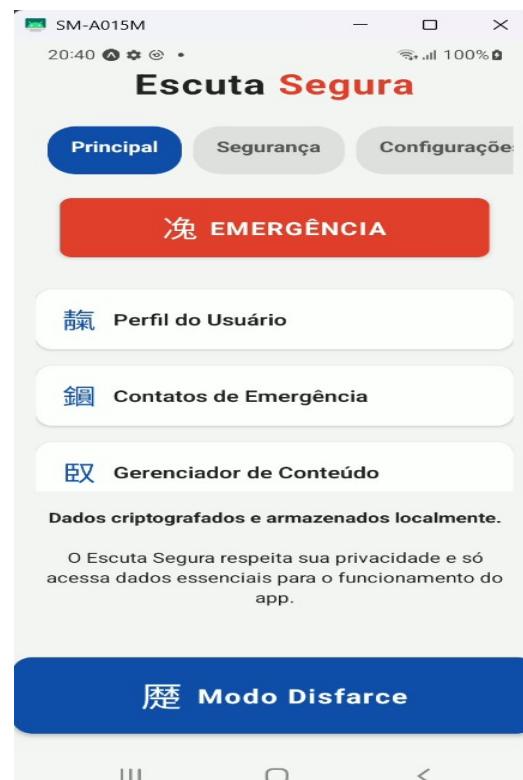


Figura 5. Tela gerada pelo copilot com caracteres estrangeiros.

Durante a transição do ecossistema Expo para o React Native Cli revelou-se que a biblioteca Microsoft Precise possuía restrições comerciais para a escuta passiva, motivando a migração para Porcupine (Picovoice) uma framework deep learning baseado em detecção de palavras chaves (*wake word*).

Por fim, enfrentou-se limitações do sistema Windows em relação (path length) que gerou falhas no processo de build do projeto. A tentativa de resolução automática do problema, via IA, trouxe corrupção da estrutura do projeto, exigindo a reconstrução manual do ambiente de desenvolvimento. Para tal, utilizou-se auxílio do Gemini, em caráter comparativo, o qual demonstrou maior eficiência na recuperação de arquivos em cenários em que o copilot apresentava respostas redundantes.

Por fim a implementação da escuta passiva foi concluída, com o *wake word* 'Socorro' via plataforma Picovoice, superando a descontinuidade do Precise e a complexidade desse ambiente.

As principais telas do aplicativo Escuta Segura são apresentadas nas figuras 6 à 16.

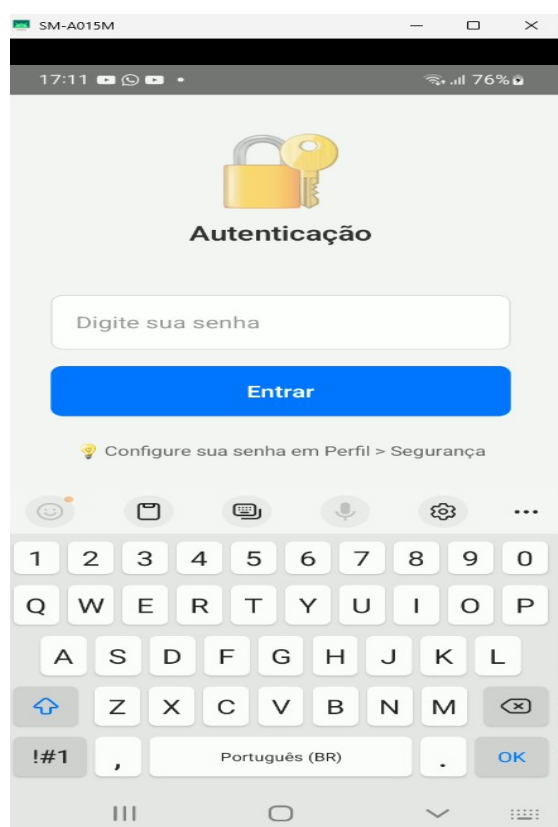


Figura 6. Tela de login do aplicativo Escuta Segura gerada pelo copilot.

O ciclo operacional do aplicativo começa na interface de autenticação (figura 6), que oferece dualidade com a interface disfarce calculadora (figura 7) configurada na tela inicial do aplicativo. No primeiro acesso o usuário é conduzido por um fluxo instrutivo através do *onboarding* (passo a passo explicativo). Uma vez autenticado o usuário é direcionado a tela inicial (figura 8), que centraliza as gestões de criar contatos de emergência, escolher canais de preferência de

notificação de emergência, editar informações de perfil, senha e palavra-chave.

A arquitetura também permite a gestão de consentimento de escuta passiva (figura 9) e ao perfil do usuário (figura 10). O módulo de configurações avançadas disponibiliza a interface de auditoria, exportação e registros de áudios capturados localmente (figura 11). Já a escuta passiva é controlada na tela exibida na figura 12, o acionamento de alertas socorro na tela exibida na figura 13 e a parametrização de opções de envio na figura 14. O gerenciamento de configurações de contatos de emergência (figura 15) e a opção de exclusão definitiva de perfil completam os recursos de controle e privacidade da solução.

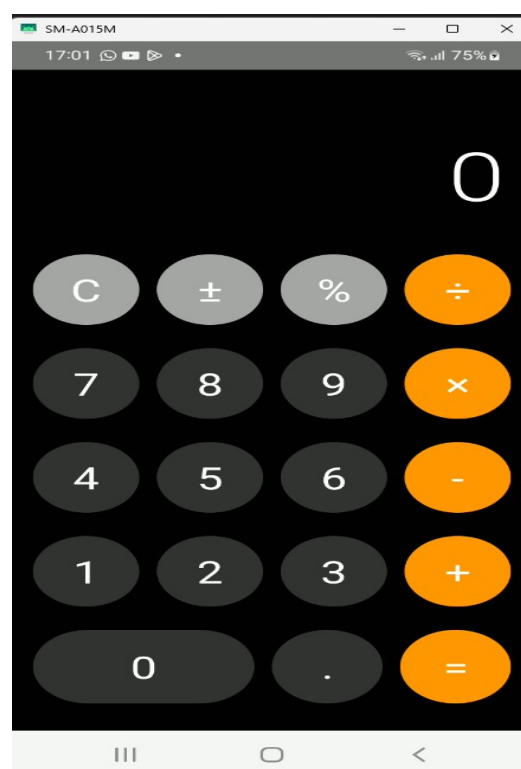


Figura 7. Tela de disfarce do aplicativo Escuta Segura gerada pelo copilot.



Figura 8. Tela principal do aplicativo.



Figura 10. Tela de perfil do usuário.

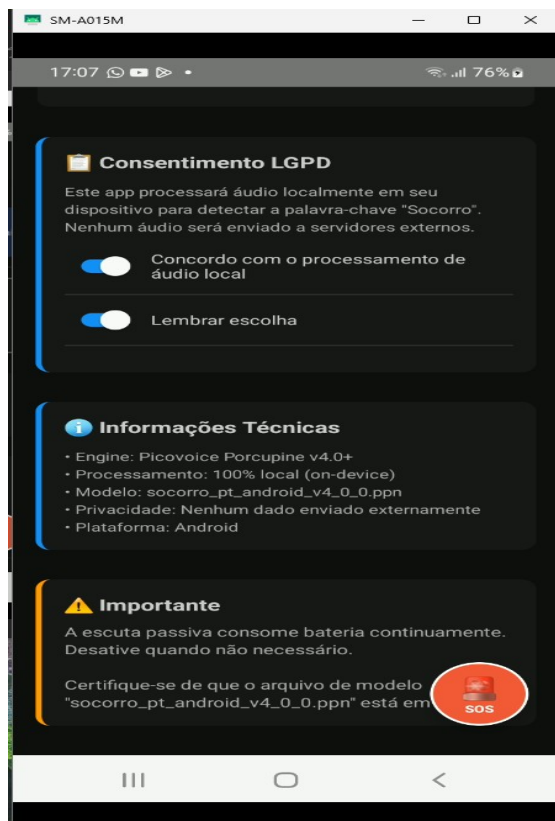


Figura 9. Tela de consentimento para habilitação de escuta passiva.



Figura 11. Tela de gerenciamento de opções de áudio.



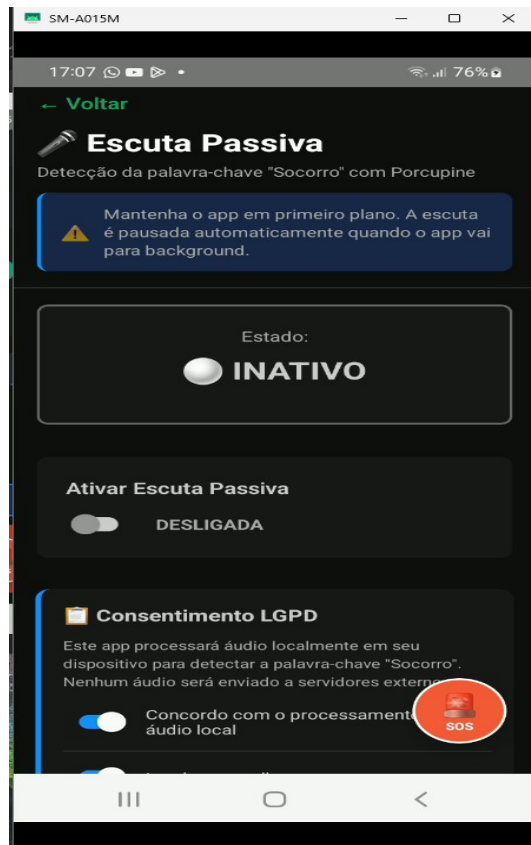


Figura 12. Tela para ativação de escuta passiva.



Figura 14. Tela de acionamento manual de emergência.

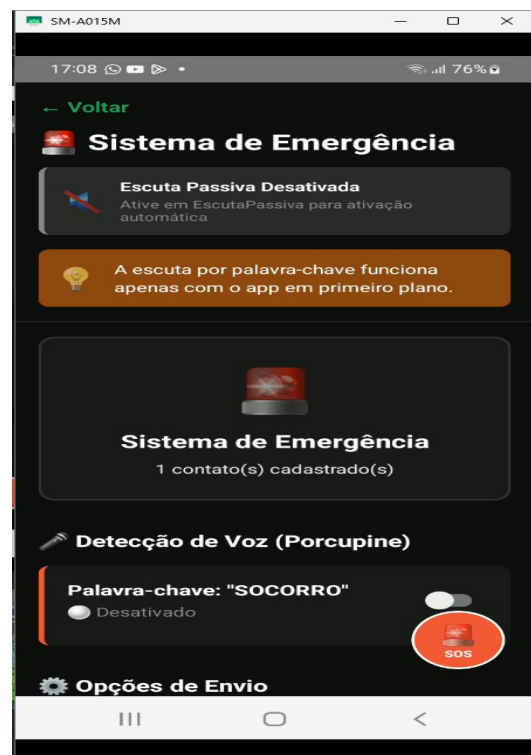


Figura 13. Tela de gestão de emergência.

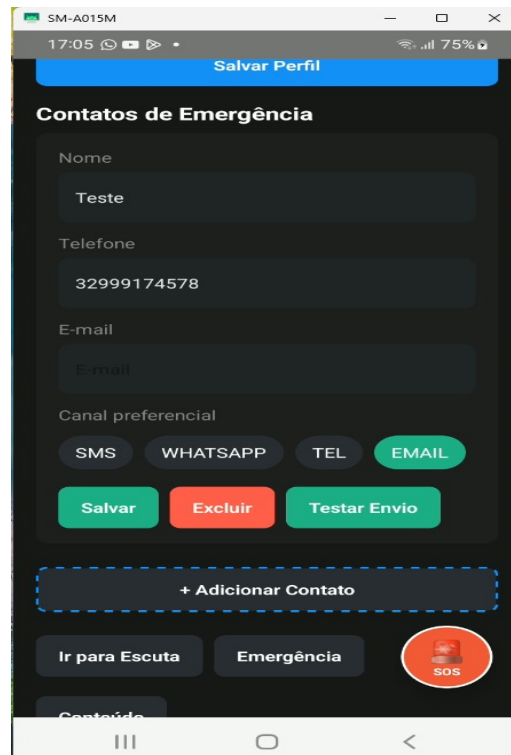


Figura 15. Tela de contatos de emergência.

A paleta de cores utilizada, bem como a requisição acessibilidade digital foram atendidas pela IA na

definição de telas do aplicativo, sendo necessários poucos ajustes.

Com respeito à persistência de dados, em concordância com a definição de requisitos não funcionais, estabeleceu-se que os dados do usuário não seriam enviados a terceiros, e adotou-se o *local-first* (persistência local), garantindo que as informações fossem processadas e armazenadas no dispositivo para futura perícia policial em casos de sinistro.

A implementação técnica utilizou o `asyncStorage` para dados estruturais e o `encryptedStorage` para dados sensíveis, configurados na entrada de dados através do primeiro acesso via *onboarding*. Os áudios são gravados localmente, e estão disponíveis para exportação e consulta posterior.

Caso a palavra-chave seja identificada pela escuta passiva, uma notificação é disparada seguindo a regra de preferência (WHATSAPP, SMS, Ligação e E-mail) com base nos contatos cadastrados, e mantendo a baixa dependência de serviços externos e a conformidade com os requisitos IEEE de privacidade do usuário (IEEE, 2023).

### CONCLUSÃO

Após todo o processo de desenvolvimento foi obtido um resultado satisfatório para o aplicativo pretendido. Como todo protótipo, e ainda por se tratar de um tema sensível, precisa ainda passar por extensa bateria de testes, avaliações e refinamentos até que se alcance um produto final seguro e pronto para uso.

A inteligência artificial se mostrou, no projeto em pauta, uma grande aliada ao desenvolvimento de software, podendo entretanto gerar subprodutos equivocados ou com falhas pontuais. O que torna fundamental a sua operação por um profissional capacitado, tornando-a por tanto, ao que parece, inadequada para usuários leigos em computação, especificamente em engenharia de software, ambientes de desenvolvimento, persistência de dados e linguagens de programação.

A evolução da inteligência artificial e da automação de processos básicos e repetitivos de desenvolvimento é benéfica, mas não substitui o pensamento humano e a expertise de um desenvolvedor de software para avaliar o desenvolvimento realizado, mas, de fato, o uso da IA trouxe benefícios e permitiu que diversas atividades fossem realizadas com maior agilidade.

Para evitar as "alucinações" da IA no projeto em pauta usou-se sempre o mesmo contexto das solicitações, evitando-se abrir um novo canal de prompts, pois a IA usa informações do contexto, o que economiza tokens e gera respostas mais precisas, também é indicado deixar explícito no comando via prompt caso não queira a modificação sem devida permissão, o que

poderia ferir a especificação de requisitos ou o contrato com o usuário/cliente.

Conforme o relatado neste trabalho, toda resposta da IA precisa ser cuidadosamente analisada. Um fenômeno comum, e que pode atrasar e incluir falhas ou ferir os requisitos definidos pelos clientes/usuários da aplicação é a inclusão de funcionalidades, layout, ou outros artefatos não solicitados pelo operador do prompt da IA. Outro problema grave é a alucinação, que pode indicar caminhos ou soluções equivocadas, ou mesmo, pela não consciência do todo, optar por caminhos/soluções que não atendam às futuras requisições, como ocorreu neste projeto quando a IA optou pelo framework Expo em detrimento do React Native CLI, promovendo uma mudança na estrutura do projeto nem sempre fácil de ser realizada. Todos esses problemas, de fato, aconteceram no projeto em questão.

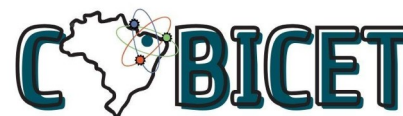
Tais problemas podem se agravar a depender da complexidade ou dimensão do problema, ou mesmo da diversidade ou quantidade de membros de equipes de desenvolvimento, podendo se tornar um desafio logístico e metodológico, principalmente no que envolve o contexto de conversação com a IA e a forma de interação, seja amígdala ou de forma mais densa.

De qualquer forma, a IA demonstrou, pelo menos neste projeto, ser uma ferramenta poderosa nas mãos do profissional desenvolvedor experiente, apesar de todos os problemas vividos e relatados, encurtando, sobremaneira, o processo de desenvolvimento, trazendo boas perspectivas para o processo de desenvolvimento de software, mas trazendo também preocupações quanto ao mercado de trabalho, principalmente para os recém formados, e também nos desafios para seu uso em projetos complexos, onde deve ser canalizada para a realização de tarefas simples e pontuais, embutidas em IDEs, dado o desafio logístico/metodológico de seu uso em amplo sentido, ou seja, abarcando a totalidade do sistema.

O aplicativo, em si, teve sua idealização com o intuito de resolução de um problema social, popular e relevante, que atinge muitas mulheres no âmbito do seio familiar, lugar que deveria ser ambiente de amparo e segurança, mas muitas das vezes o oposto é tragicamente vivenciado. Contudo, acredita-se que o produto desenvolvido neste estudo possa trazer um apoio tecnológico para a problemática da violência doméstica, e que possa ser extensivamente testado, e aperfeiçoado para que, num futuro próximo, seja utilizado para o fim a que foi proposto, que o benefício da sociedade, em específico o público feminino.

### AGRADECIMENTOS

Agradeço em primeiro lugar a Deus, autor e consumidor da fé, bem como à minha mãe Rozângela, mulher lutadora, forte e incansável na busca por uma vida de qualidade para seus filhos, e ao meu irmão



Daniilo, parceiro nas lutas da vida. Ao meu orientador Professor Sérgio que defendeu a ideia deste projeto e me orientou com maestria, ao Instituto Federal de Educação, Ciência e Tecnologia do Sudeste de Minas Gerais que me aceitou como aluna e me trouxe até aqui, com ensino de qualidade e gratuito.

## REFERÊNCIAS

BARROS, C. J. Fintech desiste de IA para atendimento a clientes e contrata “uberizados”. **UOL**, 2025. Disponível em: <https://economia.uol.com.br/colunas/carlos-juliano-barros/2025/06/03/fintech-desiste-de-ia-para-atendimento-a-clientes-e-contrata-uberizados.htm>. Acesso em: 5 jun. 2025.

BRASIL. [Lei nº 13.146, de 6 de julho de 2015]. **Lei Brasileira de Inclusão da Pessoa com Deficiência (Estatuto da Pessoa com Deficiência)**. Brasília, DF: Presidência da República, [2015]. Disponível em: [https://www.planalto.gov.br/ccivil\\_03/\\_ato2015-2018/2015/lei/113146.htm](https://www.planalto.gov.br/ccivil_03/_ato2015-2018/2015/lei/113146.htm). Acesso em: 10 jun. 2025.

BRASIL. [Lei nº 11.340, de 7 de agosto de 2006]. **Cria mecanismos para coibir a violência doméstica e familiar contra a mulher (Lei Maria da Penha)**. Brasília, DF: Presidência da República, [2006]. Disponível em: [https://www.planalto.gov.br/ccivil\\_03/\\_Ato2004-2006/2006/Lei/L11340.htm](https://www.planalto.gov.br/ccivil_03/_Ato2004-2006/2006/Lei/L11340.htm). Acesso em: 4 jun. 2025.

CDB. **Investidores Miram Startups de IA para Programadores**. Correio do Brasil, 2025. Disponível em: <https://correiodobrasil.com.br/a/investidores-miram-startups-ia-programadores>. Acesso em: 4 jun. 2025.

G1. **214 milhões de brasileiras sofreram algum tipo de violência nos últimos 12 meses, diz pesquisa**. Rio de Janeiro, 10 mar. 2025. Disponível em: <https://g1.globo.com/politica/noticia/2025/03/10/214-milhoes-de-brasileiras-sofreram-algum-tipo-de-violencia-nos-ultimos-12-meses-diz-pesquisa.ghtml>. Acesso em: 22 mar. 2026.

IEEE. **Software Requirements Specifications**. Computer Society, 2025. Disponível em: <https://www.computer.org/resources/software-requirements-specifications>. Acesso em: 10 jun. 2025.

IEEE. Alcançando expectativas de privacidade no mundo digital. [S.l.]: IEEE, 2023. Disponível em: <https://digitalprivacy.ieee.org/wp-content/uploads/2025/05/ieee-digital-privacy-model-portuguese.pdf>. Acesso em: 24 mar. 2026.

JUSBRASIL. **Lei Maria da penha e os 5 tipos de violência que ela ampara**. 2025. Disponível em: <https://www.jusbrasil.com.br/artigos/lei-maria-da-penha-e-os-5-tipos-de-violencia-que-ela-ampara/1954622922>. Acesso em: 4 jun. 2025.

MANUSCRIPT Templates for Conference Proceedings. **IEEE**, 2026. Disponível em: <https://www.ieee.org/conferences/publishing/templates>. Acesso em: 3 mar. 2026.

MDN WEB DOCS. **ARIA**. Mozilla, 2026. Disponível em: <https://developer.mozilla.org/pt-BR/docs/Web/Accessibility/ARIA>. Acesso em: 15 mar. 2026.

PEDE, M. **Mulher pede pizza para denunciar violência doméstica no DF**. G1, 2025. Disponível em: <https://g1.globo.com/df/distrito-federal/noticia/2025/06/02/mulher-pede-pizza-para-denunciar-violencia-domestica-no-df.ghtml>. Acesso em: 4 jun. 2025.

PRESSMAN, Roger S.; MAXIM, Bruce R. **Engenharia de software**. Porto Alegre: AMGH, 2021.

ROOSE, K. For Some Recent Graduates, the A.I. Job Apocalypse May Already Be Here. **The New York Times**, 30 maio 2025. Disponível em: <https://dicionario.priberam.org/no%20que%20diz%20respeito%20a>. Acesso em: [Data de acesso].

SAKAMOTO, L. Um em cada 4 empregos corre risco por causa da IA. Veja como está o seu. **UOL**, 2025. Disponível em: <https://noticias.uol.com.br/colunas/leonardo-sakamoto/2025/05/20/um-em-cada-4-empregos-corre-risco-por-causa-da-ia-veja-como-esta-o-seu.htm>. Acesso em: 3 jun. 2025.

SOFTWARE. **Qual é o papel da inteligência artificial na engenharia de software?** Programaê, 2025. Disponível em: <https://programae.org.br/software/qual-e-o-papel-da-inteligencia-artificial-na-engenharia-de-software/>. Acesso em: 5 jun. 2025.