

AgileMood: A Tool for Ongoing Measurement of Psychological Safety in Agile Teams

Arthur M. B. Melo

arthur.macedo.bandeira@gmail.com
Universidade Federal Rural de
Pernambuco
Recife, Pernambuco, Brasil

Marcelo Marinho

marcelo.marinho@ufrpe.br
Universidade Federal Rural de
Pernambuco
Recife, Pernambuco, Brasil

Suzana Sampaio

suzana.sampaio@ufrpe.br
Universidade Federal Rural de
Pernambuco
Recife, Pernambuco, Brasil

ABSTRACT

Psychological safety and individual emotions have proven essential for the effective functioning of agile teams, directly impacting collaboration, collective performance and ultimately, software quality. This paper presents *AgileMood*, a tool designed to diagnose and enhance psychological safety and emotional awareness in agile teams through the collection of periodic anonymous feedback, the calculation of specific metrics, and the generation of dynamic visualizations. The solution not only measures the overall perceived level of safety and emotions but also detects internal variations among team members, enabling data-driven interventions. *AgileMood* distinguishes itself from existing approaches by incorporating well-established theoretical foundations, preserving anonymity, and offering practical support for decision-making by team leaders. The empirical evaluation of the tool, conducted through a case study involving 41 students (7 teams of 5 to 8 team members), utilized the System Usability Scale (SUS) and the Technology Acceptance Model (TAM), demonstrating promising usability and high perceived usefulness. Available as open-source software, the tool facilitates replicability and further development by the community. As a contribution, this work proposes a practice-oriented artifact with the potential to transform psychological safety and emotions into a measurable and manageable element within agile software development teams, thereby fostering improved software quality. Future studies should empirically evaluate its application in real organizational contexts.

KEYWORDS

Psychological safety, Emotions, Agile Teams, Software Engineering, Tools

1 Introdução

A segurança psicológica em equipes refere-se à crença compartilhada de que o ambiente de trabalho é seguro para riscos inter-pessoais, permitindo que os membros da equipe expressem ideias, admitam erros e façam perguntas sem medo de reprimenda [18]. Essa condição foi identificada como essencial para o desempenho, inovação e aprendizado em grupo [13, 18].

Em contextos de desenvolvimento ágil de software, caracterizados por adaptação constante e colaboração intensiva [35, 36], a segurança psicológica torna-se ainda mais crucial para viabilizar comunicação aberta e experimentação [27, 33, 49]. Estudos mostram que equipes ágeis com maior segurança psicológica apresentam maior reflexividade, resolução de problemas complexos e capacidade de aprendizagem organizacional [11]. Além disso, a segurança

psicológica nas equipes ágeis é essencial para a qualidade do software [5], pois promove uma cultura de excelência técnica através de práticas de compartilhamento de conhecimento e uma cultura de feedback contínuo [3–5].

Entretanto, para além da segurança psicológica, as emoções desempenham um papel significativo no ambiente de desenvolvimento de software. Salido *et al.* [40] concluíram que atividades de projetos ágeis como tomada de decisão, estimativa de esforço, implementação de histórias de usuário, codificação e testes são fortemente influenciadas por fatores emocionais. Problemas emocionais podem levar à aversão a mudanças significativas no código ou nos requisitos [32], e afetar o processo de tomada de decisão durante o desenvolvimento ágil [12]. Por exemplo, quando desenvolvedores experimentam frustração, estresse ou raiva ao enfrentar problemas complexos, esses estados podem diminuir seu desempenho [15, 24, 38]. Ademais, as emoções impactam a interação e a colaboração entre os membros da equipe [30], sendo a coesão e a confiança dependentes da capacidade de gerenciar emoções individuais e coletivas [30, 41, 47]. Quando os membros da equipe estão cientes de suas próprias emoções, eles podem se aproximar de seus pares e fornecer apoio, criando consequentemente mais confiança na equipe [30]. Por outro lado, níveis baixos de segurança psicológica estão associados a menos trocas de conhecimento e pior desempenho do time [21], o que pode impactar diretamente a qualidade e a sustentabilidade das soluções de software.

A conexão entre segurança psicológica e emoções não é apenas conceitual, mas sim um elo bidirecional fundamental para o funcionamento das equipes. As emoções surgem a partir de como os indivíduos avaliam e navegam nos eventos situacionais do ambiente de trabalho [8, 14]. Nesse contexto, a literatura indica que certas práticas de gestão geram emoções negativas que, por sua vez, contribuem para um ambiente de baixa segurança psicológica. Por exemplo, o estudo de Weibel *et al.* [48] demonstra que o excesso de controle e de supervisão pode causar sentimentos de estresse, desrespeito e cinismo, que afetam diretamente a segurança psicológica [28]. De forma recíproca, a baixa segurança psicológica percebida resulta em diferentes respostas emocionais dos funcionários. A presença de confiança e autonomia no ambiente de trabalho evoca um estado emocional positivo de conforto e alegria [28], criando um ciclo virtuoso. Além disso, a segurança psicológica atua como um fator que mitiga o efeito negativo da riqueza socioemocional na integração comportamental [45], fortalecendo a coesão e a colaboração da equipe. Portanto, a gestão eficaz das emoções e a construção de um ambiente psicologicamente seguro são aspectos relacionados, o que justifica a abordagem conjunta destes temas em nossa pesquisa.

Apesar do reconhecimento da importância da segurança psicológica e da influência das emoções no desempenho e na qualidade do software, a gestão eficaz desses aspectos ainda enfrenta diversos desafios [5, 28]. A natureza subjetiva e frequentemente não verbal das emoções, aliada à dificuldade de mensurar o clima psicológico de forma contínua, cria lacunas significativas na capacidade das equipes de identificar e abordar problemas proativamente [28, 48].

Outro desafio relevante é a ausência de monitoramento contínuo. A segurança psicológica costuma ser avaliada esporadicamente, por meio de pesquisas pontuais ou retrospectivas, o que inviabiliza um acompanhamento dinâmico e em tempo real. Sem essa vigilância constante, gestores podem não identificar declínios ou problemas no clima organizacional até que seus efeitos negativos já estejam consolidados.

Além disso, as percepções heterogêneas dentro da equipe configuram um fator de risco importante. Quando os membros de um mesmo time têm visões muito divergentes sobre o quanto seguro é o ambiente, o desempenho coletivo tende a ser comprometido [22]. Estudos recentes demonstram que essa dispersão de percepções, também chamada de “falta de força climática”, prejudica o engajamento e os resultados da equipe, evidenciando a necessidade de alinhar percepções internas [22].

Também se destaca a dificuldade de fomentar confiança em ambientes ágeis. Líderes e organizações relatam obstáculos para criar um espaço psicologicamente seguro, sobretudo em equipes ágeis distribuídas, nas quais a comunicação fluida e a construção de confiança são ainda mais desafiadoras [44]. Na ausência de ferramentas apropriadas, os esforços para aprimorar a segurança psicológica dependem apenas de intervenções humanas e de práticas informais.

Por fim, observa-se a carência de ferramentas integradas que permitam avaliar e promover a segurança psicológica de maneira sistemática e alinhada ao fluxo de trabalho ágil. Ainda que seja reconhecido que a segurança psicológica impacta diretamente a qualidade do software e a capacidade das equipes de aprender com os erros e adaptar-se rapidamente [4], são poucas as organizações que dispõem de mecanismos estruturados para mensurar e desenvolver esse aspecto cultural no cotidiano [4].

O *AgileMood* foi concebido para coletar feedbacks anônimos dos membros da equipe de forma regular e contínua, possibilitando a quantificação da segurança psicológica ao longo do tempo. Além disso, a ferramenta busca identificar discrepâncias nas percepções dentro da equipe, fornecendo indicadores de dispersão que sinalizam possíveis desalinhamentos no clima psicológico. Por fim, o *AgileMood* também tem a função de facilitar intervenções e reflexões da equipe, uma vez que, ao revelar problemas relacionados à segurança psicológica, incentiva discussões durante as retrospectivas e orienta a definição de ações corretivas.

Diante desses desafios, nosso trabalho busca responder às seguintes perguntas (P) de pesquisa: P1: *AgileMood é percebida como fácil de usar nas equipes ágeis de software?* e P2: *AgileMood é percebida como útil para apoiar a Segurança Psicológica nas equipes ágeis de software?*

Este artigo tem como objetivo apresentar a ferramenta *AgileMood*, incluindo sua tecnologia e principais funcionalidades, e fornecer uma avaliação empírica da ferramenta com equipes de desenvolvimento de software ágeis. Com essa proposta, esperamos contribuir para tornar a segurança psicológica um aspecto visível e

gerenciável nos times ágeis, auxiliando líderes e membros de equipe a construir um ambiente de trabalho mais seguro, confiável e propício ao aprendizado contínuo.

Além desta introdução, a Seção 2 apresenta a fundamentação teórica que serve de base para este trabalho. A Seção 3 apresenta as etapas seguidas para o desenvolvimento da ferramenta. A Seção 4 descreve em detalhes o desenvolvimento da ferramenta proposta, assim como sua avaliação empírica, realizada por meio de um estudo de caso com estudantes de graduação do curso de Computação. Por fim, a Seção 5 fornece reflexões quanto ameaças à validades, algumas conclusões, contribuições do trabalho, expõe algumas limitações e as oportunidades de trabalhos futuros.

2 Referencial Teórico

Esta seção sintetiza a base teórica sobre o impacto de fatores humanos e sociais na engenharia de software ágil, abordando a influência das emoções na qualidade do software, a segurança psicológica em equipes de alto desempenho e as ferramentas de monitoramento, com suas lacunas que motivam este estudo.

2.1 O Papel das Emoções na Qualidade de Software

No desenvolvimento ágil de software, os estados emocionais dos engenheiros de software exercem uma influência substancial tanto em seu desempenho quanto na qualidade do produto desenvolvido [40]. Pesquisas apontam que a compreensão dos problemas e o trabalho em equipe, aspectos cruciais do desenvolvimento ágil, são moldados pelas emoções vivenciadas [25, 40]. Quando os membros de equipes ágeis possuem consciência de seus próprios estados emocionais, eles são mais capazes de autorregular-se, mantendo níveis desejáveis de desempenho, melhorando a comunicação e a colaboração, e consequentemente, elevando a qualidade do software [15, 40].

A relação entre emoções e qualidade é bidirecional. Emoções negativas, como infelicidade ou raiva, podem resultar diretamente em baixa qualidade de código, manifestando-se em código desorganizado, propenso a erros ou até mesmo na exclusão intencional de partes do projeto, levando a falhas de comunicação e atrasos na entrega [34, 40]. Já a felicidade e emoções positivas estão ligadas a uma redução no tempo de correção de problemas (*bugs*) e à melhora do desempenho cognitivo, o que sublinha a importância do bem-estar para a qualidade do código [24, 25, 34, 40].

Além do impacto individual, as emoções moldam a dinâmica da equipe. Em equipes ágeis, a segurança psicológica desempenha um papel fundamental ao criar um ambiente onde as emoções podem ser expressas e gerenciadas de forma saudável, o que é crucial para a coesão, a confiança e a adaptabilidade da equipe [30, 41, 47]. Diante dessa forte influência, a inclusão dos estados emocionais na gestão de projetos ágeis pode beneficiar tanto os engenheiros de software quanto suas equipes, culminando em melhor desempenho e qualidade de software [40].

2.2 Segurança Psicológica em Equipes Ágeis

Conforme definido por Edmondson [18], segurança psicológica é a crença compartilhada de que o ambiente é seguro para assumir riscos interpessoais. Esse conceito é especialmente relevante em

metodologias ágeis, como Scrum e Kanban, que dependem fortemente de colaboração e comunicação contínua entre os membros da equipe [11, 33].

Diversos estudos reforçam a importância da segurança psicológica no contexto ágil. Baer e Frese [6] indicam que equipes com altos níveis dessa segurança tendem a apresentar maior capacidade de adaptação frente a mudanças, uma característica essencial em ambientes dinâmicos como os de desenvolvimento de software. Complementando essa perspectiva, Breevaart et al. [9] mostram que a segurança psicológica contribui significativamente para o aumento da satisfação e do engajamento dos colaboradores, além de estar associada à redução da rotatividade.

Além disso, Alami et al. [4] demonstram que a presença de segurança psicológica está diretamente relacionada à qualidade do software entregue e à capacidade reflexiva das equipes. No entanto, esse ambiente de segurança pode ser fragilizado por barreiras culturais e dificuldades de comunicação, especialmente em equipes distribuídas geograficamente [44].

Nesse sentido, Fyhn et al. [22] destacam a importância da uniformidade das percepções internas para maximizar os efeitos positivos da segurança psicológica. De forma alinhada, Lenberg e Feldt [29] observam que a percepção compartilhada de segurança psicológica, aliada à definição clara de normas e padrões do grupo, exerce influência direta tanto na produtividade quanto no bem-estar dos profissionais em equipes ágeis de desenvolvimento.

2.3 Ferramentas e Abordagens de Monitoramento Existentes

Embora a literatura reconheça a relevância da segurança psicológica e das emoções para o desempenho e a qualidade em equipes ágeis, as estratégias correntes de monitoramento ainda apresentam alcance limitado.

O *Emotional Dashboard* [43] propõe o monitoramento das emoções de desenvolvedores através de postagens em redes sociais, visando identificar períodos emocionais que impactam o desenvolvimento. Contudo, suas limitações incluem a dependência de dados de mídias sociais e a ausência de validação em ambientes reais de desenvolvimento. Já o *EmoRefex* [31], protótipo de plataforma baseada em inteligência artificial para geração de insights emocionais, apoia desenvolvedores no autorrelato de estados afetivos e gestores na obtenção de recomendações voltadas ao bem-estar das equipes. Entre os desafios reportados destacam-se a necessidade de validação em cenários de uso reais (para além de estudos de levantamento) e a suscetibilidade a vieses na inferência emocional por modelos de IA generalistas (e.g., ChatGPT). Por sua vez, o *EmoVizPhy* [23] utiliza a visualização de emoções auto-relatadas e dados biométricos, para auxiliar na autorreflexão do desenvolvedor. Porém, suas limitações incluem o pequeno número e a natureza dos participantes (estudantes), além da intrusão e complexidade inerentes ao uso de sensores fisiológicos.

O *Emotimonitor* [2] é uma ferramenta reportada na literatura que inspirou parte deste trabalho. Trata-se de uma extensão para o Trello¹, focada em capturar as emoções de membros de equipes ágeis através de reações com emojis em cartões de tarefas. Ele

fornecer aos líderes um painel com visualizações das emoções reportadas durante o desenvolvimento.

Além disso, existem práticas difundidas em equipes ágeis para avaliar informalmente o “humor” ou a “saúde” da equipe. Por exemplo, a técnica do *Team Mood* ou *Mood Marbles* [1] utiliza votações anônimas com cartinhas ou bolinhas coloridas em retrospectivas para que os membros indiquem como se sentiram durante a interação. Há também ferramentas comerciais que implementam esses *health checks* (verificações de saúde) de equipe de forma online, medindo aspectos diversos (comunicação, alinhamento, felicidade no trabalho, etc.).

A mensuração contínua do estado emocional das equipes enfrenta obstáculos significativos, dada a natureza subjetiva e multifacetada das emoções. Por serem complexas e frequentemente não verbais, essas manifestações são difíceis de capturar de forma consistente [26]. A ausência de mecanismos estruturados para seu reconhecimento e monitoramento impede que as equipes compreendam o impacto das emoções sobre o desempenho individual e coletivo, bem como sobre a qualidade do software [43].

Diante dessas lacunas, observa-se a carência de ferramentas integradas que permitam avaliar e promover a segurança psicológica e o reconhecimento emocional de maneira sistemática e alinhada ao fluxo de desenvolvimento ágil. Embora o impacto desses fatores na qualidade do software e na capacidade de aprendizagem das equipes seja amplamente reconhecido, poucas organizações dispõem de mecanismos estruturados para mensurar e desenvolver esses aspectos culturais e emocionais no cotidiano [4]. A dependência de intervenções humanas e práticas informais, na ausência de ferramentas apropriadas, limita a proatividade na gestão do ambiente psicologicamente seguro e emocionalmente consciente.

3 Metodologia

Este trabalho foi conduzido em três etapas metodológicas complementares. A primeira consistiu em uma revisão exploratória da literatura, com o objetivo de embasar teoricamente o desenvolvimento da proposta. A segunda etapa envolveu a concepção e o desenvolvimento da ferramenta *AgileMood*. Por fim, a terceira etapa correspondeu à sua avaliação empírica, realizada por meio de um estudo de caso de natureza exploratória e avaliativa. A seguir, apresentam-se os detalhes de cada etapa da pesquisa.

3.1 Revisão Exploratória

Revisões exploratórias são particularmente úteis para mapear conceitos-chave, identificar lacunas de conhecimento e integrar perspectivas fragmentadas, além de contribuírem para o refinamento das perguntas de pesquisa e para a construção de uma base conceitual sólida [42].

Com esse propósito, procedeu-se à identificação dos principais construtos teóricos relacionados à segurança psicológica e às emoções em ambientes de desenvolvimento ágil, etapa que sustentou a definição dos requisitos iniciais da ferramenta proposta e viabilizou a estruturação conceitual do *AgileMood*. Na fase de concepção, esses requisitos foram fundamentados na literatura sobre emoções e segurança psicológica, bem como em estudos específicos que examinam a presença da segurança psicológica em equipes de desenvolvimento ágil de software; adicionalmente, foram consultados cinco

¹<https://trello.com/>

membros de equipes ágeis distintas e dois especialistas com múltiplas certificações ágeis. Reconhece-se, contudo, como limitação, a inexistência de participação direta de psicólogos profissionais no desenho inicial da solução.

Paralelamente, realizou-se uma análise comparativa estruturada por um processo de *benchmarking*, ancorado na revisão exploratória, por meio da qual foram identificadas ferramentas como *Emotional Dashboard*, *EmoReflex*, *EmoVizPhy* e *Emotimonitor*, além de práticas informais como o *Team Mood*. O exame considerou três dimensões: (i) forma de coleta (direta/anônima ou indireta/intrusiva), (ii) escopo da análise (humor pontual versus construtos mais robustos, como segurança psicológica) e (iii) periodicidade (medições contínuas ou pontuais). Observou-se que nenhuma das soluções analisadas contemplava, de modo integrado, essas dimensões, o que reforçou a necessidade de desenvolvimento do *AgileMood*.

3.2 Concepção da Ferramenta

Na segunda etapa do estudo, foram realizadas a concepção, o desenvolvimento e os testes da ferramenta *AgileMood*. Essa aplicação foi arquitetada como uma solução *web modular*, com o objetivo de permitir integração facilitada em distintos ambientes de desenvolvimento. A Figura 1 apresenta uma visão geral dos principais componentes da ferramenta, os quais são detalhados a seguir.

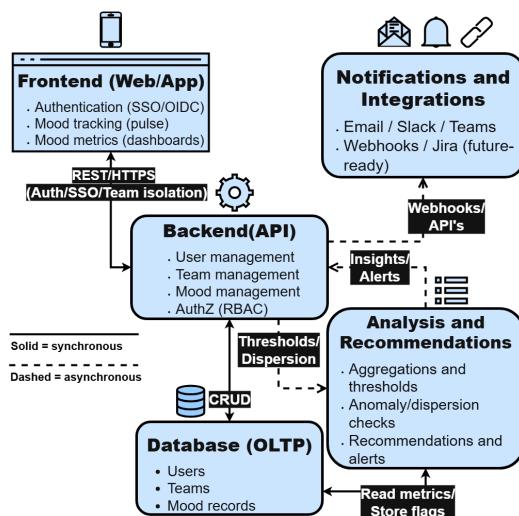


Figure 1: Principais componentes da ferramenta

A **interface do usuário (frontend)** foi implementada como uma aplicação web responsiva, acessível tanto em desktops quanto em dispositivos móveis. Essa interface constitui o principal ponto de interação entre os usuários e o sistema, permitindo que os membros da equipe respondam aos questionários e que os líderes visualizem os resultados consolidados. A interface foi projetada com foco na simplicidade e clareza, a fim de incentivar altas taxas de resposta e facilitar a interpretação das métricas geradas.

No que se refere ao **backend e ao banco de dados**, a arquitetura contempla um servidor central, desenvolvido em *Node.js* ou *Python Flask*, que expõe APIs REST seguras. Essas APIs são responsáveis

por receber as respostas aos questionários e armazená-las em um banco de dados, de forma agregada por equipe e por Sprint, sem identificação individual dos participantes. Além disso, o *backend* executa os cálculos estatísticos, como médias e desvios-padrão, ao final de cada rodada de feedback, gerando os dados que alimentam o *dashboard*.

O **módulo de análise e recomendações** é responsável por interpretar as métricas coletadas e aplicar regras pré-definidas para a geração de *insights* (percepções). Por exemplo, caso a pontuação média fique abaixo de um determinado limiar, ou a dispersão dos dados ultrapasse certo valor, o módulo sinaliza essas ocorrências, destacando-as visualmente no *dashboard*. Complementarmente, o sistema pode apresentar mensagens de recomendação previamente definidas, alinhadas aos problemas identificados como, por exemplo: “Trabalhar a cultura de *blameless post-mortem*², quando há medo de admitir erros”.

Com o intuito de reduzir barreiras à adoção, o *AgileMood* oferece recursos de **integração com ferramentas ágeis** já utilizadas pelas equipes. Por meio de *webhooks* e APIs, é possível conectá-lo a plataformas como *Slack* ou *Microsoft Teams*, viabilizando o envio automático de convites e lembretes de resposta dos questionários.

A **segurança e a autenticação** também foram consideradas na arquitetura da ferramenta. Cada equipe possui um espaço reservado no sistema, acessível apenas por seus membros mediante login. Para isso, são utilizados mecanismos padrão de autenticação, como *OAuth2*, *SAML* ou similares, com possibilidade de integração com sistemas de *Single Sign-On* corporativos. Toda a comunicação entre cliente e servidor é protegida por criptografia (HTTPS), garantindo a privacidade e a integridade das informações transmitidas.

Por fim, a adoção de uma arquitetura modular proporciona ao *AgileMood* elevada escalabilidade e potencial de evolução. Isso permite, por exemplo, a incorporação de novos módulos analíticos baseados em aprendizado de máquina para detecção de padrões comportamentais, sem a necessidade de reestruturação do núcleo da aplicação.

3.3 Estudo de Caso

A terceira etapa corresponde à avaliação empírica da ferramenta por meio de um estudo de caso exploratório e avaliativo [39], conduzido no contexto de cursos de graduação em Computação.

Participaram da investigação sete equipes de estudantes vinculadas às disciplinas do terceiro ano, sendo três equipes de “Projeto” (Desenvolvimento de Projetos de Software) e quatro de Engenharia de Software. Os detalhes das equipes podem ser verificados na Tabela 1. As equipes utilizaram a ferramenta ao longo de duas sprints (15 dias cada) em seus respectivos projetos acadêmicos, em condições naturais de uso, respeitando a dinâmica das disciplinas.

A coleta e o uso de dados que envolvem as percepções e emoções dos desenvolvedores, como no caso do *AgileMood*, requerem atenção rigorosa aos aspectos éticos e de privacidade. No desenvolvimento e estudo preliminar desta ferramenta, tomamos medidas para garantir a conduta ética e a proteção dos participantes. A participação foi voluntária, com os estudantes-desenvolvedores informados sobre a natureza do estudo. Além disso, a coleta de feedback foi realizada de forma anônima, assegurando que as respostas individuais não

²Prática do SRE/DevOps para análise de incidentes sem culpabilização individual.

Table 1: Características das equipes analisadas no estudo de caso

Time	Descrição do Projeto	Qty.	Papeis
Track&Save ¹	Sistema inteligente para apoio na compra de equipamentos para montagem de computadores voltados a jogos, utilizando recursos de inteligência artificial.	8	Scrum Master (1); Desenvolvedor back-end (3); Desenvolvedor front-end (2); Desenvolvedor fullstack (1); Engenheiro de Teste (1).
UniHome ¹	Aplicativo para auxiliar estudantes na busca por moradias próximas às universidades, considerando restrições individuais, com funcionalidade de cadastro de imóveis por locadores.	6	Scrum Master (1); Desenvolvedor Back-end (2); Desenvolvedor Front-End (2); Engenheiro de Teste (1).
DentiSig ¹	Plataforma para agendamento de consultas odontológicas e gestão integrada do prontuário eletrônico, com acesso compartilhado entre dentistas e pacientes.	7	Scrum Master (1); Desenvolvedor Back-end (2); Desenvolvedor Front-End (2); Desenvolvedor fullstack (1); Engenheiro de Teste (1).
Leia+ ²	Sistema web para escolas de ensino fundamental, focado na gestão do aprendizado com paradidáticos, promovendo acessibilidade aos livros e incentivando a leitura por meio de gamificação.	5	Scrum Master (1); Desenvolvedor full-stack e testador (4).
Nivela ²	Aplicativo para alunos do ensino médio que avalia o nível de aprendizagem por matéria e recomenda conteúdos para foco nos estudos.	5	Desenvolvedor full-stack e testador (5) com liderança alternada.
SigAlunos ²	Sistema para monitorar o desempenho acadêmico dos alunos com base em indicadores como frequência, notas e participação, utilizando IA preditiva para prever notas futuras e gerar rankings de desempenho	5	Desenvolvedor full-stack e testador (5) com liderança alternada.
AnalyticsFlow ²	Ferramenta de analytics e feedback contínuo, baseada em técnicas de IA para apoio na correção de atividades os estudantes.	5	Desenvolvedor full-stack e testador (5) com liderança alternada.

¹Equipes da Disciplina de “Projeto”²Equipes da Disciplina de Engenharia de Software

pudesse ser rastreadas até um participante específico, um fator crucial para promover a liberdade de expressão e reduzir a pressão por feedback positivo ou negativo.

Ao término desse período, foram aplicados dois instrumentos consolidados para avaliação de software: o *System Usability Scale* (SUS), que mensura a percepção de usabilidade [10], e o *Technology Acceptance Model* (TAM), que avalia aspectos como utilidade percebida, facilidade de uso e intenção de uso futuro [17].

Segundo Bangor et al. [7], os escores SUS podem ser agrupados em categorias interpretativas associadas à experiência do usuário. Pontuações superiores a 85 são consideradas excelentes, indicando alta aceitação do sistema; escores entre 70 e 84 são classificados como bons a muito bons, refletindo usabilidade satisfatória; valores entre 50 e 69 indicam uma usabilidade marginal, aceitável apenas em determinados contextos; e escores abaixo de 50 revelam sérios problemas de usabilidade. Essa escala interpretativa permite que pesquisadores e profissionais compreendam o desempenho relativo de um sistema em termos de usabilidade, situando-o em um continuum de qualidade com base em percentis e critérios empíricos validados. A Tabela 2 apresenta os itens avaliados.

Table 2: Itens avaliado de acordo com o SUS

ID	Itens
1.	Eu acho que gostaria de usar o AgileMood com frequência.
2.	Achei o AgileMood desnecessariamente complexo.
3.	Achei o AgileMood fácil de usar.
4.	Eu precisaria da ajuda de uma pessoa com conhecimentos técnicos para usar o AgileMood.
5.	As diversas funcionalidades do AgileMood estão bem integradas.
6.	Achei que o AgileMood apresenta muita inconsistência.
7.	Eu imagino que a maioria das pessoas aprenderia a usar o AgileMood rapidamente.
8.	Achei o AgileMood muito confuso de usar.
9.	Me senti confiante usando o AgileMood.
10.	Precisei aprender muitas coisas antes de conseguir usar o AgileMood.

O TAM foi utilizado como referencial teórico e instrumental metodológico para avaliar a aceitação da tecnologia pelos usuários. Desenvolvido por Davis [16], o modelo propõe que a *Utilidade Percebida* (PU - *Perceived Utility*) e a *Facilidade de Uso Percebida* (PEOU - *Perceived Ease of Use*) são determinantes fundamentais da *Atitude em Relação ao Uso* (ATU - *Attitude Toward Use*) e da *Intenção Comportamental de Uso* (BI - *Behavioral Intention*), que, por sua vez, influenciam o uso efetivo da tecnologia. Para operacionalizar o modelo, foi aplicado um questionário estruturado contendo itens validados para cada constructo, avaliados em escala Likert de cinco

pontos (1 = discordo totalmente; 5 = concordo totalmente). A PEOU mede o grau em que o usuário acredita que o sistema é de fácil manuseio; a PU avalia a percepção de contribuição do sistema para o desempenho nas tarefas; a ATU reflete a avaliação afetiva quanto ao uso; e a BI expressa a predisposição comportamental para utilizar a tecnologia. A Tabela 3 apresenta a descrição do instrumento. As médias de cada constructo foram calculadas a partir das respostas individuais, possibilitando a análise descritiva dos dados e a interpretação dos níveis de aceitação com base nos princípios do modelo.

Table 3: Itens avaliado de acordo com o TAM

Descrição dos itens avaliados por construtos do TAM
Facilidade de Uso Percebida (Perceived Ease of Use - PEOU)
PEOU1:Eu considero o AgileMood fácil de usar. PEOU2:Interagir com o AgileMood é claro e compreensível. PEOU3:Aprender a utilizar o AgileMood foi simples. PEOU4:Torna-se natural lembrar como utilizar o AgileMood entre uma Sprint e outra.
Utilidade Percebida (Perceived Usefulness - PU)
PU1:O AgileMood melhora minha capacidade de acompanhar o clima psicológico da equipe. PU2:O uso do AgileMood ajuda a identificar problemas que afetariam a produtividade da equipe. PU3:O AgileMood facilita decisões mais informadas durante as retrospectivas. PU4:Usar o AgileMood melhora a dinâmica de confiança dentro da equipe.
Atitude em Relação ao Uso (Attitude Toward Use - ATU)
ATU1:Eu gosto da ideia de usar o AgileMood regularmente nas Sprints. ATU2:Utilizar o AgileMood me parece uma boa prática de equipe.
Intenção Comportamental de Uso (Behavioral Intention - BI)
BI1:Pretendo continuar utilizando o AgileMood nas próximas Sprints. BI2:Indicaria o AgileMood para outras equipes ágeis.

A coleta de dados foi realizada por meio dos questionários aplicados individualmente aos participantes. A adoção de um delineamento de estudo de caso permitiu compreender a experiência prática dos estudantes com a ferramenta em um ambiente educacional, preservando a autonomia dos times e respeitando a naturalidade do processo de desenvolvimento.

4 Resultados

Nesta seção, são apresentados os resultados das etapas da pesquisa. Na Etapa 1, descrita na Seção 2, foi apresentada a base teórica oriunda da revisão exploratória. A seguir, na Subseção 4.1, apresentamos as funcionalidades implementadas para o desenvolvimento da ferramenta, e na Subseção 4.2, os resultados da aplicação do estudo de caso.

4.1 AgileMood - Requisitos e P blico-Alvo

O AgileMood foi desenvolvido com foco em equipes \'geis de desenvolvimento de software e seus l ders (*Scrum Masters*, *agile coaches* e gerentes de projeto) como principais usu rios. Ele visa apoiar tanto equipes que j  adotam metodologias \'geis quanto aquelas em transi o para um ambiente mais \'gil. Para os membros do time, a ferramenta serve como um canal seguro para expressar preocup es e percep es que, de outra forma, poderiam ficar ocultas. Para os facilitadores (l ders de equipe, Agile Coaches), o AgileMood oferece visibilidade cont nua sobre um aspecto intang vel do desempenho de equipe, permitindo interven es mais informadas. Em resumo, o AgileMood busca apoiar todos os envolvidos na melhoria do clima de seguran a psicol gica, promovendo um ambiente onde cada integrante se sinta confort vel para contribuir e aprender, com funcionalidades centrais definidas a partir de boas pr ticas identificadas na literatura e em ferramentas correlatas [2, 21].

A Figura 2 apresenta a tela de coleta de feedbacks, cujas funcionalidades s o descritas a seguir. Os par grafos seguintes, sintetizam os principais requisitos da ferramenta.

Figure 2: Coleta do feedback dos colaboradores

RF01 - Question rio Anônimo Peri dico: A ferramenta deve aplicar periodicamente um question rio anônimo de 7 itens aos membros da equipe definidos para um projeto/equipe. Este question rio est  baseado na escala validada de Edmondson [18] com 7 itens que avaliam diferentes aspectos da seguran a psicol gica, como a capacidade de assumir riscos interpessoais ou pedir ajuda sem constrangimento. A periodicidade da aplic o do question rio deve ser configur vel por usu rios com perfil de l der ou gerente, permitindo a sele o de intervalos como “ao final de cada Sprint”, “semanalmente” ou “mensalmente”. A aplic o do question rio ocorrer  via notifica o no aplicativo e/ou envio de e-mail aos membros da equipe eleg veis.

RF02 - C culo de M tricas de Seguran a Psicol gica: A ferramenta deve agregar as respostas an micas dos question rios para gerar um *score* m dio de seguran a psicol gica do time [21].

Al m da pontua o geral, o AgileMood deve calcular e apresentar indicadores de dispers o interna das respostas, como desvio-padr o ou amplitude [22], a fim de evidenciar a variabilidade nas percep es entre os membros da equipe.

RF03 - Dashboard de Visualiza o: A ferramenta deve prover um painel visual (*dashboard*) que permita a l ders e membros da equipe acompanhar a evolu o hist rica das m tricas de seguran a psicol gica calculadas (RF02), conforme apresentado na Figura 3. O dashboard deve exibir gr ficos simples que representem o hist rico do score m dio e dos indicadores de dispers o interna a cada p rio de coleta (por exemplo, a cada *Sprint*). Essa visualiza o torna tang vel um aspecto subjetivo do clima da equipe, facilitando discuss es baseadas em dados. A Figura 4 apresenta a tela do dashboard.

RF04 - Recomenda es e Dicas: Com base nos resultados agregados dos question rios, a ferramenta deve fornecer recomenda es gerais para melhorar a seguran a psicol gica do time. Essas recomenda es devem ser contextualizadas, sendo sugeridas quando m tricas espec ficas ca em abaixo de um limiar pr -definido ou apresentarem alta dispers o. As sugest es devem incluir pr ticas \'geis ou din micas de *team-building* pertinentes. Por exemplo, se identificada baixa seguran a em um item espec fico (como “membros do time evitam apontar problemas”), o AgileMood sugere pr ticas \'geis ou din micas de *team-building* pertinentes extra das da literatura [19, 37].

RF05 - Controle de Acesso e Privacidade: Assegurar aos seus usu rios controle de acesso e privacidade. O acesso ao dashboard completo (RF03) e às m tricas detalhadas deve ser restrito a membros com pap is autorizados, como o *Scrum Master* ou l der de equipe designado, por meio de autentica o e autoriza o adequadas. Todas as respostas individuais aos question rios devem ser mantidas em sigilo e processadas de forma a garantir o anonimato completo, exibindo apenas dados agr gados (RF02). A ferramenta n o deve armazenar, processar ou exibir qualquer dado individual identific vel relacionado às respostas dos question rios, garantindo a confidencialidade e incentivando a honestidade nas respostas.

Adicionalmente, o AgileMood oferece outro fluxo de uso, permitindo que qualquer membro da equipe registre a qualquer momento seu estado emocional, garantindo um acompanhamento cont nuo e proativo do clima emocional.

RF06 - Registro Cont nuto de Sentimentos: Qualquer integrante da equipe deve poder acessar o AgileMood a qualquer momento durante a Sprint e registrar seu estado emocional atual, selecionando um ou mais sentimentos de uma lista pr -definida pela ferramenta. Ao ser registrado um novo sentimento, o *Scrum Master* ou o gerente de projetos devem receber uma notifica o imediatamente indicando que um novo sentimento foi registrado (sem revelar a identidade do remetente neste momento).

RF07 - Envio de Mensagens An micas ou Identificadas: A ferramenta deve permitir que o integrante da equipe, opcionalmente, envie mensagem de texto adicional, junto ao registro de sentimento (RF06). Esta mensagem pode ser enviada de forma an mica ou identificada, conforme escolha do usu rio. O prop sito desta funcionalidade ´ permitir que o integrante forne  contexto sobre seu sentimento registrado, descreva preocup es espec ficas, ou apresente sugest es.

Avatar	Nome	Email	Função	Ações
A	arthur	arthur@email.com	Funcionário	<button>Remover</button>
V	victor yghor	devictoryghor@gmail.com	Funcionário	<button>Remover</button>
M	marcelo	marcelo@email.com	Funcionário	<button>Remover</button>
G	gabriel	gabriel@email.com	Funcionário	<button>Remover</button>

Emoção	Intensidade	Usuário	Observações	Data	Ações
Feliz	Muito Baixa	Arthur	teste	Data não disponível	<button>Enviar Feedback</button>
Neutro	Média	Victor	to de boa	Data não	<button>Enviar Feedback</button>

Figure 3: Visualização de um time por um gerente

Informações do Time
Dados gerais e estatísticas do time

O time "Time 1" possui 4 membros e um total de 9 registros de emoções, sendo 6 anônimos (67%) e 3 identificados. A emoção mais frequente é "Feliz" 😊 com 2 registros. A intensidade média das emoções é 4,0/5. 44% das emoções registradas são negativas, fique atento.

Nome do Time:	Time 1
Total de Membros:	4
Total de Registros de Emoções:	9
Registros Anônimos:	6
Registros Identificados:	3
Data de Criação:	12/03/2025

Distribuição de Emoções
Frequência de cada emoção registrada no período

Intensidade Média
Intensidade média de cada emoção registrada no período

Figure 4: Tela de Dashboard do manager

RF08 - Interação Anônima Garantida: Quando uma mensagem (RF07) for enviada de forma anônima, o Scrum Master deve ter a capacidade de respondê-la diretamente através do *AgileMood*, sem que a identidade da pessoa remetente seja revelada em nenhum momento. O sistema deve garantir o completo anonimato em toda a comunicação subsequente relacionada a essa mensagem, permitindo uma troca confidencial e segura de informações.

RF09 - Monitoramento e Ação Proativa: A ferramenta deve fornecer um painel de monitoramento que permita ao Scrum Master (ou PM) visualizar a qualquer momento os sentimentos registrados pelos membros da equipe (RF06). Este painel deve facilitar a identificação de tendências emocionais, padrões ou problemas emergentes através de visualizações agregadas. Com base nessas informações, ações imediatas podem ser tomadas para resolver problemas detectados, oferecendo suporte individual ou coletivo à equipe, e promovendo um ambiente psicologicamente seguro e saudável.

Esse módulo garante que o *AgileMood* seja um recurso contínuo para a gestão emocional das equipes, fortalecendo ainda mais a segurança psicológica e o bem-estar durante todo o ciclo da *Sprint*.

4.1.1 Exemplo de Uso da Ferramenta. Para ilustrar o funcionamento do *AgileMood*, consideremos um cenário hipotético em uma equipe Scrum. Ao final de cada Sprint, o *Scrum Master* agenda uma breve pesquisa via *AgileMood*:

- (1) **Coleta de Feedback Anônimo:** Os integrantes da equipe recebem uma notificação (por email ou mensagem integrada ao *Slack*) com um link para o questionário do *AgileMood*. Eles respondem anônimamente às 7 afirmativas da escala de segurança psicológica (por exemplo, “No meu time, posso admitir erros sem sentir que serei julgado”) usando uma escala Likert de 1 (Discordo totalmente) a 5 (Concordo totalmente). O processo de resposta leva em torno de 2 minutos.
- (2) **Agregação e Cálculo:** Assim que todos respondem (ou ao expirar de um prazo definido), o *AgileMood* consolida os dados. Suponha que nesta Sprint a pontuação média do time foi 4,2 de 5, porém com alto desvio-padrão em uma das questões, indicando que algumas pessoas se sentem menos seguras que outras para apontar problemas.

- (3) **Visualização dos Resultados:** O Scrum Master acessa o dashboard do *AgileMood*. Uma seção resume o *score* geral atual de segurança psicológica e sua tendência em relação às *Sprints* anteriores. Outra seção destaca “alertas”: por exemplo, “*Heterogeneidade alta nas respostas sobre franqueza para dar feedback negativo*”. As Figuras 4 e 3 mostram exemplos de telas de resultados exibindo a pontuação agregada e a variação das respostas entre membros.
- (4) **Reflexão na Retrospectiva:** Munida dessas informações, a equipe discute os resultados na retrospectiva da *Sprint*. O *Scrum Master* pode perguntar o que está dificultando alguns membros de serem francos, abrindo espaço para um diálogo seguro.
- (5) **Acompanhamento Contínuo:** Nas *sprints* seguintes, o *AgileMood* permitirá verificar se as ações tomadas surtiram efeito, ou seja, se a percepção de segurança psicológica aumentou e se alinhou entre todos da equipe. Dessa forma, o uso contínuo da ferramenta incentiva um ciclo de melhoria contínua do clima de trabalho, alinhado aos princípios ágeis de feedback rápido e adaptação.

4.1.2 Disponibilidade de Artefatos. A fim de incentivar a adoção e contribuições da comunidade, o *AgileMood* é disponibilizado como software livre de código aberto. O projeto está licenciado sob a licença MIT, uma licença permissiva que permite uso, modificação e distribuição livre, inclusive para fins comerciais, desde que mantido o aviso de copyright. Essa escolha se alinha às recomendações de *Open Science*, permitindo que outros pesquisadores e praticantes examinem o código, reproduzam o funcionamento da ferramenta e até estendam suas capacidades para novos contextos.

O código-fonte completo do *AgileMood*, incluindo frontend, backend e scripts de implantação, está publicado em um repositório público do GitHub³⁴). Junto ao repositório, fornecemos documentação para instalação e uso da ferramenta.

Além do código, disponibilizamos um pacote do artefato para avaliação reproduzível, disponível na plataforma Zenodo⁵. Esse pacote contém:

- **Demo online:** Mantemos uma instância de demonstração do *AgileMood* em funcionamento, onde interessados podem experimentar a ferramenta em um ambiente simulado.
- **Vídeo Demonstrativo:** Um vídeo de aproximadamente 4 minutos foi gravado, mostrando os principais fluxos de uso do *AgileMood* – desde o preenchimento do questionário por um membro da equipe até a análise do dashboard pelo Scrum Master. Esse vídeo está hospedado em um serviço público (Zenodo) para fácil acesso.

4.2 Avaliação da Ferramenta

A partir das respostas aos questionários do SUS e TAM, foi possível obter a percepção de 26 dos 41 usuários que utilizaram o *AgileMood* ao longo de duas sprints, garantindo ao menos dois respondentes por equipe.

Essa amostra permitiu responder à pergunta de pesquisa *Q1: AgileMood é percebida como fácil de usar nas equipes ágeis de software?*

³<https://github.com/Arth19/AgileMood-Backend>

⁴<https://github.com/Arth19/AgileMood-FrontEnd>

⁵<https://zenodo.org/records/1547783>

A pontuação média obtida na System Usability Scale (SUS) foi de 82,1. Esse resultado posiciona o sistema na faixa de usabilidade considerada “muito boa” segundo a escala interpretativa proposta por Bangor et al. [7], indicando um elevado grau de aceitação e facilidade de uso percebida pelos participantes. A pontuação está significativamente acima da média histórica de 68, comumente utilizada como referência para avaliar a adequação de sistemas interativos, o que sugere que os usuários conseguiram realizar suas tarefas de forma eficiente, eficaz e satisfatória. Além disso, o uso contínuo do sistema em dois ciclos de *sprint* confere maior robustez à avaliação, pois reflete a percepção de usabilidade em contextos reais de uso, e não apenas em situações simuladas ou pontuais. Esses achados reforçam a adequação da interface e a maturidade das funcionalidades implementadas, oferecendo subsídios positivos para sua evolução nas próximas iterações.

Com base nas respostas da pergunta aberta do questionário: “*Você gostaria de sugerir alguma melhoria para a usabilidade do AgileMood?*”, os usuários reconhecem que o *AgileMood* apresenta uma experiência fluida e funcional, embora apontem oportunidades de aprimoramento visual e de expressividade. O participante P1 destacou: “*Eu gostei muito das possibilidades de emoções, é uma forma simples de expressar como estou me sentindo*”, o que reforça a eficácia da interface ao representar estados emocionais de forma acessível. Essa percepção indica que o design cumpre seu papel central de facilitar a interação sem exigir esforço cognitivo elevado por parte do usuário.

Entretanto, algumas sugestões sinalizam pontos de atenção quanto à evolução do sistema. O participante P2 mencionou: “*Somente em questão visual, pois está um pouco poluído visualmente*”, sugerindo que o layout poderia ser mais limpo e organizado. Já o participante P3 comentou: “*Acredito que há margens para outros sentimentos*”, indicando que o repertório de opções emocionais poderia ser ampliado para abranger experiências mais diversas. Esses comentários evidenciam um equilíbrio entre satisfação com a simplicidade e expectativas por uma interface ainda mais refinada e expressiva.

Já em relação à pergunta de pesquisa *Q2: AgileMood é percebida como útil para apoiar a Segurança Psicológica nas equipes ágeis de software?*, os resultados obtidos a partir da aplicação do modelo TAM revelam percepções positivas dos usuários em relação à tecnologia avaliada (Veja Figura 5).

A média geral de **Facilidade de Uso Percebida (PEOU)** foi de 4,29, indicando que os participantes consideraram o sistema fácil de utilizar. Esse resultado está alinhado com a literatura, que reconhece a PEOU como um dos principais preditores da aceitação tecnológica [16, 46]. A **Utilidade Percebida (PU)** apresentou uma média de 3,79, um valor moderadamente elevado, porém inferior à PEOU. Isso sugere que, embora os usuários reconheçam a facilidade de uso do sistema, o impacto percebido sobre seu desempenho individual pode ainda não estar completamente consolidado. A **Atitude em Relação ao Uso (ATU)** obteve média de 3,98, refletindo uma avaliação afetiva global positiva em relação à tecnologia, em consonância com estudos que indicam que atitudes favoráveis tendem a emergir quando o sistema é percebido como útil e de fácil utilização. Por fim, a **Intenção Comportamental de Uso (BI)** apresentou média de 3,73, ligeiramente inferior aos demais constructos, o que pode indicar uma predisposição positiva à continuidade de uso, embora acompanhada de alguma hesitação. De acordo com a literatura,

níveis mais elevados de PU e PEOU tendem a fortalecer a BI [46], o que sugere que ações voltadas à valorização da utilidade percebida podem contribuir significativamente para ampliar a intenção de uso futuro. Esses resultados, portanto, revelam um cenário promissor de aceitação tecnológica, com oportunidades de aprimoramento na comunicação de benefícios concretos da ferramenta.

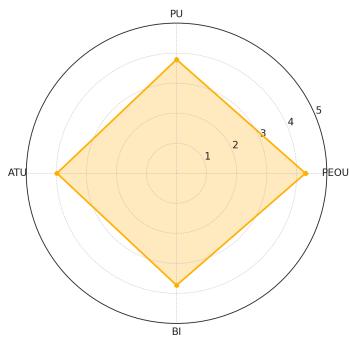


Figure 5: Diagrama De Radar – Constructos Do TAM

Sob a ótica da literatura, os achados estão em consonância com o modelo teórico proposto por Davis [16] e posteriormente expandido por Venkatesh e Davis [46]. De acordo com esses autores, a PEOU exerce influência direta sobre a PU. O fato de a PEOU (4,29) apresentar média superior à PU (3,79) sugere que a facilidade percebida ainda não está sendo plenamente convertida em percepção de utilidade. Esse descompasso pode decorrer de uma falta de clareza quanto aos benefícios funcionais do sistema ou de um possível desalinhamento entre suas funcionalidades e as demandas específicas dos usuários. Além disso, como a PU é reconhecida como o principal preditor da BI, o resultado obtido ($PU = 3,79 > BI = 3,73$) reforça que esforços para ampliar a percepção de utilidade podem impactar diretamente a intenção de uso. A diferença entre ATU (3,98) e BI (3,73) também corrobora essa interpretação: mesmo diante de uma atitude favorável, barreiras práticas, organizacionais ou de percepção de valor podem limitar a conversão dessa disposição em adoção efetiva.

Nesse contexto, algumas ações estratégicas podem ser adotadas para fortalecer a aceitação da tecnologia. A primeira delas é a melhoria na comunicação dos benefícios concretos do sistema, evidenciando seu impacto sobre o desempenho, a produtividade e a organização do trabalho, aspecto ainda percebido de forma moderada pelos usuários, conforme indica a média de Utilidade Percebida ($PU = 3,79$). Além disso, a promoção de casos de uso reais e depoimentos de usuários satisfeitos pode ampliar essa percepção de utilidade e reforçar a confiança dos demais usuários. Também é recomendável a identificação e remoção de barreiras externas que não estão relacionadas diretamente à usabilidade, como carga de trabalho, disponibilidade de tempo, suporte institucional ou necessidade de capacitação. Os resultados positivos nos constructos de Facilidade de Uso Percebida (PEOU = 4,29) e Atitude em Relação ao Uso (ATU = 3,98) revelam um potencial latente de aceitação que pode ser ativado com ações específicas de valorização do sistema e estímulo ao engajamento dos usuários nas próximas fases de implementação e uso.

Com base nas respostas coletadas, a experiência do usuário com o AgileMood pode ser considerada amplamente positiva, destacando-se a usabilidade e a proposta da ferramenta. Muitos usuários elogiaram a simplicidade e a facilidade de uso, como afirma P3: “*Gostei. É simples, fácil de usar e pode ser uma ferramenta interessante para o time*”. Da mesma forma, P5 comentou: “*Foi boa, uma ferramenta bem feita e simples de utilizar*”. Esse padrão de respostas indica que a ferramenta atende às expectativas no que diz respeito à interface e naveabilidade, atributos essenciais para a adoção em ambientes ágeis.

Além disso, os usuários demonstram percepção de valor na proposta do AgileMood enquanto recurso de acompanhamento e melhoria contínua no ambiente de trabalho. P1, por exemplo, afirmou: “*Até agora eu estou gostando. Tenho ambas as experiências como colaboradora e facilitadora, e em ambos os papéis achei útil*”. Já P2 foi direto ao dizer: “*Uma experiência ótima, muito bem feito!*”. Embora haja sugestões de melhorias, como no relato de P4: “*A experiência foi boa, só não foi melhor porque tive problemas no preenchimento das respostas anteriores*”, o conjunto de feedbacks revela uma aceitação consistente, reforçando o potencial do AgileMood como ferramenta de suporte à gestão ágil de equipes.

Diante dos dados obtidos por meio dos questionários SUS e TAM⁶, bem como das percepções qualitativas expressas pelos participantes, é possível afirmar que o AgileMood foi amplamente bem recebido pelos usuários ao longo de sua aplicação em dois ciclos de *sprint*. A avaliação de usabilidade revelou uma experiência consistente e positiva, com pontuação elevada na escala SUS e médias satisfatórias nos constructos do TAM, especialmente no que se refere à Facilidade de Uso Percebida (PEOU). As respostas também evidenciaram uma atitude favorável em relação à ferramenta e uma predisposição à continuidade de uso, ainda que acompanhadas de sugestões pertinentes para aprimoramentos visuais e funcionais. Esses resultados não apenas respondem às questões de pesquisa propostas, mas também reforçam a adequação da solução desenvolvida às práticas de equipes ágeis, apontando caminhos claros para seu aperfeiçoamento incremental.

5 Discussão e Considerações Finais

Este trabalho teve como objetivo principal desenvolver e apresentar o AgileMood, uma ferramenta voltada para a avaliação contínua e promoção da segurança psicológica em equipes ágeis de desenvolvimento de software. A ferramenta surge como resposta a lacunas identificadas na literatura e na prática, especialmente no que diz respeito à ausência de mecanismos sistemáticos e integrados para mensurar e apoiar o desenvolvimento de um ambiente seguro e colaborativo nas equipes.

As principais descobertas deste estudo indicam que o AgileMood permite visualizar o estado da segurança psicológica de uma equipe por meio de métricas objetivas, visualizações comprehensíveis e recomendações práticas. A ferramenta se destaca por preservar o anonimato dos membros do time, oferecer integração fluida com o fluxo ágil e apoiar ciclos contínuos de melhoria, possibilitando a identificação de problemas no clima da equipe. Esses problemas tornam-se visíveis, permitindo que scrum masters e gerentes atuem

⁶<https://doi.org/10.5281/zenodo.17289926>

de forma proativa para garantir o bem-estar dos integrantes durante a execução de suas atividades.

A principal contribuição deste trabalho reside no desenvolvimento de um artefato tecnológico fundamentado na literatura científica, que operacionaliza o constructo de segurança psicológica com base em métricas validadas. Além disso, ao disponibilizar o *AgileMood* como software livre e documentado, o projeto favorece sua replicação, adaptação e evolução por parte da comunidade acadêmica e profissional, promovendo avanços tanto práticos quanto teóricos.

Com base na avaliação empírica conduzida com sete equipes de estudantes em cursos de Computação, os resultados obtidos indicam uma aceitação positiva da ferramenta utilizada ao longo das sprints. A diversidade dos projetos desenvolvidos demonstra a aplicabilidade da ferramenta em diferentes domínios e escopos técnicos. O uso em contextos reais de sala de aula, com autonomia metodológica e papéis distintos em cada time, permitiu a utilização da ferramenta e a verificação das percepções dos usuários.

A análise realizada com base nos instrumentos SUS e TAM, forneceu evidências consistentes de usabilidade satisfatória e aceitação tecnológica. As pontuações obtidas no SUS situaram a ferramenta na faixa interpretativa de usabilidade boa a muito boa, conforme o referencial de Bangor et al. [7], indicando que os usuários perceberam o sistema como funcional e fácil de utilizar. Os resultados do TAM reforçam esse achado: a Facilidade de Uso Percebida (PEOU) obteve o maior valor médio, refletindo uma experiência de interação fluida; a Utilidade Percebida (PU), embora positiva, apresentou valor moderado, sugerindo que os ganhos funcionais ainda podem ser mais bem evidenciados. A Atitude em Relação ao Uso (ATU) foi favorável, enquanto a Intenção Comportamental de Uso (BI), embora consistente, foi a menor entre os construtos, indicando um potencial de adesão que pode ser fortalecido.

Esses achados apontam para implicações práticas importantes. A experiência dos estudantes com a ferramenta, em condições naturais de uso, evidencia seu potencial de apoio a projetos acadêmicos orientados por metodologias ágeis. No entanto, os dados também sinalizam oportunidades de aprimoramento, especialmente no que se refere à comunicação dos benefícios do sistema e à superação de barreiras contextuais que impactam a intenção de uso continuado. Assim, os resultados reforçam a relevância de abordagens centradas no usuário e da combinação entre instrumentos padronizados e observação em campo como estratégias eficazes para avaliar soluções tecnológicas no contexto educacional.

5.1 Ameaças à Validade

A literatura em engenharia de software empírica ressalta a importância de identificar e registrar as ameaças à validade [20]. A discussão sobre essas ameaças e suas mitigações foi estruturada de acordo com as taxonomias, propostas por Wohlin et al. [50], que categorizam as ameaças em validade de construto, interna e externa e de conclusão.

Para a validade de construto, utilizamos instrumentos de coleta de dados amplamente reconhecidos na literatura, como o SUS e o TAM. A aplicação desses questionários garante que os construtos de usabilidade e aceitação foram mensurados de forma consistente. A validade interna, teve sua principal ameaça, o viés de seleção,

minimizada. A coleta de dados foi realizada em um contexto natural de uso com feedback anônimo, o que reduziu o risco de distorção dos resultados. A validade de conclusão estatística foi tratada por meio da análise descritiva dos dados, com o cálculo de médias e desvios-padrão. A amostra de 41 estudantes, adequada para um estudo de caso exploratório, permite conclusões confiáveis sobre a usabilidade e a utilidade percebida da ferramenta. Por fim, a ameaça mais significativa é a validade externa, que diz respeito à generalização dos resultados. O estudo empírico foi conduzido em um ambiente acadêmico com estudantes, o que pode não ser totalmente representativo de equipes ágeis em ambientes corporativos. Esta limitação é um ponto de partida para estudos futuros.

5.2 Limitações e trabalhos futuros

As limitações deste trabalho devem ser consideradas na interpretação de seus resultados e em estudos futuros. O estudo foi conduzido com um número restrito de sete equipes de estudantes de terceiro ano de engenharia de software e avaliado por apenas dois *sprints* em um contexto acadêmico, o que pode não refletir totalmente a realidade e as pressões de equipes profissionais em ambientes industriais, limitando a generalização dos achados. A dependência de dados auto-relatados, mesmo que anônimos, introduz um viés inherente. Além disso, reconhecemos como limitação a inexistência de participação direta de psicólogos no desenho inicial da solução. Finalmente, este trabalho apresenta principalmente a proposição e pilotagem da ferramenta *AgileMood*, sendo necessários estudos futuros para avaliar empiricamente o impacto direto e mensurável de sua utilização na melhoria contínua da qualidade do software em contextos organizacionais reais.

Como trabalhos futuros, planejamos expandir a validação empírica do *AgileMood* para além do estudo inicial. Isso inclui a aplicação da ferramenta em equipes ágeis profissionais e em ambientes industriais reais, com um maior número de participantes e por períodos mais extensos, a fim de observar mudanças sustentadas na segurança psicológica e nos estados emocionais, bem como seu impacto a longo prazo na colaboração, desempenho e, crucialmente, na qualidade do software entregue.

Além disso, o sistema foi idealizado para ser integrado a ferramentas de gestão de projetos, como Jira, Trello ou Azure DevOps. Nesse contexto, por exemplo, um membro logado em uma dessas ferramentas, ao executar uma atividade, teria a opção de enviar uma mensagem ou registrar uma emoção sobre como está se sentindo naquele momento de desenvolvimento. Essas funcionalidades serão implementadas em uma próxima versão.

Intencionamos também realizar estudos que permitam quantificar o impacto da ferramenta em métricas objetivas de projeto e produto. Além disso, consideraremos ainda o desenvolvimento de novas funcionalidades, como análises preditivas ou recomendações mais personalizadas, sempre mantendo o foco na não intrusividade e na privacidade dos usuários.

REFERENCES

- [1] Red Hat Open Innovation Labs. 2020. Open Practice Library. <https://openpracticelibrary.com>. [Online; accessed 17-July-2025].
- [2] Mohammed-Amr Abd El-Migid, Damon Cai, Thomas Niven, Jeffrey Vo, Kashumi Madampe, John Grundy, and Rashina Hoda. 2022. Emotimonitor: A trello power-up to capture and monitor emotions of agile teams. *Journal of Systems and Software* 186 (2022), 111206.

- [3] Adam Alami, Oliver Krancher, and Maria Paasivaara. 2022. The journey to technical excellence in agile software development. *Information and Software Technology* 150 (2022), 106959.
- [4] Adam Alami, Mansooreh Zahedi, and Oliver Krancher. 2023. Antecedents of psychological safety in agile software development teams. *Information and Software Technology* 162 (2023), 107267.
- [5] Adam Alami, Mansooreh Zahedi, and Oliver Krancher. 2024. The role of psychological safety in promoting software quality in agile teams. *Empirical Software Engineering* 29, 5 (2024), 119.
- [6] Markus Baer and Michael Frese. 2003. Innovation is not enough: Climates for initiative and psychological safety, process innovations, and firm performance. *Journal of Organizational Behavior: The International Journal of Industrial, Occupational and Organizational Psychology and Behavior* 24, 1 (2003), 45–68.
- [7] Aaron Bangor, Philip T Kortum, and James T Miller. 2008. An empirical evaluation of the system usability scale. *Intl. Journal of Human-Computer Interaction* 24, 6 (2008), 574–594.
- [8] Roy F. Baumeister, Kathleen D. Vohs, C. Nathan DeWall, and Liang Zhang. 2007. How Emotion Shapes Behavior: Feedback, Anticipation, and Reflection, Rather Than Direct Causation. *Personality and Social Psychology Review* 11, 2 (2007), 167–203. doi:10.1177/1088868307301033
- [9] Kimberley Breevaart, Arnold B Bakker, Evangelia Demerouti, and Daantje Derkx. 2016. Who takes the lead? A multi-source diary study on leadership, work engagement, and job performance. *Journal of Organizational Behavior* 37, 3 (2016), 309–325.
- [10] John Brooke et al. 1996. SUS-A quick and dirty usability scale. *Usability evaluation in industry* 189, 194 (1996), 4–7.
- [11] Marte Pettersen Buvil and Anastasia Tkalicich. 2021. Psychological safety in agile software development teams: Work design antecedents and performance consequences. *arXiv preprint arXiv:2109.15034* (2021).
- [12] Lan Cao and Eun Hee Park. 2017. Understanding goal-directed emotions in agile software development teams. (2017).
- [13] Abraham Carmeli and Jody Hoffer Gittell. 2009. High-quality relationships, psychological safety, and learning from failures in work organizations. *Journal of Organizational Behavior: The International Journal of Industrial, Occupational and Organizational Psychology and Behavior* 30, 6 (2009), 709–729.
- [14] Shane Connelly and Brita S. Torrence. 2018. The Relevance of Discrete Emotional Experiences for Human Resource Management: Connecting Positive and Negative Emotions to HRM. *Research in Personnel and Human Resources Management* 36 (2018), 1–49. doi:10.1108/S0742-730120180000036001
- [15] Broderick Crawford, Ricardo Soto, Claudio León de la Barra, Kathleen Crawford, and Eduardo Olguín. 2014. The influence of emotions on productivity in software engineering. In *International Conference on Human-Computer Interaction*. Springer, 307–310.
- [16] Fred D Davis. 1989. Perceived usefulness, perceived ease of use, and user acceptance of information technology. *MIS quarterly* (1989), 319–340.
- [17] Fred D Davis, Richard P Bagozzi, and Paul R Warshaw. 1989. User acceptance of computer technology: A comparison of two theoretical models. *Management science* 35, 8 (1989), 982–1003.
- [18] Amy Edmondson. 1999. Psychological safety and learning behavior in work teams. *Administrative science quarterly* 44, 2 (1999), 350–383.
- [19] Amy C Edmondson and Zhike Lei. 2014. Psychological safety: The history, renaissance, and future of an interpersonal construct. *Annu. Rev. Organ. Psychol. Organ. Behav.* 1, 1 (2014), 23–43.
- [20] Robert Feldt and Ana Magazinius. 2010. Validity threats in empirical software engineering research—an initial survey.. In *Seke*. 374–379.
- [21] M Lance Frazier, Stay Fainshmidt, Ryan L Klinger, Amir Pezeshkan, and Veselina Vracheva. 2017. Psychological safety: A meta-analytic review and extension. *Personnel psychology* 70, 1 (2017), 113–165.
- [22] Bård Fyhn, Henning Bang, Therese Egeland, and Vidar Schei. 2023. Safe among the unsafe: Psychological safety climate strength matters for team performance. *Small Group Research* 54, 4 (2023), 439–473.
- [23] Daniela Grassi, Filippo Lanubile, Nicole Novielli, and Alexander Serebrenik. 2023. Towards supporting emotion awareness in retrospective meetings. In *2023 IEEE/ACM 45th International Conference on Software Engineering: New Ideas and Emerging Results (ICSE-NIER)*. IEEE, 101–105.
- [24] Daniel Graziotin, Fabian Fagerholm, Xiaofeng Wang, and Pekka Abrahamsson. 2018. What happens when software developers are (un) happy. *Journal of Systems and Software* 140 (2018), 32–47.
- [25] Daniel Graziotin, Xiaofeng Wang, and Pekka Abrahamsson. 2014. Happy software developers solve problems better: psychological measurements in empirical software engineering. *PeerJ* 2 (2014), e289. URL <https://www.ncbi.nlm.nih.gov/pmc/articles/PMC3961150> (2014).
- [26] Daniel Graziotin, Xiaofeng Wang, and Pekka Abrahamsson. 2015. Do feelings matter? On the correlation of affects and the self-assessed productivity in software engineering. *Journal of Software: Evolution and Process* 27, 7 (2015), 467–487.
- [27] André Luiz Nunes Kaiser, Eliezer Dutra, and Gleison Santos. 2025. Investigating How Psychological Safety Can Be Fostered in Agile Teams. In *Simpósio Brasileiro de Engenharia de Software (SBES)*. SBC, 348–359.
- [28] Helena Lee. 2021. Changes in workplace practices during the COVID-19 pandemic: the roles of emotion, psychological safety and organisation support. *Journal of Organizational Effectiveness: People and Performance* 8, 1 (2021), 97–128. doi:10.1108/JOEP-07-2020-0125
- [29] Per Lenberg and Robert Feldt. 2018. Psychological safety and norm clarity in software engineering teams. In *Proceedings of the 11th international workshop on cooperative and human aspects of software engineering*. 79–86.
- [30] Tan Trung Luong, Uthayasan Kar Sivarajah, and Vishanth Weerakkody. 2021. Do agile managed information systems projects fail due to a lack of emotional intelligence? *Information Systems Frontiers* 23, 2 (2021), 415–433.
- [31] Kashumi Madampe, John Grundy, Minh Nguyen, Ellen Welstead-Cloud, Vinh Tuan Huynh, Linh Doan, William Lay, and Sayed Hashim. 2025. EmoReflex: an AI-powered emotion-centric developer insights platform. *Automated Software Engineering* 32, 1 (2025), 22.
- [32] Kashumi Madampe, Rashna Hoda, and John Grundy. 2022. The role of emotional intelligence in handling requirements changes in software engineering. *arXiv preprint arXiv:2206.11603* (2022).
- [33] Marcelo Marinho, Fernanda Albuquerque, Luis Amorim, and Suzana Sampaio. 2025. Explorando a Influência da Segurança Psicológica na Qualidade do Software em Equipes Ágeis. In *Simpósio Brasileiro de Engenharia de Software (SBES)*. SBC, 60–70.
- [34] Marcelo Marinho, Luís Amorim, Rafael Camara, Brigitte Renata Oliveira, Marcos Sobral, and Suzana Sampaio. 2021. Happier and further by going together: The importance of software team behaviour during the COVID-19 pandemic. *Technology in society* 67 (2021), 101799.
- [35] Marcelo Marinho, John Noll, Ita Richardson, and Sarah Beecham. 2019. Plan-driven approaches are alive and kicking in agile global software development. In *2019 ACM/IEEE International Symposium on Empirical Software Engineering and Measurement (ESEM)*. IEEE, 1–11.
- [36] Claudio Saraiava Mattos, Eliezer Dutra, and Gleison Santos. 2024. An Instrument for Assessing Power Distance in Agile Organizations-Preliminary Results. In *Proceedings of the XXIII Brazilian Symposium on Software Quality*. 136–146.
- [37] Alexander Newman, Ross Donohue, and Nathan Eva. 2017. Psychological safety: A systematic review of the literature. *Human resource management review* 27, 3 (2017), 521–535.
- [38] Nicole Novielli, Fabio Calefato, and Filippo Lanubile. 2018. A gold standard for emotion annotation in stack overflow. In *2018 IEEE/ACM 15th International Conference on Mining Software Repositories (MSR)*. IEEE, 145–17 (2018).
- [39] Per Runeson and Martin Höst. 2009. Guidelines for conducting and reporting case study research in software engineering. *Empirical software engineering* 14, 2 (2009), 131–164.
- [40] Borrego Gilberto Salido, Martin, Ramón René Palacio Cinco, Luis-Felipe Rodriguez, et al. 2023. Agile software engineers' affective states, their performance and software quality: a systematic mapping review. *Journal of Systems and Software* 204 (2023), 111800.
- [41] Kurt Schneider, Jil Klünder, Fabian Kortum, Lisa Handke, Julia Straube, and Simone Kauffeld. 2018. Positive affect through interactions in meetings: The role of proactive and supportive statements. *Journal of Systems and Software* 143 (2018), 59–70.
- [42] Falk Schuetzenmeister. 2010. University research management: An exploratory literature review. (2010).
- [43] Leo Silva, Marilia Castro, Miriam Silva, Milena Santos, Uirá Kulesza, Margarida Lima, and Henrique Madeira. 2022. Emotional Dashboard: a Non-Intrusive Approach to Monitor Software Developers' Emotions and Personality Traits. In *2022 IEEE 22nd International Conference on Software Quality, Reliability and Security (QRS)*. IEEE, 366–375.
- [44] Anastasiia Tkalicich, Darja Smite, Nina Haugland Andersen, and Nils Brede Moe. 2022. What happens to psychological safety when going remote? *Ieee Software* 41, 1 (2022), 113–122.
- [45] Philippe Vandekerckhove, Tiny Steijvers, Wout Hendriks, and Wim Voordecker. 2018. Socioemotional wealth separation and decision-making quality in family firm TMTs: The moderating role of psychological safety. *Journal of Management Studies* 55, 4 (2018), 648–676. doi:10.1111/joms.12277
- [46] Viswanath Venkatesh and Fred D Davis. 2000. A theoretical extension of the technology acceptance model: Four longitudinal field studies. *Management science* 46, 2 (2000), 186–204.
- [47] Sai Datta Vishnubhotla, Emilia Mendes, and Lars Lundberg. 2018. An insight into the capabilities of professionals and teams in agile software development: A systematic literature review. In *Proceedings of the 2018 7th international conference on software and computer applications*. 10–19.
- [48] Antoinette Weibel, Deanne N. Den Hartog, Nicole Gillespie, Rosalind Searle, Frederik Six, and Denise Skinner. 2016. How do controls impact employee trust in the employer? *Human Resource Management* 55, 3 (2016), 437–462.
- [49] M West. 2018. How teams improve staff engagement and why it matters.
- [50] Claes Wohlin, Per Runeson, Martin Höst, Magnus C Ohlsson, Björn Regnell, Anders Wesslén, et al. 2012. *Experimentation in software engineering*. Vol. 236. Springer.