

**UNIVERSIDADE DE SÃO PAULO
FACULDADE DE CIÊNCIAS FARMACÊUTICAS
Curso de Graduação em Farmácia-Bioquímica**

**Desenvolvimento de aplicativo para pessoas com Transtorno
Bipolar: uma ferramenta para auxiliar o paciente no autocuidado e
o profissional da saúde no seu manejo.**

João Vitor Harada Soares

Trabalho de Conclusão do Curso de
Farmácia-Bioquímica da Faculdade de
Ciências Farmacêuticas da Universidade
de São Paulo.

Orientador(a):
Dra. Maria Aparecida Nicoletti

São Paulo

2023

SUMÁRIO

	Pág.
LISTA DE ABREVIATURAS	1
RESUMO	2
1. INTRODUÇÃO	3
2. OBJETIVO	5
3. MATERIAIS E MÉTODOS	5
4. RESULTADOS	12
5. DISCUSSÃO	17
6. CONCLUSÃO	18
7. REFERÊNCIAS	19

LISTA DE ABREVIATURAS

TBH	Transtorno Bipolar do Humor
PNAF	Política Nacional de Assistência Farmacêutica
MVP	<i>Minimum Viable Product</i>
LCM	<i>Life Chart Method</i>
NIMH	<i>National Institute of Mental Health</i>
SQL	<i>Structured Query Language</i>

RESUMO

Aplicativos para *smartphone* voltados para diferentes áreas da saúde vêm ganhando cada vez mais destaque. Todavia, uma área um pouco menos explorada é a relacionada a transtornos mentais como, por exemplo, o transtorno bipolar. O transtorno bipolar apresenta-se como um transtorno psiquiátrico crônico e complexo, caracterizado pela alternância entre episódios de depressão e mania. Por ser um transtorno de características muito variáveis, é importante que o paciente registre suas mudanças de humor, energia e sono diário, não só com o objetivo de adquirir mais autoconhecimento e monitorar o autocuidado, mas também para que o profissional de saúde responsável possa manejar o tratamento medicamentoso e orientar o paciente com maior precisão. Portanto, o objetivo deste projeto foi o desenvolvimento de um aplicativo gratuito para dispositivos Android, cuja principal funcionalidade será o registro dessas variáveis importantes para pacientes que sofrem de transtorno bipolar. Por se tratar de um aplicativo para a área da saúde, é importante que suas funcionalidades estejam embasadas científicamente. Por esse motivo, o *Life Chart Method* (LCM) do Instituto Nacional de Saúde Mental (NIMH) nos Estados Unidos foi utilizado como referência para o registro de humor no aplicativo. A fim de se obter um aplicativo performático e seguro, utilizou-se o *Figma* como ferramenta de *design* e o *Flutter* como ferramenta para a etapa de desenvolvimento. O resultado é um aplicativo simples, com aparência clean e boa usabilidade, e que poderia ser uma alternativa válida para o LCM físico em suas futuras versões. Com a ajuda de ferramentas de lançamento e distribuição de aplicativos como a Google® Play Console, sugestões de melhorias e adição de novas funcionalidades poderiam ser enviadas pelos usuários. Embora seja uma tarefa complexa e desafiadora, a criação de um aplicativo para monitoramento do transtorno bipolar pode impactar positivamente a vida dos pacientes, sendo uma ferramenta muito útil para o autocuidado e acompanhamento das tendências às flutuações de humor.

Palavras-chave: transtorno bipolar, aplicativo, *Life Chart Method*

1. INTRODUÇÃO

Atualmente, o uso de aplicativos em dispositivos como o *smartphone* faz parte da rotina da maioria das pessoas. Fazer compras *online*, registrar lembretes, controlar despesas, interagir em redes sociais: tudo é feito pelo celular. Neste cenário, aplicativos para as áreas da saúde e bem-estar, também, vem ganhando cada vez mais destaque nas lojas virtuais, principalmente após os períodos de quarentena durante a pandemia de COVID-19, já que saúde mental e bem-estar passaram a ser temas amplamente discutidos, assim como as consultas médicas passaram a ser realizadas virtualmente. Dentre esses aplicativos, diversos são os exemplos de projetos que têm como objetivo a melhora da saúde mental, tais como aplicativos com exercícios para respiração, programas de meditação, terapia guiada para diminuir a ansiedade e o estresse, entre muitos outros. Ainda neste tópico, uma área um pouco menos explorada em aplicativos é a relacionada a transtornos mentais, mais especificamente ao transtorno bipolar.

O transtorno bipolar do humor (TBH) apresenta-se como um transtorno psiquiátrico crônico, complexo, incapacitante, de etiologia multifatorial e um delicado manejo farmacológico. Os episódios de humor no transtorno bipolar variam entre depressão e mania, e em ambos se observam alterações do humor, psicomotricidade, cognição e funções vegetativas que podem estar acompanhados ou não de sintomas psicóticos (Humes et al., 2016).

Embora não exista um padrão específico de alternância de episódios, uma vez que é possível que o humor se altere de um para outro sem necessariamente passar por um período de eutimia, o transtorno bipolar é classificado nos seguintes tipos (Humes et al., 2016):

Tabela 1 – Tipos de transtorno bipolar

Tipos	Descrição
Tipo I	Caracterizado por episódios de mania com duração de, no mínimo, sete dias, e fases de humor deprimido, geralmente com duração mínima de duas semanas e podendo se estender a vários meses. Profundas mudanças comportamentais e de conduta são notáveis em ambos os casos, sendo que o quadro pode se agravar a ponto de exigir internação hospitalar por conta do risco aumentado de suicídio e complicações psiquiátricas.
Tipo II	Caracterizado por uma alternância entre os episódios de depressão e os de hipomania (estado mais leve de euforia, excitação, otimismo e, às vezes, de agressividade), sem prejuízo maior para o comportamento e as atividades do portador.
Não especificado ou misto	Os sintomas sugerem o diagnóstico de transtorno bipolar, mas não são suficientes nem em número nem no tempo de duração para classificar a doença em um dos dois tipos anteriores.
Ciclotimia	É o quadro mais leve do transtorno bipolar, marcado por oscilações recorrentes do humor. O paciente alterna sintomas de hipomania e de depressão leve que não são intensos ou longos o suficiente para serem classificados como hipomania ou depressão.

Fonte: NIMH, 2022

Embora a maioria dos estudos epidemiológicos tenha identificado que o transtorno bipolar do tipo I afeta cerca de 1% da população adulta, estudos mais recentes sugerem que até 4% a mais podem ser afetados com outros espectros bipolares tais como o transtorno bipolar do tipo II e a ciclotimia (Smith et al., 2012).

Por ser um transtorno de características muito variáveis, o registro das mudanças de humor, qualidade do sono e outros eventos pelo paciente é fundamental para o tratamento, já que, dessa maneira, as crises, oscilações de humor e gatilhos se tornam mais previsíveis. O registro desses dados também pode ser de grande valor para o profissional da saúde responsável, pois com eles é possível ajustar o tratamento medicamentoso e orientar o paciente com maior precisão.

É importante ressaltar que o farmacêutico ocupa um lugar de destaque nesse processo, tendo na Política Nacional de Assistência Farmacêutica (PNAF) um dos seus pilares para colaborar com o tratamento deste usuário. A PNAF comprehende um conjunto de ações voltadas à promoção, proteção e recuperação da saúde - tanto individual como coletiva - e tem o medicamento como insumo essencial, visando seu acesso e seu uso racional (Leonardi, s.d.).

2. OBJETIVO

Desenvolver um aplicativo para dispositivos Android, cujas principais funcionalidades foram o registro de episódios, humor, sono diário, medicamentos e outras informações relacionadas aos distúrbios percebidos, bem como uma seção voltada para informações sobre o transtorno bipolar, aprendizado e atualização em Educação em Saúde. O aplicativo será gratuito e estará disponível para download na loja virtual Google® Play Store.

3. MATERIAIS E MÉTODOS

O desenvolvimento de um aplicativo envolve diversas etapas, desde a concepção da ideia até o lançamento no mercado. As etapas variam dependendo do método de desenvolvimento utilizado, mas, de modo geral, as principais são:

- i. Análise de requisitos: nessa etapa, são definidos os requisitos funcionais e não funcionais do aplicativo, ou seja, quais funcionalidades ele deve ter, quais sistemas operacionais deve suportar, quais são as necessidades de segurança, entre outros aspectos.
- ii. *Design* e prototipação: é a etapa em que o *design* do aplicativo é criado, incluindo a identidade visual, a arquitetura de informação, a naveabilidade e a

- interação do usuário com o aplicativo. Nessa etapa também é possível criar protótipos para testar o fluxo de navegação e as funcionalidades.
- iii. Desenvolvimento: é a etapa em que o aplicativo é codificado. É nessa fase que os desenvolvedores criam as funcionalidades, integrando as tecnologias necessárias para o funcionamento do aplicativo.
 - iv. Publicação: Após o término dos testes, o aplicativo é lançado no mercado e distribuído nas lojas de aplicativos, como a App Store e a Google® Play.
 - v. Atualizações: depois que o aplicativo é lançado, é importante continuar trabalhando nele, corrigindo falhas, adicionando novas funcionalidades e melhorando a experiência do usuário. As atualizações do aplicativo devem ser distribuídas regularmente para manter a relevância do aplicativo no mercado (Carrington, 2023).

Dependendo do tamanho do projeto para um aplicativo, isto é, da equipe envolvida, dos custos e do tempo disponível, seria necessário adicionar mais etapas, como, por exemplo, uma etapa para pesquisa de mercado e outra etapa para fase de testes. Portanto, é importante esclarecer que o aplicativo deste projeto terá apenas sua primeira versão publicada e será de porte pequeno. Pode-se, então, dizer que o aplicativo resultante deste projeto é um MVP, que é a sigla para *Minimum Viable Product*, ou Produto Mínimo Viável, em português.

MVP se trata de uma técnica utilizada no desenvolvimento de produtos, principalmente em *startups*, para validar uma ideia de negócio ou produto de forma rápida e com o mínimo de recursos possíveis. É a versão mais simples do produto que ainda assim entrega valor ao usuário, possuindo apenas os recursos e funcionalidades essenciais para resolver um problema específico ou atender a uma necessidade do cliente. O objetivo do MVP é testar a aceitação do produto no mercado, validar hipóteses e colher *feedback* dos usuários, para que o produto possa ser aprimorado de forma iterativa e com base em dados reais (Rouse, 2020).

3.1. Principais funcionalidades

Embora existam diversos aplicativos publicados que cumprem uma função similar, essas ferramentas, ainda não são utilizadas em tratamentos profissionais (Patoz et al., 2021). Além disso, os aplicativos são rapidamente descontinuados pelos

pacientes após o *download*, e essa baixa adesão pode ser explicada por alguns fatores, tais como: 1) a falta de aplicativos cientificamente validados, o que leva a uma falta de confiabilidade por parte dos usuários e 2) a falta de usabilidade, geralmente causada por funcionalidades desnecessárias para o usuário ou até mesmo por um *design* pouco intuitivo e confuso (Patoz et al., 2021).

Logo, tratando-se do transtorno bipolar, um aplicativo para monitoramento e tratamento da doença deve seguir alguma metodologia validada cientificamente e possuir boa usabilidade. Uma dessas metodologias é o *Life Chart Method* (LCM) do Instituto Nacional de Saúde Mental (NIMH) nos Estados Unidos, que consiste em uma ferramenta de uso diário que permite avaliar a alternância de episódios de depressão e mania (Denicoff et al., 2000).

Essa ferramenta permite avaliar o curso da doença tanto de maneira prospectiva quanto retrospectiva, possibilitando a identificação de fatores de risco que possam desencadear episódios graves e a efetividade da medicação. No entanto, o uso dessa ferramenta pode ser tedioso para o uso rotineiro, pois requer dos pacientes um treinamento e orientação adequados, além de grande quantidade de dados a serem inseridos (Rajagopalan et al., 2017). Além disso, a metodologia LCM exige o registro dos dados em papel, o que requer uma maior disciplina do paciente, além de diminuir a praticidade de uso diário e se tornar mais sujeita ao esquecimento.

A digitalização dessa ferramenta poderia, então, solucionar esses problemas. É importante mencionar que uma versão eletrônica do LCM já foi desenvolvida e, de acordo com estudos recentes sobre esta versão, a autoavaliação pelos pacientes utilizando a ferramenta digital é uma alternativa válida e econômica às versões utilizadas por clínicos (Rajagopalan et al., 2017).

Em um estudo recente envolvendo pacientes com transtorno bipolar e profissionais da saúde, buscou-se entender quais seriam os requisitos para um aplicativo de monitoramento de transtorno bipolar, na perspectiva do usuário. Como resultado, a liberdade de personalizar o aplicativo, a facilidade de uso e a confiabilidade foram os requisitos mais mencionados pelos participantes (Geerling et al., 2021). A fim de atender a esses requisitos, procurou-se desenvolver um aplicativo intuitivo, com um *design clean*, com boa usabilidade e com as seguintes funcionalidades:

- i. Registro diário de humor baseado no LCM: o usuário poderá registrar diariamente o seu humor, horas dormidas, eventuais sintomas e medicação, ou seja, todos os dados requisitados em um formulário do LCM, de maneira rápida e intuitiva. Depois de salvos, os dados poderão ser visualizados em um gráfico contendo os detalhes de cada registro. Os registros também serão editáveis, caso o usuário queira fazer alguma alteração.
- ii. Personalização de notificações: o usuário poderá adicionar notificações com horário personalizado, para que um lembrete de preenchimento do diário apareça no horário programado.
- iii. Estatísticas: o usuário terá acesso a um resumo estatístico acerca da sua variação de humor, média de sono diário e média de humor diário.
- iv. Armazenamento local dos dados: a privacidade do usuário estará completamente garantida, visto que os dados não serão utilizados por terceiros e nem serão armazenados em servidores. Todos os dados estarão salvos, apenas, no dispositivo do usuário. Isto também possibilita o uso *offline* do aplicativo, ou seja, sem a necessidade de conexão com a *internet*.

3.2. Design

A fase de *design* contempla todo o planejamento do aplicativo que não envolve código, ou seja, consiste em esquematizar o aplicativo por completo, incluindo as cores utilizadas, a fonte, o tamanho dos textos, as ilustrações, as interações e as animações. As ferramentas utilizadas durante o planejamento das funcionalidades são de total liberdade de escolha dos membros participantes, basta haver uma comunicação eficiente e oportunidade para troca de ideias. Para a etapa de planejamento do *design* foi utilizado o *Figma*, uma ferramenta colaborativa de design de interfaces disponível para a *web* e também em formato de aplicativo para *desktop* (*Figma*, s.d.).

Geralmente, a elaboração de um *design* envolve também a criação de padrões e regras para ajudar no trabalho de outros *designers* e desenvolvedores, com o objetivo de manter uma mesma identidade visual. Essas padronizações são englobadas em um guia, denominado *Design System* (Sebrae, 2022). Para facilitar a criação de um *design clean*, intuitivo e com boa acessibilidade, utilizou-se para este

projeto o *Material Design 3* (ou *Material 3*), um *Design System* desenvolvido e disponibilizado pela Google®. O *Material 3* é um guia bastante completo e que abrange as mais diversas necessidades devido à sua alta flexibilidade (Material Design, s.d.).

Após a escolha do *Design System*, iniciou-se a etapa de elaboração da interface de usuário utilizando o *Figma*, ou seja, as telas e interações no aplicativo. Depois de concluir as telas e interações do aplicativo no *Figma*, deu-se início à etapa de desenvolvimento.

3.3. Desenvolvimento

O desenvolvimento é a parte principal do projeto, pois se trata da elaboração do aplicativo final que será publicado nas lojas virtuais. Para esta etapa, diversas são as abordagens que podem ser adotadas, visto que existe uma grande variedade de ferramentas que possibilitam o desenvolvimento de um aplicativo. Uma abordagem extensivamente difundida na área é o desenvolvimento por meio de *frameworks*, que são ferramentas que servem de infraestrutura básica para construir aplicativos ou sistemas. Um *framework* é composto por um conjunto de bibliotecas, módulos e ferramentas que permitem desenvolver aplicativos com mais eficiência e rapidez, pois muitas das tarefas de programação já estão pré-definidas (Codecademy, 2021).

Eles fornecem uma estrutura padronizada para o desenvolvimento de aplicativos, o que facilita a colaboração entre desenvolvedores, torna o código mais legível e ajuda a manter uma consistência em todo o projeto. Existem diversos *frameworks* disponíveis para diferentes linguagens de programação, como *Ruby on Rails*, *Django* para *Python*, *Laravel* para *PHP*, *Angular* para *JavaScript*, entre outros (Codecademy, 2021). Cada *framework* tem suas próprias características, funcionalidades e padrões de *design*, mas todos compartilham o objetivo de tornar o desenvolvimento de aplicativos mais fácil, eficiente e padronizado.

Geralmente, se tratando de um aplicativo ou *site*, é preciso separar a etapa de desenvolvimento em dois grupos: o desenvolvimento do *front-end* e o desenvolvimento do *back-end*.

Front-end é a parte do projeto com a qual o usuário final efetivamente interage, podendo ser um aplicativo instalado no *smartphone*, uma página na *web*, um programa de computador, etc. *Design*, animações, cores e interações pertencem ao *front-end*, sendo que o responsável pelo processamento dos arquivos de código é o próprio dispositivo do usuário (Codesido, 2009). Para este projeto utilizou-se o *Flutter*, um *framework open-source* e gratuito, feito para criar aplicações multiplataforma compiladas nativamente com uma única base de código. Grandes empresas como *Nubank*®, *iFood*®, *Globo*®, *QuintoAndar*® e *Alibaba*® já adotaram o *Flutter* em alguns de seus projetos (Flutter, s.d.).

Já o *back-end* é a parte do projeto com a qual o usuário final não interage e não conhece, pois é a parte do projeto que é processada em servidores. O banco de dados do aplicativo, os serviços de autenticação de usuário, realização de pagamentos e qualquer operação que exija alto poder de processamento ou segurança são de responsabilidade do *back-end* (Lemonaki, 2022). Para esta etapa será utilizado o *Firebase*, também desenvolvido pela *Google*®, que é uma plataforma de multiserviços em nuvem e fornece soluções para banco de dados, autenticação de usuários, *crash reports*, etc. (*Firebase*, s.d.). Entretanto, visto que uma das funcionalidades principais do aplicativo é o armazenamento local dos dados, a fim de garantir a privacidade do usuário, optou-se por não utilizar o serviço de banco de dados do *Firebase* para esta versão inicial do aplicativo. Ao invés disso, utilizou-se uma adaptação do *SQLite* especialmente disponibilizada para o *Flutter*.

SQLite é um sistema de gerenciamento de banco de dados relacional que utiliza o *Structured Query Language* (SQL) para gerenciar e manipular dados. É uma biblioteca de *software* que fornece uma estrutura para armazenar e gerenciar bancos de dados, e é projetada para ser eficiente, confiável e fácil de usar (*SQLite*, s.d.). Uma das principais características do *SQLite* é que ele é uma biblioteca de banco de dados incorporável, o que significa que é possível incorporar o *SQLite* diretamente em um aplicativo, sem a necessidade de um servidor de banco de dados separado. Isso o torna uma opção popular para aplicativos que precisam de um armazenamento de dados leve e portátil.

Para este projeto, a principal utilidade do *back-end* foi a de rastrear erros e analisar falhas no aplicativo utilizando um outro serviço disponibilizado gratuitamente pelo

Firebase, chamado de *Firebase Crashlytics*. Ele ajuda os desenvolvedores a rastrear e corrigir falhas em seus aplicativos móveis. Quando um aplicativo é executado e ocorre uma falha, o *Firebase Crashlytics* coleta automaticamente informações sobre a falha, como o tipo de erro, a pilha de chamadas e o dispositivo em que ocorreu. Esses dados são, então, consolidados e apresentados em um painel de controle fácil de usar, permitindo que os desenvolvedores identifiquem rapidamente problemas em seus aplicativos. Além disso, o *Firebase Crashlytics* também fornece recursos como rastreamento de eventos e relatórios personalizados, permitindo que os desenvolvedores aprofundem as causas raiz das falhas em seus aplicativos e tomem medidas para corrigi-las (Firebase Crashlytics, 2023). Isso ajuda a garantir que os aplicativos móveis estejam funcionando da melhor maneira possível e oferecendo uma ótima experiência do usuário.

Por fim, um outro tópico essencial para o desenvolvimento eficiente de um sistema ou aplicativo é o versionamento. Também conhecido como controle de versão, o versionamento é o processo de gerenciar diferentes versões de um arquivo ou conjunto de arquivos ao longo do tempo. É uma prática comum na programação de *software*, mas também pode ser aplicada a outros tipos de arquivos, como documentos de texto, planilhas e imagens (O'Sullivan, 2009). O controle de versão permite que os desenvolvedores mantenham um histórico completo de todas as alterações feitas em um arquivo, permitindo que eles acompanhem o progresso do desenvolvimento, revertam para versões anteriores se necessário e trabalhem em equipe de forma colaborativa. Isso é particularmente importante em projetos de *software* complexos, onde várias pessoas estão trabalhando no mesmo conjunto de arquivos.

Existem várias ferramentas de controle de versão disponíveis, incluindo *Git*, *SVN* e *Mercurial*. Ademais, existem plataformas de hospedagem de código-fonte baseada em nuvem para controle de versão e colaboração, permitindo que os desenvolvedores trabalhem em equipe em projetos de *software* e gerenciem o controle de versão do código-fonte de seus projetos. Utilizar uma destas plataformas é sempre considerado uma boa prática, já que uma ferramenta de versionamento armazena os dados localmente, trazendo riscos de perda ou corrompimento dos dados. Optou-se pelo *Git* como ferramenta de versionamento e o *Github* como

plataforma de hospedagem do código, visto que são ferramentas extremamente populares entre os desenvolvedores (Abdalslam, 2023).

3.4. Publicação

A última fase do projeto é o lançamento e distribuição do aplicativo para o público. Nesta fase, o objetivo é alcançar a maior quantidade possível de usuários, garantindo que o aplicativo esteja disponível nas principais plataformas de distribuição de aplicativos, como a App Store da Apple e a Google® Play Store. Já que este projeto objetivou apenas no lançamento para dispositivos *Android*, a plataforma utilizada para distribuição será exclusivamente a Google® Play Store.

Para que se possa publicar um aplicativo na Google® Play Store, é necessário possuir uma conta de desenvolvedor. Uma conta de desenvolvedor pode ser criada sem restrições, basta fornecer um *e-mail* válido e pagar uma taxa de inscrição única. Depois disso, já é possível publicar um aplicativo, desde que ele respeite as regras e políticas de privacidade da plataforma (Play Console Help, 2023).

Após o lançamento, é importante monitorar o desempenho do aplicativo e garantir que ele esteja funcionando corretamente. Isso pode incluir a análise de dados de uso, o monitoramento de avaliações e comentários dos usuários, e a correção de quaisquer *bugs* ou problemas que surgirem. Uma estratégia bem planejada e executada pode ajudar a alcançar um grande número de usuários, garantir que o aplicativo esteja atendendo às necessidades dos usuários e coletar *feedbacks* para a evolução do aplicativo.

4. RESULTADOS

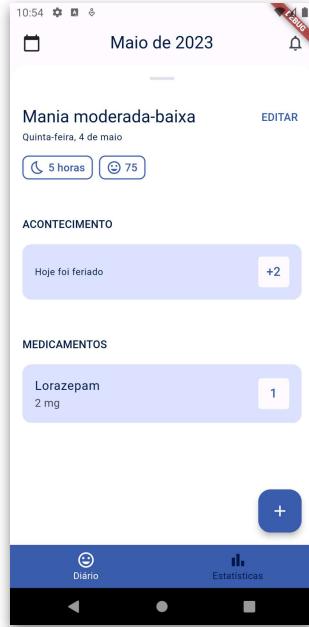
Na página inicial do aplicativo, é possível visualizar um gráfico que mostra a variação dos episódios de mania e depressão. No eixo horizontal encontram-se as datas dos registros em ordem decrescente, e no eixo vertical encontram-se as diferentes gravidades do episódio. A primeira linha corresponde à mania severa, enquanto a última à depressão severa. A linha do meio corresponde ao equilíbrio.

Figura 1 – Tela de diário



Fonte: screenshot

Figura 2 – Tela de diário (expandida)

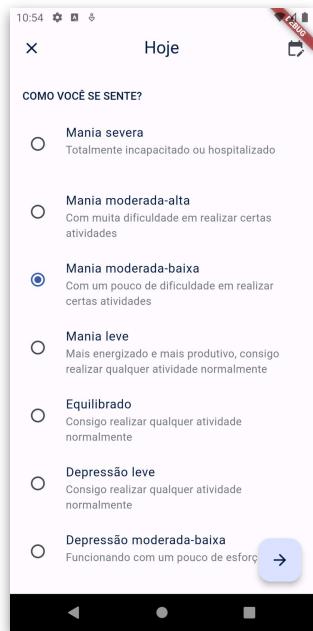


Fonte: screenshot

Logo abaixo do gráfico encontram-se os detalhes de cada registro. Informações como a gravidade do humor, a data do registro, as horas de sono e o nível de humor aparecem no topo. Ao rolar a tela para baixo, o usuário pode encontrar mais informações sobre o registro, tais como acontecimentos naquele dia e o seu respectivo nível de impacto no humor, medicamentos e suas dosagens, e observações escritas pelo usuário.

No canto inferior direito encontra-se um botão para adicionar um novo registro. Ao pressioná-lo, usuário será direcionado para a tela de registro diário. Nesta tela, o usuário deverá informar a gravidade do seu episódio no dia. Para tornar a escolha do usuário mais assertiva, o LCM correlaciona a gravidade do episódio ao prejuízo funcional do paciente (Leverich et al., 2002), e portanto, o mesmo foi feito neste formulário. Cada item possui como título a gravidade do episódio e o prejuízo funcional como descrição. Para alterar a data do registro, basta pressionar o ícone do calendário no canto superior direito da tela e selecionar a data desejada.

Figura 3 – Formulário de gravidade do humor



Fonte: screenshot

Figura 4 – Alterar da data do registro



Fonte: screenshot

Após a seleção da gravidade do episódio, deve-se pressionar o botão no canto inferior da tela para prosseguir. A última tela do registro diário dispõe de outros dados como: nível de humor de 0 a 100, horas dormidas, sintomas, medicamentos, acontecimento e observações. Os campos para medicamentos e acontecimento são opcionais.

Figura 5 – Formulário para outros dados



Fonte: screenshot

Após registrar as informações, deve-se pressionar o botão “Salvar”. Em seguida, o usuário será direcionado de volta para a tela de diário, onde poderá visualizar o novo registro.

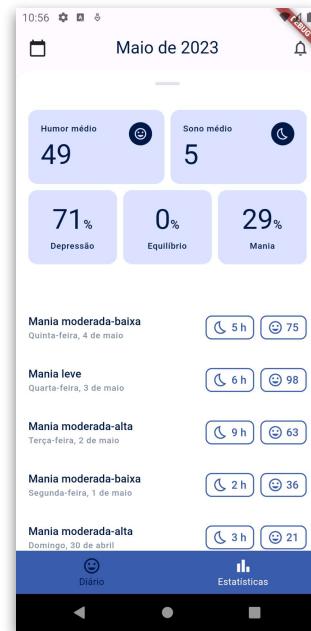
Além da tela de diário, o usuário poderá acessar suas estatísticas ao pressionar o menu “Estatísticas” na parte mais inferior do aplicativo. Nesta aba, um outro tipo de gráfico é mostrado, no qual é possível não só visualizar a variação de episódios, como também a variação do sono e do humor. Outra diferença está no intervalo de tempo mostrado: enquanto que na tela de diário o usuário deve rolar o gráfico para ver registros mais antigos, aqui os registros são agrupados para preencher a largura da tela. Assim, é possível observar a variação dos parâmetros durante todo o intervalo de tempo escolhido.

Figura 6 – Tela de estatísticas



Fonte: screenshot

Figura 7 – Tela de estatísticas (expandida)

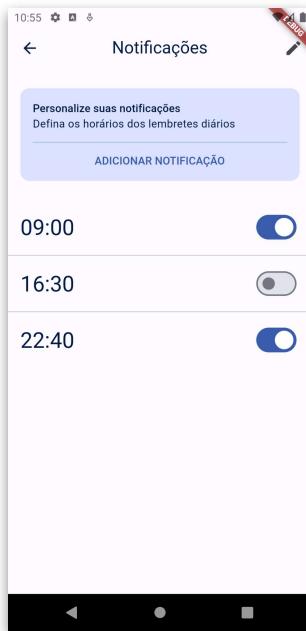


Fonte: screenshot

Além do gráfico, é possível visualizar algumas informações adicionais calculadas pelo aplicativo, como a média de humor, a média de horas dormidas e a proporção de episódios de depressão, equilíbrio ou mania. Ao rolar a tela para baixo, encontra-se uma lista dos registros resumidos em ordem decrescente de data.

Por fim, no canto superior de ambas as telas de diário e estatísticas encontra-se um ícone de notificação. Ao pressioná-lo, o usuário é direcionado para a tela de notificações, onde poderá adicionar ou remover notificações com horário personalizado.

Figura 8 – Tela de notificações



Fonte: screenshot

As notificações irão aparecer diariamente para o usuário no horário em que foram definidas. Ainda, é possível ativar ou desativar notificações deslizando o botão à direita de cada item.

Ao todo, foram necessárias aproximadamente 6 semanas para o planejamento das funcionalidades e do *design*, enquanto que a etapa de desenvolvimento durou cerca de 10 semanas. A etapa de publicação necessitou apenas de uma semana, já que, normalmente, aplicativos são aprovados pela Google® Play Store em poucos dias.

O aplicativo pode ser baixado gratuitamente em <https://play.google.com/store/apps/details?id=com.jvsoares.moodify>. Já o código-fonte do aplicativo no Github pode ser acessado em https://github.com/jv-soares/moodify_app.

5. DISCUSSÃO

O desenvolvimento completo de um aplicativo é uma tarefa complexa e trabalhosa. Normalmente, é necessário uma equipe composta por desenvolvedores, *designers*, testadores e coordenadores de projeto para criar um aplicativo seguro e com boa usabilidade. Para criar um aplicativo com funcionalidades semelhantes ao *Instagram*, por exemplo, estima-se que seriam necessárias 1180 horas para entregar um produto funcional apenas para dispositivos Android (Altynpara et al., 2023). O trabalho aumentaria consideravelmente caso dispositivos *iOS* também fossem considerados.

Porém, o tempo necessário para criar um aplicativo depende de inúmeras variáveis, como a *expertise* do time de desenvolvedores e *designers*, o *framework* utilizado, os sistemas operacionais alvo, a complexidade das funcionalidades, entre outras. Graças ao *Flutter*, o tempo necessário para criar aplicativos foi reduzido, significativamente, em comparação com outros *frameworks*, e por esse motivo sua popularidade vem crescendo recentemente. Ainda, o *Flutter* vem se consolidando como o *framework* com a melhor curva de aprendizagem, principalmente na perspectiva de desenvolvedores iniciantes (Karasavvas, 2022).

Todavia, pelo fato de o *Flutter* ser um *framework* recente, ainda carece de alguns aspectos que outras abordagens possuem, principalmente em relação às abordagens nativas, isto é, que são focadas em apenas uma plataforma, como Android ou *iOS* (Freire et al., 2019). Grandes empresas ainda costumam preferir desenvolvedores especializados na abordagem nativa, mesmo que isso exija o dobro de pessoas na equipe. Por outro lado, *startups* e outras pequenas empresas são as principais adeptas ao *Flutter* e outros *frameworks* recentes, e isso ocorre porque esses *frameworks* permitem que os desenvolvedores criem um aplicativo para múltiplas plataformas utilizando o mesmo código-fonte (Hodiak, 2022). Desse modo, o tempo e consequentemente os custos necessários são significativamente reduzidos.

É importante lembrar que um *MVP* se trata de um produto inicial e, conforme o número de usuários vai crescendo e estes vão fornecendo *feedback*, o aplicativo vai

continuamente ganhando novas funcionalidades e melhorias. Portanto, algumas possíveis melhorias para o aplicativo seriam:

- *Design* mais intuitivo: a colaboração de um *designer* de aplicativos poderia trazer melhorias na interface visual, na experiência do usuário e na facilidade de uso do aplicativo. Geralmente, equipes de *design* costumam ser divididas em *UX design* e *UI design*, que lidam com a experiência do usuário e com a interface do usuário, respectivamente. O *UX design* costuma lidar mais com a usabilidade do aplicativo, de maneira a deixá-lo intuitivo e com uma disposição lógica das funcionalidades. Já o *UI design* costuma lidar exclusivamente com a aparência do aplicativo, as cores, a fonte dos textos, as animações, etc.;
- Informações sobre o transtorno bipolar: uma seção informativa sobre o transtorno bipolar e o LCM poderia ser adicionada ao aplicativo;
- Compartilhamento de dados com profissionais: em uma seção separada, o usuário poderia escolher compartilhar seus registros com profissionais da saúde à sua escolha;
- Personalização do formulário diário: os usuários poderiam ser capazes de personalizar os sintomas, adicionar fotos e vídeos, entre outros;
- Mais *insights*: o aplicativo poderia fornecer mais *insights* para o usuário, como, por exemplo, prever episódios de mania ou depressão com base no histórico de registros.

Por fim, existem limitações pontuais do aplicativo nesta primeira versão, principalmente, no que se refere ao registros dos dados conforme o LCM. Por adicionar uma complexidade significativa tanto no *design* quanto no desenvolvimento, as variações de humor que ocorrem no mesmo dia, também chamadas de variações ultradianas, não foram implementadas.

6. CONCLUSÃO

A criação de um aplicativo para monitoramento do transtorno bipolar é um projeto desafiador e complexo, envolvendo diferentes etapas que exigem tempo, recursos e uma equipe multidisciplinar. Para que o aplicativo tivesse um grau suficiente de embasamento científico, utilizou-se a metodologia LCM como base para o registro de episódios de depressão ou mania no aplicativo, entre outras variáveis.

Com o uso de ferramentas como o *Figma* e o *Flutter*, foi possível desenvolver um aplicativo rápido, intuitivo e com aparência *clean*. Como em qualquer desenvolvimento de sistema, foram enfrentados diversos obstáculos no decorrer do projeto, que consequentemente aumentaram o tempo demandado.

Apesar dos desafios, a criação de um aplicativo para monitoramento do transtorno bipolar apresenta um enorme potencial para impactar positivamente a vida dos pacientes, sendo uma ferramenta muito útil para o autocuidado e o acompanhamento das tendências às flutuações de humor, possibilitando um manejo mais adequado do histórico acumulado, tanto para o paciente como para o profissional da saúde, incluindo o uso de medicamentos e de terapias não farmacológicas.

É importante destacar que a tecnologia não substitui o papel do profissional de saúde, mas pode ser um recurso valioso para auxiliar no tratamento e gerenciamento do transtorno bipolar. Além disso, é essencial que seja desenvolvida de forma ética e responsável, visando sempre o bem-estar e a saúde dos usuários.

7. REFERÊNCIAS

About SQLite. SQLite. Disponível em <<https://www.sqlite.org/about.html>>. Acesso em: 06 mai. 2023.

ALTYNPARA, E; CHABANOVKSA, D. Disclosed: how much does it cost to make an app like instagram? Cleveroad, 2023. Disponível em <<https://www.cleveroad.com/blog/cost-to-build-an-app-like-instagram/>>. Acesso em: 06 mai. 2023.

Bipolar disorder. NIMH. Disponível em <<https://www.nimh.nih.gov/health/publications/bipolar-disorder/>>. Acesso em: 09 mai. 2023.

CARRINGTON, M. Mobile app development process: ultimate guide to build an app. Velvetech, 2023. Disponível em <<https://www.velvetech.com/blog/mobile-app-development-process/>>. Acesso em: 06 mai. 2023.

CODESIDO, I. What is front-end development? The Guardian, 2009. Disponível em <<https://www.theguardian.com/help/insideguardian/2009/sep/28/blogpost>>. Acesso em: 06 mai. 2023.

Como usar o Play Console. Ajuda do Play Console. Disponível em <<https://support.google.com/googleplay/android-developer/answer/6112435?hl=pt-BR>>. Acesso em: 06 mai. 2023.

Creative tools meet the internet. Figma. Disponível em <<https://www.figma.com/about/>>. Acesso em: 06 mai. 2023.

DENICOFF, K. D; LEVERICH, G. S; NOLEN, W. A; RUSH, A. J; MCELROY, S. L; KECK, P. E; SUPPES, T; ALTSCHULER L. L; KUPKA, R; FRYE, M. A; HATEF, J; BROTMAN, M. A; POST, R. M. Validation of the prospective NIMH-Life-Chart Method (NIMH-LCM) for longitudinal assessment of bipolar illness. *Psychol Med*, 30(6), p. 1391-1397, 2000.

Firebase. Disponível em <<https://firebase.google.com/>>. Acesso em: 06 mai. 2023.

Firebase Crashlytics. Firebase, 2023. Disponível em <<https://firebase.google.com/docs/crashlytics>>. Acesso em: 06 mai. 2023.

Flutter apps in production. Flutter. Disponível em <<https://flutter.dev/showcase>>. Acesso em: 06 mai. 2023.

FREIRE, A; MOREIRA, A; FERREIRA, R; LESSINGER, R; MARACCINI, V. ANDRADE, V. Mobile architecture task force: why we think Flutter will help us scale mobile development at Nubank. Nubank, 2019. Disponível em <<https://cdn.nubank.com.br/mobile/taskforce/nubank-mobile-architecture-task-force-mission-report.pdf>>. Acesso em: 06 mai. 2023.

GEERLING, B; KELDERS, S. M; KUPKA, R. W; STEVENS, A. W. M. M; BOHLMEIJER, E. T. How to make online mood-monitoring in bipolar patients a success? A qualitative exploration of requirements. *International Journal of Bipolar Disorders*, 9(1), p. 39, 2021.

HODIAK, V. Flutter for startups: how Flutter became the leader in the cross-platform development. Otakoyi, 2022. Disponível em <<https://otakoyi.software/blog/flutter-for-startups-how-flutter-became-the-leader-in-the-cross-platform-development/>>. Acesso em: 06 mai. 2023.

HUMES, E. C; VIEIRA, M. E. B; JÚNIOR, R. F; HÜBNER, M. M. C. OLMOS, R. D. Psiquiatria interdisciplinar. Barueri, SP: Manole, 2016.

KARASAVVAS, T. Why Flutter is the most popular cross-platform mobile SDK. Stack Overflow Blog, 2022. Disponível em <<https://stackoverflow.blog/2022/02/21/why-flutter-is-the-most-popular-cross-platform-mobile-sdk/>>. Acesso em: 06 mai. 2023.

LEMONAKI, D. Frontend vs backend: what's the difference? Freecodecamp, 2022. Disponível em <<https://www.freecodecamp.org/news/frontend-vs-backend-whats-the-difference/>>. Acesso em: 06 mai. 2023.

LEONARDI, E. Atenção farmacêutica em psicofármacos. ICTQ. Disponível em <<https://ictq.com.br/varejo-farmaceutico/915-atencao-farmaceutica-em-psicofarmacos>>. Acesso em: 25 ago. 2022.

LEVERICH, G, S; POST, R. M; SPEARING, M. K. The NIMH Life Chart Manual for Recurrent Affective Illness. NIMH, 2002.

Material Design. Disponível em <<https://m3.material.io/>>. Acesso em: 06 mai. 2023.

O que é design system. Sebrae, 2022. Disponível em <<https://www.sebrae.com.br/sites/PortalSebrae/artigos/o-que-e-design-system,2e9ef4498f6a2810VgnVCM100000d701210aRCRD>>. Acesso em: 06 mai. 2023.

O'SULLIVAN, B. Mercurial: the definitive guide. 2009. Disponível em <<http://hgbook.red-bean.com/read/>>. Acesso em: 06 mai. 2023.

PATOZ, MC; HIDALGO-MAZZEI, D; PEREIRA, B; BLANC, O; CHAZERON, I; MURRU, A; VERDOLINI, N; PACCHIAROTTI, I; VIETA, E; LLORCA, PM; SAMALIN, L. Patients' adherence to smartphone apps in the management of bipolar disorder: a systematic review. *International Journal of Bipolar Disorders*, 9:19, 2021.

RAJAGOPALAN, A; SHAH, P; ZHANG, M. W; HO, R. C. Digital Platforms in the Assessment and Monitoring of Patients with Bipolar Disorder. *Brain Sciences*, 7(11), p. 150, 2017.

ROUSE, M. Minimum Viable Product. Techopedia, 2020. Disponível em <<https://www.techopedia.com/definition/27809/minimum-viable-product-mvp>>. Acesso em: 06 mai. 2023.

SMITH, D. J.; WHITHAM, E. A.; GHAEMI, S. N. Bipolar disorder. *Handbook of clinical neurology*, 106, p. 251–263, 2012.

Version control clients statistics, trends and facts 2023. Abdalslam, 2023. Disponível em <<https://abdalslam.com/version-control-clients-statistics>>. Acesso em: 06 mai. 2023.

What is a framework? Codecademy, 2021. Disponível em <<https://www.codecademy.com/resources/blog/what-is-a-framework/>>. Acesso em: 06 mai. 2023.

Documento assinado digitalmente
 JOAO VITOR HARADA SOARES
Data: 24/05/2023 11:23:16-0300
Verifique em <https://validar.iti.gov.br>

Documento assinado digitalmente
 MARIA APARECIDA NICOLETTI
Data: 22/05/2023 12:35:37-0300
Verifique em <https://validar.iti.gov.br>

Data e assinatura do aluno

Data e assinatura da orientadora