

UNIVERSIDADE FEDERAL DO MARANHÃO
CENTRO DE CIÊNCIAS EXATAS E TECNOLOGIA
DEPARTAMENTO DE INFORMÁTICA
CURSO DE GRADUAÇÃO EM CIÊNCIA DA COMPUTAÇÃO

ÊMILE CUNHA LOPES

**CHATBOT COMO FERRAMENTA NO AUXÍLIO AO COMBATE À DEPRESSÃO E
PREVENÇÃO AO SUICÍDIO**

SÃO LUÍS

2020

ÊMILE CUNHA LOPES

CHATBOT COMO FERRAMENTA NO AUXÍLIO AO COMBATE À DEPRESSÃO E
PREVENÇÃO AO SUICÍDIO

Monografia apresentada ao curso de Ciência da Computação da Universidade Federal do Maranhão, como parte dos requisitos necessários para obtenção do grau de Bacharel em Ciência da Computação.

Orientadora: Prof.^a M.^a Débora Rodrigues Stefanello

SÃO LUÍS

2020

Ficha gerada por meio do SIGAA/Biblioteca com dados fornecidos pelo(a) autor(a).
Núcleo Integrado de Bibliotecas/UFMA

Lopes, Êmile Cunha.

Chatbot como ferramenta no auxílio ao combate à
depressão e prevenção ao suicídio / Êmile Cunha Lopes. -
2020.

84 f.

Orientador(a): Débora Rodrigues Stefanello.

Monografia (Graduação) - Curso de Ciência da
Computação, Universidade Federal do Maranhão, São Luís,
2020.

1. Chatbots. 2. Depressão. 3. Dispositivos móveis.
4. Prevenção ao suicídio. I. Stefanello, Débora
Rodrigues. II. Título.

ÊMILE CUNHA LOPES

CHATBOT COMO FERRAMENTA NO AUXÍLIO AO COMBATE À DEPRESSÃO E
PREVENÇÃO AO SUICÍDIO

Monografia apresentada ao curso de Ciência da Computação da Universidade Federal do Maranhão, como parte dos requisitos necessários para obtenção do grau de Bacharel em Ciência da Computação.

Aprovada em:

BANCA EXAMINADORA

Prof.^a M.^a Débora Rodrigues
Stefanello (Orientadora)
Universidade Dom Bosco (UNDB)

Prof. Dr. Geraldo Braz Junior
Universidade Federal do Maranhão (UFMA)

Prof. Dr. Cícero Costa Quarto
Universidade Estadual do Maranhão (UEMA)

Prof. Msc. Rodrigo Monteiro de Lima
Universidade Dom Bosco (UNDB)

Aos meus pais que tanto batalharam e oraram pra que eu pudesse ter a chance de cursar uma graduação e, principalmente, para que eu me tornasse uma mulher segundo o coração de Deus. Vocês são meus padrões.

AGRADECIMENTOS

Agradeço em primeiro lugar a Deus, que é tão misericordioso que me permitiu concluir este trabalho.

Agradeço também aos meus pais, por serem minha base e me encherem de amor e dedicação.

À minha irmã Jóine, pela paciência, incentivo e por sempre me mostrar minhas loucuras.

Agradeço ao meu namorado Luís Alberto pelo apoio imprescindível para a realização desse trabalho e por me fazer tão bem.

Agradeço à Amanda, pelos anos de irmandade, risadas e carinho.

À Igreja Batista Filadélfia, minha família que eu tanto amo e que tem cuidado de mim, especialmente através da minha querida Célula Esmirna.

À minha orientadora pela paciência e por não desistir de mim mesmo depois de tantos sumiços.

Sou grata a cada um por tudo.

“O temor do Senhor é o princípio da sabedoria, e o conhecimento do Santo é entendimento.”

(Provérbios 9:10)

RESUMO

Tendo em vista que o Brasil é país da América Latina que mais sofre com depressão, trabalhos em Português do Brasil que aliam tecnologia ao combate à essa doença se mostram bastante importantes. Nesse contexto, desenvolver o protótipo de um chatbot mobile que possa servir como ferramenta no auxílio ao combate a depressão e prevenção ao suicídio, é o principal objetivo deste trabalho. Para tanto, é necessário identificar a relação entre um aplicativo de troca de mensagens e o combate a alguns sintomas da depressão, desenvolver uma aplicação para Android pela qual o chatbot possa ser testado, testar a aplicação desenvolvida e analisar os resultados obtidos. Realiza-se, então, a partir de um fluxo de diálogo proposto e utilizando o IBM Watson Assistant, o desenvolvimento do chatbot, que depois é incorporado numa aplicação Android construída utilizando o Android Studio como ambiente de desenvolvimento. Essa aplicação foi testada por 10 pessoas durante uma semana e ao final desse tempo, um questionário foi aplicado para avaliar a redução de sintomas no usuário, as funções cognitivas do chatbot e o desempenho e design da aplicação de teste. 7 dentre as 10 pessoas reportaram ter tido algum tipo de problema ao receber mensagens do chatbot. Entretanto, 8 dentre as 10 pessoas considerou as informações apresentadas pelo chatbot úteis. Além disso, verifica-se que durante o período de testes houve uma leve redução na observação de alguns sintomas em algumas pessoas. Assim, em vista desses fatos, considera-se que o chatbot desenvolvido ajudou na redução de sintomas da depressão, além de possuir funções cognitivas de nível regular que, juntamente com suas estratégias de acompanhamento, contribuem para que o mesmo seja relevante para o usuário.

Palavras-chave: Dispositivos Móveis. Chatbots. Depressão. Prevenção ao Suicídio.

ABSTRACT

Since Brazil is the Latin American country that suffers the most from depression, solutions in Brazilian Portuguese that use technology to combat this disease are very important. In this context, developing a prototype of a chatbot mobile that can be used as a tool to help combat depression and suicide prevention is the main goal of this research. Therefore, it is necessary to identify the relationship between a messaging application and the fight against some symptoms of depression, develop an Android application through which chatbot can be tested, test the developed application and analyze the results obtained. It then takes place, from a proposed dialog flow and using IBM Watson Assistant, chatbot development, which is then incorporated into an Android application built using Android Studio as the development environment. This application was tested by 10 people for a week and at the end of that time, a questionnaire was administered to assess user symptom reduction, chatbot's cognitive functions, and the performance and design of the test application. 7 out of 10 people reported having a problem receiving chatbot messages. However, 8 out of 10 people found the information provided by the chatbot helpful. In addition, it is found that during the testing period there was a slight reduction in the observation of some symptoms in some people. Thus, in view of these facts, it is considered that the developed chatbot has helped to reduce symptoms of depression, besides having regular level cognitive functions that, together with its accompanying strategies, contribute to its relevance to the user.

Keywords: Mobile devices. Chatbots. Depression. Suicide prevention.

LISTA DE FIGURAS

Figura 1	– Tela principal do aplicativo <i>Operation Reach Out</i>	20
Figura 2	– Lista de vídeos (<i>Operation Reach Out</i>)	21
Figura 3	– Exibição do vídeo selecionado (<i>Operation Reach Out</i>)	22
Figura 4	– Aplicativo "Ajuda Já"	23
Figura 5	– Apresentação da página do <i>Woebot</i> no Facebook	24
Figura 6	– Exemplo de primeiro contato com o <i>Woebot</i> e escolha de atividade baseada em Terapia Cognitivo Comportamental (TCC)	25
Figura 7	– Exemplo de mensagem padrão exibida quando o <i>Woebot</i> detecta uma emergência	26
Figura 8	– Resultados dos testes do <i>Woebot</i>	27
Figura 9	– Quadro de conceitos de Inteligência Artificial	31
Figura 10	– Exemplo da <i>intent</i> "pedir pizza"	35
Figura 11	– Onde habilitar as <i>system entities</i> no Watson Assistant	35
Figura 12	– Exemplo de diálogo no Watson Assistant	36
Figura 13	– Exemplo de um nó de diálogo no Watson Assistant	37
Figura 14	– Exemplo de como habilitar <i>slots</i> em um nó no Watson Assistant	37
Figura 15	– Exemplo de <i>slots</i> para o nó #pedir_pizza	38
Figura 16	– Exemplo de como habilitar digressão em um nó no Watson Assistant	39
Figura 17	– Esquema geral de integração do Firebase Cloud Messaging	47
Figura 18	– Diagrama do fluxo de conversa do protótipo de <i>chatbot</i>	49
Figura 19	– <i>Intents</i> do protótipo de <i>chatbot</i>	51
Figura 20	– <i>Entities</i> do protótipo de <i>chatbot</i>	51
Figura 21	– Nós do diálogo	52
Figura 22	– Exemplo de entrada do <i>dataset</i> "Diversos"	53
Figura 23	– Exemplo de entrada do <i>dataset</i> "Interações sociais"	53
Figura 24	– Diagrama de entidade relacionamento do banco de dados da aplicação	54
Figura 25	– Arquitetura do envio de notificação	58
Figura 26	– Protótipo das telas do aplicativo	59
Figura 27	– Telas após a implementação	59
Figura 28	– "Em quantos dias do período de teste, aproximadamente, você interagiu com o bot?"	62

Figura 29 – "Como você avaliaria as estratégias de acompanhamento do Bot?"	62
Figura 30 – "Quanto você considera que o Bot te ajudou no combate aos sintomas que você apresenta?"	63
Figura 31 – "Quanto você considera que o Bot entendeu suas mensagens?"	63
Figura 32 – "Aproximadamente quantas coisas você conseguiu aprender com o Bot?" . .	64
Figura 33 – "Quão útil para sua vida foi o que você aprendeu com o Bot?"	64
Figura 34 – "Você conseguiu enviar mensagem para o Bot?"	65
Figura 35 – "Você teve algum problema na hora de instalar o aplicativo?"	66
Figura 36 – "Você percebeu algum problema ao receber as mensagens (ex: receber men- sagens fora de contexto)?"	66
Figura 37 – "Alguma vez a aplicação “travou” no seu dispositivo?"	67
Figura 38 – "Quão demorado era pra receber mensagens do Bot?"	67
Figura 39 – "Quanto você avaliaria o esquema de cores do aplicativo?"	68
Figura 40 – "Como você avaliaria a organização e o visual do aplicativo?"	68

LISTA DE TABELAS

Tabela 1 – Tabela que representa a quantidade de usuários onde houve redução, por sintoma	61
Tabela 2 – Comparação dos trabalhos relacionados	70

LISTA DE CÓDIGOS-FONTE

Código-fonte 1 – Credenciais do serviço do Watson Assistant no Android	54
Código-fonte 2 – Objeto <i>MessageRequest</i>	55
Código-fonte 3 – Chamada à API do Watson Assistant	55
Código-fonte 4 – Método <i>accessLocation()</i>	56

LISTA DE ABREVIATURAS E SIGLAS

DSM V	Manual Diagnóstico e Estatístico de Transtornos Mentais V
IDE	Ambiente de Desenvolvimento Integrado (<i>Integrated Development Environment</i>)
OMS	Organização Mundial da Saúde
PLN	Processamento de Linguagem Natural
PTSD	Transtorno de Estresse Pós-Traumático
SDK	<i>Software Development Kit</i>
SQL	Linguagem de Consulta Estruturada
TCC	Terapia Cognitivo Comportamental
XML	<i>Extensible Markup Language</i>

SUMÁRIO

1	INTRODUÇÃO	16
1.1	Problema de Pesquisa	17
1.2	Justificativa	17
1.3	Objetivos	18
1.3.1	<i>Geral</i>	18
1.3.2	<i>Específicos</i>	18
1.4	Organização do Trabalho	19
1.5	Trabalhos Relacionados	19
1.5.1	<i>Operation Reach Out (MILITARY COMMUNITY AWARENESS INC, 2016)</i>	19
1.5.2	<i>Ajuda Já</i>	21
1.5.3	<i>Reporting Mental Health Symptoms: Breaking Down Barriers to Care with Virtual Human Interviewers (LUCAS et al., 2017)</i>	23
1.5.4	<i>Woebot (FITZPATRICK; DARCY; VIERHILE, 2017)</i>	24
2	FUNDAMENTOS TEÓRICOS	28
2.1	Transtorno Depressivo Maior	28
2.2	Processamento de Linguagem Natural	29
2.3	Inteligência Artificial	31
2.3.1	<i>Aprendizado de máquina</i>	32
2.3.2	<i>Cognitive Computing</i>	32
2.4	IBM Watson	33
2.4.1	<i>IBM Watson Assistant</i>	34
2.5	Android	38
2.5.1	<i>Android Studio</i>	40
2.5.2	<i>SQLite</i>	41
2.5.3	<i>Conceitos de um aplicativo Android</i>	42
2.6	Firestore	44
2.6.1	<i>Firestore Cloud Functions</i>	45
2.6.2	<i>Cloud Messaging</i>	46
3	METODOLOGIA DE DESENVOLVIMENTO	48
3.1	Tecnologias Utilizadas	48

3.2	Métodos no Desenvolvimento do Protótipo de <i>Chatbot</i>	48
3.2.1	<i>Fluxo de diálogo proposto</i>	48
3.2.2	<i>Intents e entities do projeto</i>	50
3.2.3	<i>Nós principais do diálogo</i>	52
3.3	Métodos no Desenvolvimento da Aplicação de Teste	52
3.3.1	<i>Formação das bases de dados necessárias</i>	52
3.3.2	<i>Modelagem do banco de dados</i>	53
3.3.3	<i>Incorporação do chatbot na aplicação</i>	54
3.3.4	<i>Tratamento das mensagens de “Indicação de interação social”</i>	56
3.3.5	<i>Serviço de notificação</i>	57
3.3.6	<i>Design da interface</i>	58
4	EXPERIMENTOS	60
4.1	Visão Geral	60
4.2	Resultados e Discussão	60
4.2.1	<i>Influência do chatbot na diminuição de sintomas</i>	60
4.2.2	<i>Funções cognitivas e relevância do bot para o participante</i>	61
4.2.3	<i>Utilidade das informações</i>	63
4.2.4	<i>Performance</i>	65
4.2.5	<i>Design da interface</i>	67
4.3	Análise de Trabalhos Relacionados	68
5	CONCLUSÕES E TRABALHOS FUTUROS	71
5.1	Limitações e Trabalhos Futuros	72
	REFERÊNCIAS	74
	APÊNDICES	77
	APÊNDICE A – TERMO DE CONSENTIMENTO LIVRE E ESCLARE- CIDO	77
	APÊNDICE B – QUESTIONÁRIO	78

1 INTRODUÇÃO

Quando se fala em robôs ou Inteligência Artificial é muito comum as pessoas lembrarem dos grandes filmes sobre o assunto como o clássico *A.I. - Inteligência Artificial*, de Steven Spielberg, onde cientistas desenvolvem um robô que acredita ser um humano e age como tal, ou a franquia *O Exterminador do Futuro*, com todo o seu cenário futurista de revolução das máquinas que aprenderam a pensar sozinhas. Apesar de ambos os cenários serem extremos, já é possível observar sistemas como os *chatbots* que interagem com as pessoas de forma a automatizar tarefas humanas o máximo possível.

Chatbot é uma palavra que vem do inglês onde “bot” significa robô e “chat” significa conversa, fazendo com que o seu significado seja literalmente “robô de conversa”. *Chatbots* são também conhecidos como assistentes virtuais ou agentes virtuais e tem a característica de simular o comportamento humano em uma troca de mensagens.

Há 60 anos atrás surgiu o primeiro *chatbot*. ELIZA, como foi chamado, foi criado por Joseph Weizenbaum para conversar com uma pessoa simulando um terapeuta humano (WEIZENBAUM, 1966). Ainda que quando criado só respondesse com uma reformulação da afirmação que o usuário fez, esse *chatbot* pioneiro é até hoje referência no desenvolvimento de robôs de conversação.

Atualmente, a nova geração de *chatbots* é mais sofisticada e inteligente. Eles são encontrados embutidos em dispositivos móveis, dispositivos relacionados à Internet das Coisas, entre outros, e estão cada dia mais presentes no dia a dia do ser humano. *Bots* como a “Siri”, da Apple e o “Google Assistant” da Google, por exemplo, que são assistentes virtuais bastante populares no meio dos dispositivos móveis, têm revolucionado a maneira com que tarefas online como compras, buscas, organização financeira, etc, são realizadas.

Uma das finalidades para o uso de *chatbots* que vem sendo explorada é o auxílio no monitoramento e na identificação de sintomas de transtornos mentais. Segundo Lucas et al. (2014), pessoas tendem a dar respostas mais realistas e a se abrir mais em conversas quando estão cientes de que estão conversando com um robô que parece um ser humano em vez de um ser humano real, o que facilita a obtenção de informações mais precisas do paciente.

Dentre os transtornos mentais cuja a ocorrência tem se tornado cada vez mais comum no Brasil e no mundo, está a depressão. De acordo com a Organização Mundial da Saúde (OMS) mais de 300 milhões de pessoas no mundo inteiro sofrem com a depressão, um número que cresceu rapidamente em apenas 10 anos (2005 - 2015) e que representa 4,4% da população do

planeta. Devido ao elevado número de pessoas com depressão no mundo já era de se esperar que no Brasil também houvesse um grande índice de pessoas depressivas. Ao todo estima-se que 5,8% da população brasileira seja afetada pela doença. Este índice faz com que o Brasil seja o país na América Latina que mais sofre com a depressão, à frente de Cuba e Paraguai, ambos com 5,5% e 5,2%, respectivamente (WORLD HEALTH ORGANIZATION, 2017). Além disso, a OMS também estima que aproximadamente 800 mil pessoas por ano cometem suicídio, que se torna o segundo principal motivo para mortes de pessoas de idade entre 15 a 29 anos (WORLD HEALTH ORGANIZATION, 2018).

Tendo em vista esse contexto, este trabalho propõe o desenvolvimento de um protótipo de *chatbot* que possa ajudar no combate de sintomas da depressão e na prevenção de suicídio.

1.1 Problema de Pesquisa

Considerando que uma grande parte da população mundial e, conseqüentemente, do Brasil sofre de algum tipo de transtorno mental, desenvolver soluções que aliem tecnologias como um robô de conversa, por exemplo, à terapia tradicional torna-se um caminho natural a ser percorrido para ajudar no tratamento desses transtornos. Sendo assim, “Do ponto de vista tecnológico, qual o impacto de um *chatbot* na qualidade de vida de uma pessoa com depressão?” e “Como desenvolver um *chatbot* que através de suas funções cognitivas ajude na diminuição dos sintomas da depressão e na prevenção ao suicídio?” são as questões que norteiam o desenvolvimento desta pesquisa.

1.2 Justificativa

Dos trabalhos encontrados na literatura que relacionam o uso de *chatbots* com o combate à depressão e ao suicídio, os principais como o *Woebot* (FITZPATRICK; DARCY; VIERHILE, 2017), *SimSensei Kiosk* (DEVAULT *et al.*, 2014), entre outros, são em língua inglesa, têm um enfoque em uma única abordagem teórica da psicologia e tentam, em sua maioria, reproduzir o comportamento de um terapeuta. Desta forma, trabalhos sobre *chatbots* que sejam capazes de conversar em Português do Brasil, que não tem como objetivo substituir a psicoterapia tradicional e ainda assim sejam relevantes, são uma contribuição de grande valor à área de *chatbots* e assistentes virtuais.

Tendo em vista todas essas informações, o produto desta pesquisa é o protótipo de um *chatbot* que através das suas funções cognitivas, conversando em Português do Brasil, ajude pessoas com sintomas de depressão influenciando de alguma forma no desenvolvimento da habilidade de comunicação do usuário, no aumento do interesse do mesmo em atividades que favoreçam uma melhoria na qualidade de vida e na redução da incidência de comportamento suicida. Portanto, entender os conceitos, definições e ferramentas necessárias ao desenvolvimento deste tipo de *chatbot* estão entre as principais funções desta pesquisa.

Por fim, deve-se ficar claro que o trabalho proposto não é um substituto para o terapeuta no tratamento da depressão e sim uma ferramenta de auxílio no combate aos sintomas desse transtorno que tem afetado uma parte significativa de pessoas no Brasil e no mundo, provando-se assim, o valor e a relevância desta pesquisa para o usuário e para a sociedade em geral.

1.3 Objetivos

1.3.1 Geral

O objetivo geral deste trabalho consiste em desenvolver o protótipo de um *chatbot mobile* que possa ser utilizado como ferramenta de auxílio e prevenção à depressão e ao suicídio.

1.3.2 Específicos

- (a) Realizar estudos Bibliográficos em plataformas específicas para o conhecimento científico e tecnológico, tais como: IEEE, Google Acadêmico, Springer, ACM;
- (b) Identificar a relação entre um aplicativo de troca de mensagens e o combate a alguns sintomas da depressão;
- (c) Analisar o *framework* IBM Watson Assistant e seus benefícios no desenvolvimento de um *chatbot*;
- (d) Desenvolver uma aplicação para Android pela qual o *chatbot* possa ser testado;
- (e) Testar a aplicação desenvolvida;
- (f) Analisar os resultados obtidos nos testes.

1.4 Organização do Trabalho

Além deste primeiro capítulo introdutório, este trabalho contém mais 4 capítulos.

São eles:

- Capítulo 2: Apresentação de conceitos e informações relevantes para o entendimento desta pesquisa, como Inteligência Artificial e desenvolvimento de *chatbots*, transtorno depressivo maior, além dos conceitos de desenvolvimento de aplicativos para Android;
- Capítulo 3: Estudo sobre a metodologia utilizada para o desenvolvimento do protótipo e da aplicação de teste;
- Capítulo 4: Exposição e análise dos resultados relacionados ao desenvolvimento e ao teste do protótipo;
- Capítulo 5: Conclusões finais acerca da contribuição desta pesquisa e sugestões para trabalhos futuros.

1.5 Trabalhos Relacionados

Um estudo realizado por pesquisadores da Universidade de Harvard (TOROUS; FRIEDMAN; KESHAVAN, 2014) concluiu que o uso de aplicativos em *smartphones* contribui para uma melhor captura de dados em tempo real do paciente, coisa que antes desse tipo de tecnologia era bem difícil de alcançar, e, além disso, representam uma oportunidade de desenvolvimento de novas modalidades de pesquisa na área de saúde mental. Sendo assim, algumas aplicações e pesquisas têm surgido e se destacado no ramo de combate à depressão e prevenção ao suicídio, como os aplicativos *Operation Reach Out* (MILITARY COMMUNITY AWARENESS INC, 2016) e *Ajuda Já*¹, o *bot* terapeuta utilizado na pesquisa *Reporting Mental Health Symptoms: Breaking Down Barriers to Care with Virtual Human Interviewers* (LUCAS *et al.*, 2017), e o *chatbot* *Woebot* (FITZPATRICK; DARCY; VIERHILE, 2017).

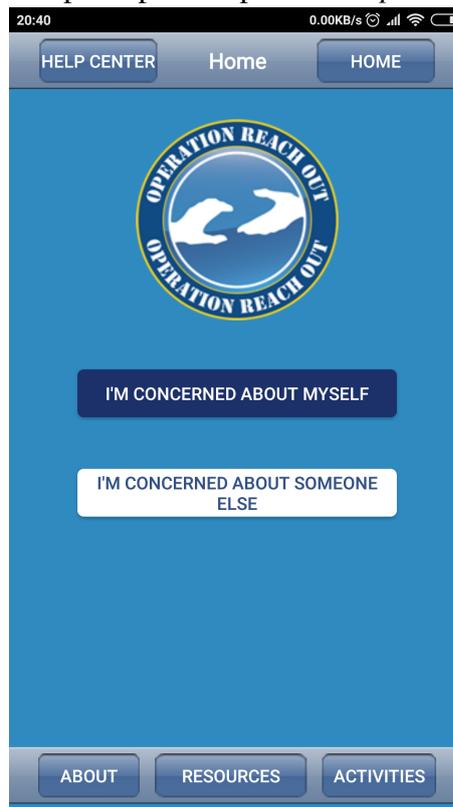
1.5.1 *Operation Reach Out* (MILITARY COMMUNITY AWARENESS INC, 2016)

Disponível para Android e iOS, o aplicativo *Operation Reach Out* foi desenvolvido pela *Military Community Awareness* para militares mas pode ser utilizado por qualquer pessoa que esteja pensando em suicídio (MILITARY COMMUNITY AWARENESS INC, 2016).

Após aceitar os termos de uso do aplicativo, na tela principal(demonstrada na Figura

¹ <http://appsdobem.com.br/#ajuda-ja>

Figura 1 – Tela principal do aplicativo *Operation Reach Out*

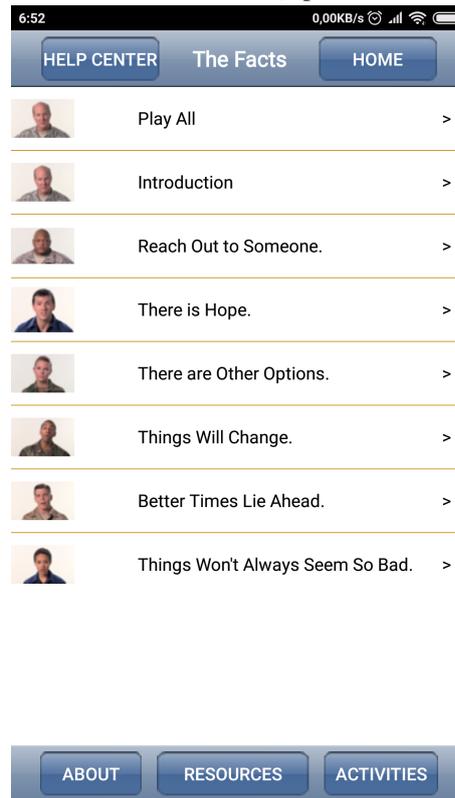


Fonte: Autoria própria.

1) o usuário faz uma seleção de para quem o conteúdo é necessário (se para um amigo ou pro próprio usuário) de forma que o texto da próxima tela é personalizado de acordo com a seleção.

Na tela seguinte(Figura 2), o usuário vê uma lista de vídeos que podem ser assistidos em qualquer ordem com temas como “Entre em contato com alguém”, “Há esperança”, “Existem outras opções”, etc, que levam a pessoa a reconsiderar escolher viver. Nesta tela ainda é possível observar dois conjuntos de botões: um superior e um inferior. No superior, o usuário possui a opção de voltar para a tela principal (“Home”) ou clicar em “Help center” para ir para uma tela onde ele pode contactar um dentre uma lista de contatos que possui como padrão o contato do centro de prevenção de suicídio mas que pode aumentar à medida que o usuário acrescenta seus próprios contatos de emergência. No conjunto de botões inferior, o usuário tem a opção de ver informações sobre o aplicativo (“About”), uma lista de links úteis relacionados à prevenção ao suicídio (“Resources”) e sugestões de atividades (“Activities”), como por exemplo “Ajudar alguém na biblioteca”, que podem ser classificadas pelo usuário onde as atividades que ele não gostou não vão ser sugeridas novamente.

Quando um vídeo é selecionado ou a opção de executar todos (“Play all”) é selecionada, o vídeo é executado automaticamente em tela cheia. Quando o botão “voltar” do

Figura 2 – Lista de vídeos (*Operation Reach Out*)

Fonte: Autoria própria.

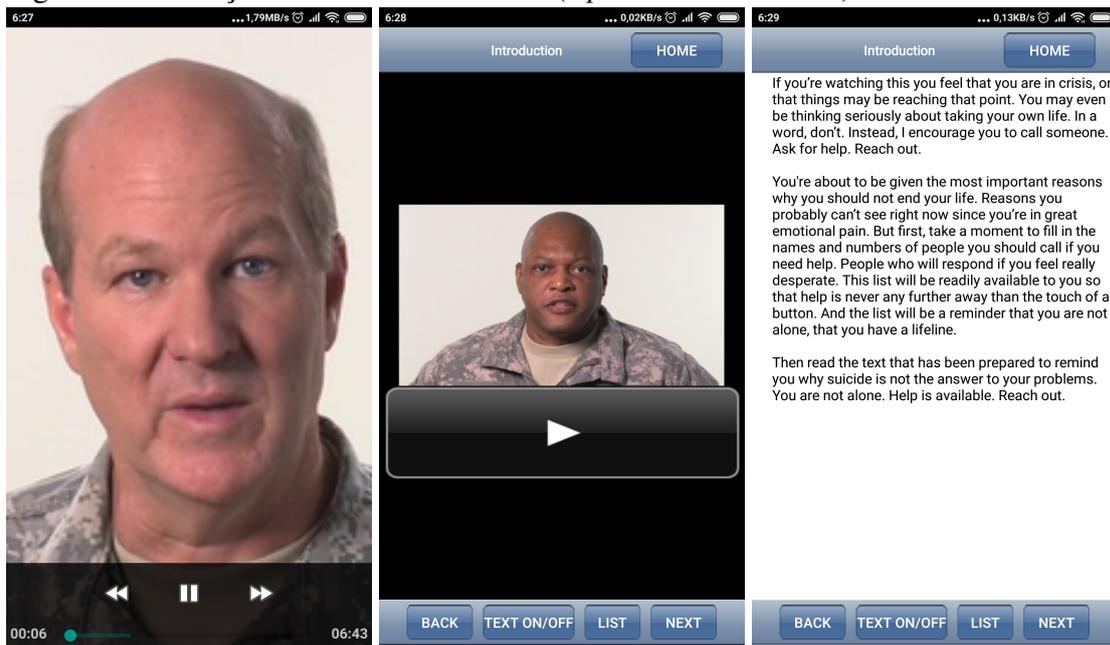
dispositivo é clicado durante a execução do vídeo, uma tela sobre o vídeo atual é exibida com um menu na parte inferior onde o usuário pode navegar entre os vídeos, indo para o próximo da lista (“Next”) ou voltando pro anterior (“Back”), escolher entre visualizar uma versão textual do que é dito no vídeo ou visualizar o próprio vídeo (“Text On/Off”) e retornar para a visualização da lista de vídeos (“List”). A Figura 3 mostra como são representados estes elementos.

Em 2016 o aplicativo foi listado entre os 13 melhores aplicativos do ano relacionados à depressão pelo Healthline.com e também possui, atualmente, 4 de 5 estrelas na Google Play.

1.5.2 Ajuda Já

Considerando ainda a área de aplicações que ajudam pessoas em um momento pré tentativa de suicídio, desenvolvedores brasileiros incentivados pela incubadora sem fins lucrativos de projetos sociais digitais no Brasil, *Apps do Bem*, criaram o aplicativo “Ajuda já”. Neste aplicativo, desenvolvido para plataforma IOS, o usuário se depara primeiramente com uma tela de cadastro onde ele registra 3 contatos de emergência. Uma vez confirmada a inserção desses contatos no aplicativo, o usuário consegue contactar facilmente uma pessoa de confiança na hora de uma tentativa de suicídio, clicando no botão “Ajuda Já”. A Figura 4 mostra as

Figura 3 – Exibição do vídeo selecionado (*Operation Reach Out*)



Fonte: Autoria própria.

principais telas deste aplicativo.

Figura 4 – Aplicativo "Ajuda Já"



Fonte: <https://itunes.apple.com/br/app/ajuda-já/id1200600469?mt=8>

1.5.3 Reporting Mental Health Symptoms: Breaking Down Barriers to Care with Virtual Human Interviewers (LUCAS et al., 2017)

Um trabalho altamente relevante para o desenvolvimento de aplicações de conversa automatizada voltadas para saúde mental foi o desenvolvido por Lucas et al. (2017). Os pesquisadores descobriram que a maioria dos soldados, quando voltavam de combate, tinham dificuldade de reportar sintomas de Transtorno de Estresse Pós-Traumático (PTSD) por medo de afetarem suas carreiras militares. Sendo assim, eles desenvolveram um *bot* terapeuta chamado *SimSensei Kiosk* (DEVAULT et al., 2014) que entrevistava os militares com perguntas gerais e perguntas relacionadas aos sintomas de PTSD, e que ainda exibia um comportamento característicos de seres humanos como acenar com a cabeça algumas vezes. Os pesquisadores reportam que o *bot* se mostrou muito eficaz em revelar mais sintomas de PTSD do que quando os militares apenas

respondiam a um questionário escrito.

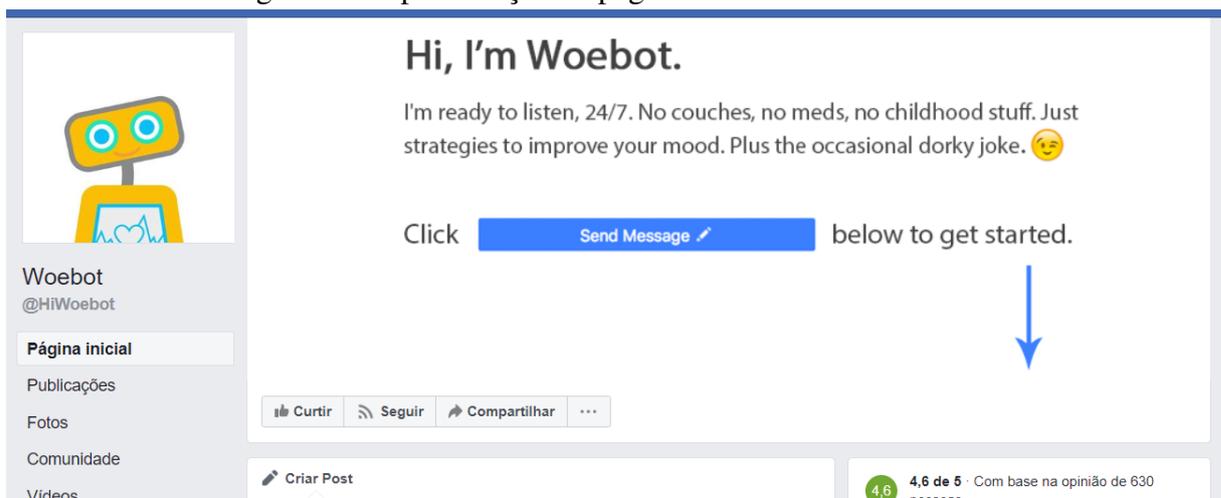
1.5.4 *Woebot (FITZPATRICK; DARCY; VIERHILE, 2017)*

Aplicativos baseados em Terapia Cognitivo Comportamental(TCC) se tornaram muito comuns quando o assunto é depressão e ansiedade. Esses aplicativos utilizam esses conceitos para ajudar o usuário em várias atividades do dia a dia através de instruções sobre respiração ou análise da qualidade do sono, por exemplo, de forma a diminuir os níveis de estresse e ansiedade do indivíduo.

Levando isso em consideração e baseando-se no fato de que as pessoas tendem a se abrir mais conversando com *chatbots* do que com pessoas reais, alguns pesquisadores da Universidade de Stanford criaram o *Woebot* (FITZPATRICK; DARCY; VIERHILE, 2017). Esse *chatbot* é uma aplicação desenvolvida utilizando AWS Lambda e Node.js para o Facebook Messenger e que combina técnicas de terapia cognitivo comportamental e processamento de linguagem natural fazendo com que os usuários se expressem mais naturalmente do que se estivessem em uma sessão real de terapia.

Para iniciar a interação com o *chatbot*, o usuário só precisa enviar uma mensagem à página do *Woebot* no Facebook (demonstrada na Figura 5) e logo em seguida o próprio *chatbot* já dá início à interação.

Figura 5 – Apresentação da página do *Woebot* no Facebook

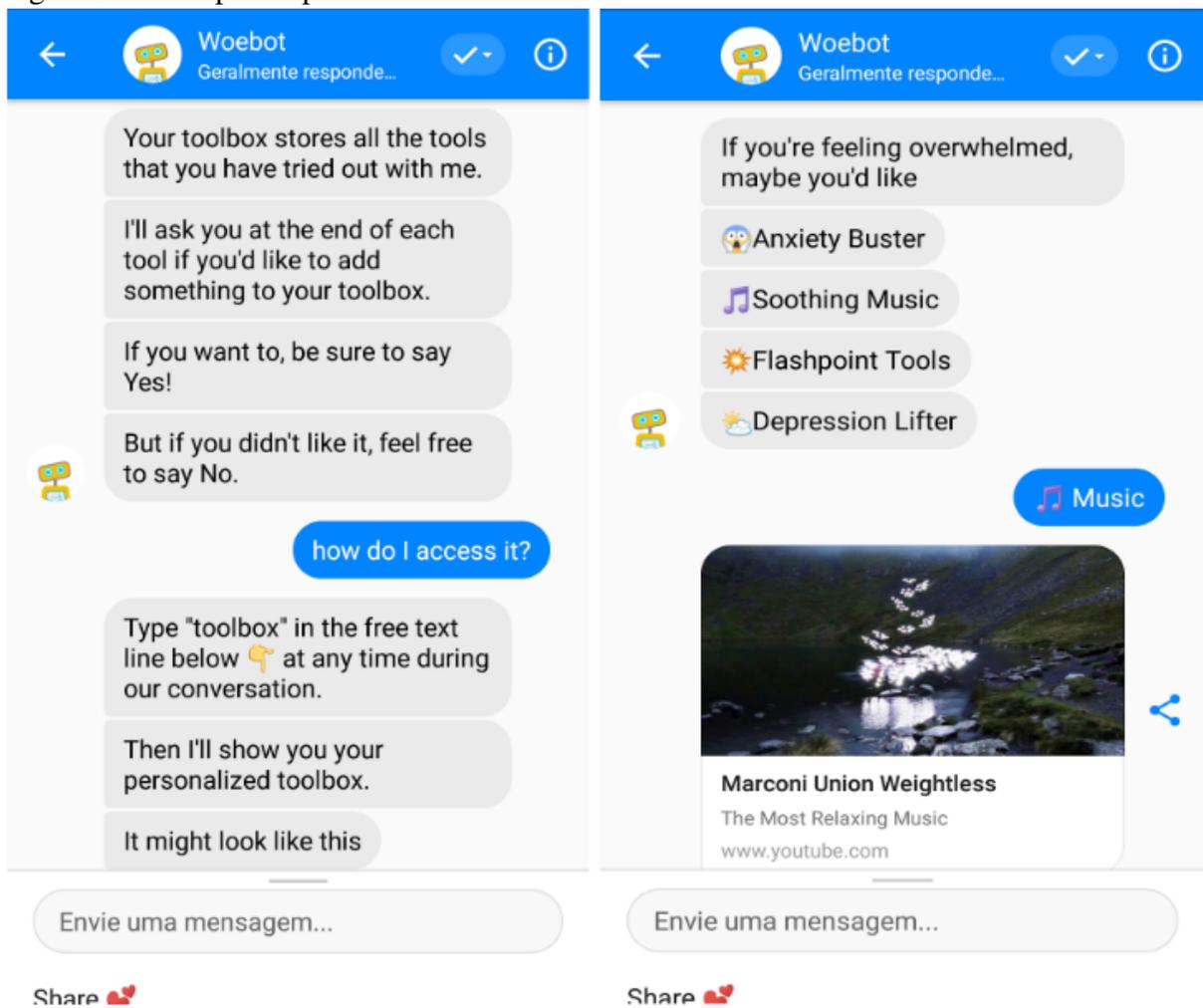


Fonte: FITZPATRICK; DARCY; VIERHILE (2017)

Por ser construído para Facebook Messenger, a tela que o usuário interage com o *chatbot* é bem familiar pois é exibida como qualquer outra conversa no Messenger. Em um primeiro contato o *chatbot* explica seu funcionamento, a utilidade dos botões (que ajudam a

direcionar a conversa de acordo com o que o usuário esteja precisando) e como acessar o menu. Durante as conversas o *chatbot* permite que a qualquer momento o usuário escolha a melhor hora para o “*check in*” de forma que todo dia o *chatbot* inicia uma conversa aproximadamente no mesmo horário. Estas funcionalidades estão exemplificadas na Figura 6.

Figura 6 – Exemplo de primeiro contato com o *Woebot* e escolha de atividade baseada em TCC

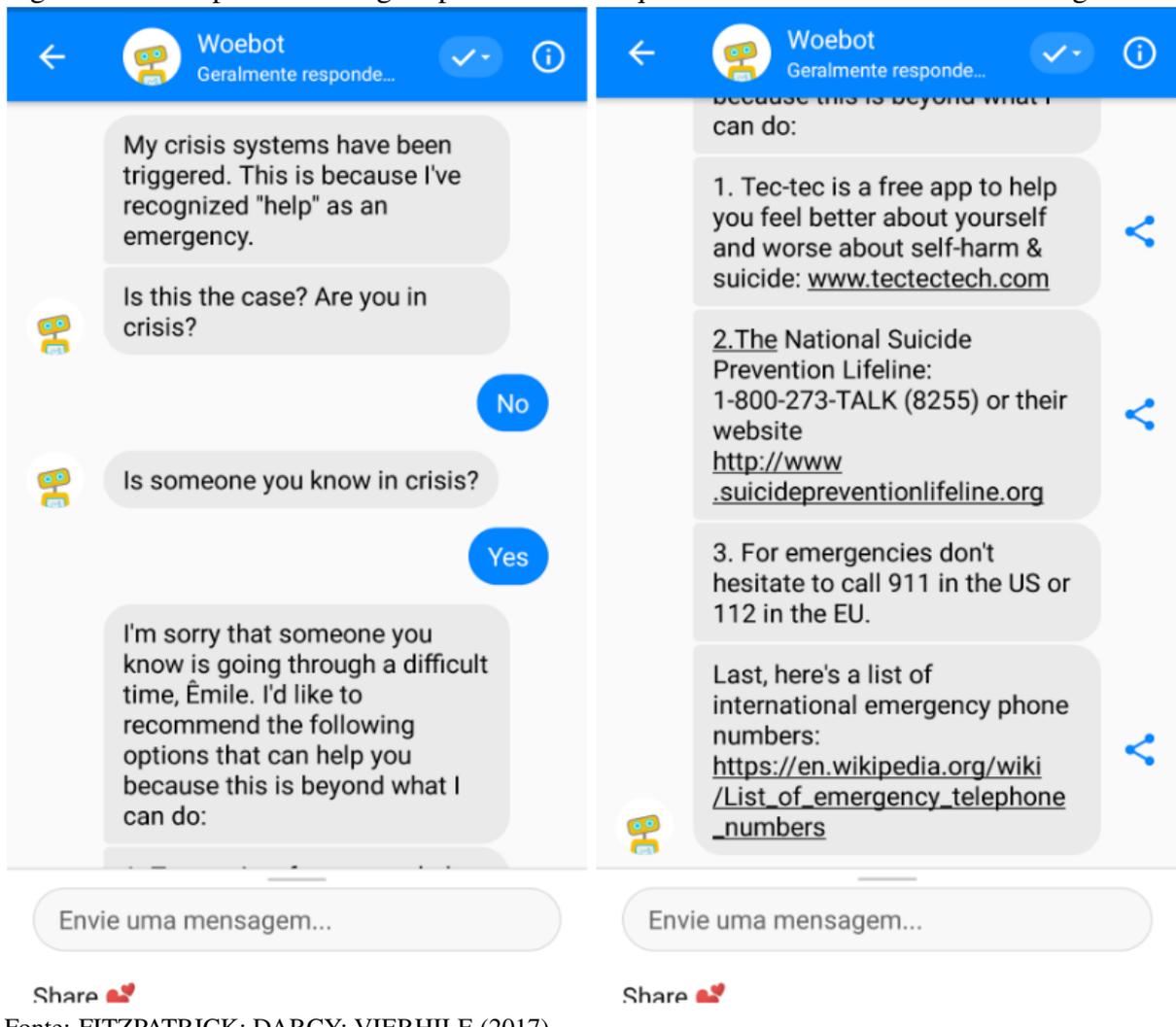


Fonte: FITZPATRICK; DARCY; VIERHILE (2017)

Em geral as conversas iniciam com o *chatbot* perguntando como o usuário se sente naquele momento, o que contribui para registrar a evolução do humor da pessoa durante a semana. De acordo com o humor que o usuário registra, o *Woebot* vai sugerindo diferentes atividades baseadas em TCC para diminuir ansiedade. Além disso, em casos onde o *chatbot* detecta que o usuário digitou algo que remete a uma crise de ansiedade ou depressão, ou a palavra-chave 'S.O.S.', ele exibe uma mensagem padrão com números e sites importantes onde o usuário pode adquirir ajuda para esse momento de emergência, como mostrado na Figura 7.

Para testar o *Woebot*, os pesquisadores separaram um grupo de 70 pessoas, com

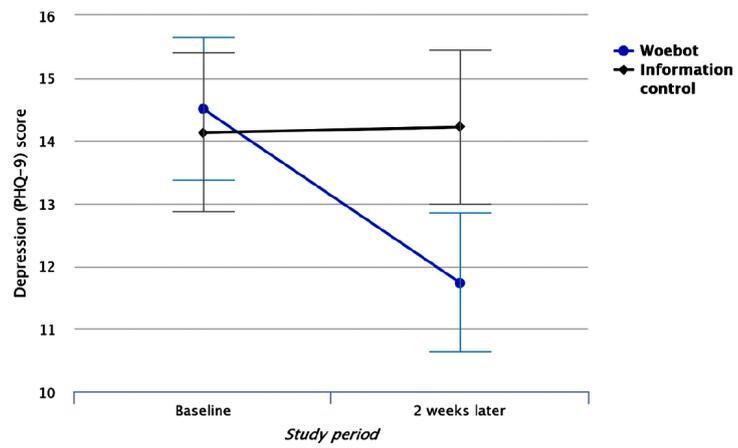
Figura 7 – Exemplo de mensagem padrão exibida quando o *Woebot* detecta uma emergência



Fonte: FITZPATRICK; DARCY; VIERHILE (2017)

idade entre 18 a 28 anos, em dois grupos: os que interagiram com o *Woebot* e os que leram um *ebook* sobre depressão em estudantes universitários. O teste durou duas semanas e todos os participantes responderam, antes e depois dos testes, três questionários indicados pelo *DSM - IV* para monitorar os sintomas da depressão e da ansiedade: o *The Patient Health Questionnaire-9* (Questionário Sobre a Saúde do Paciente - PHQ-9), *Generalized Anxiety Disorder-7* (Transtorno Geral de Ansiedade - GAD-7) e *Positive and Negative Affect Scale* (Escala Positiva e Negativa de Afeto - PANAS). De fato, os pesquisadores reportaram que os estudantes que conversaram com o *Woebot* sentiram menos stress e ansiedade depois de duas semanas do que os que leram o livro, como está indicado na Figura 8.

Figura 8 – Resultados dos testes do *Woebot*
Change in depression by group



results.PNG

Fonte: FITZPATRICK; DARCY; VIERHILE (2017)

2 FUNDAMENTOS TEÓRICOS

Este capítulo apresenta a revisão bibliográfica realizada no presente trabalho sobre temas que são fundamentais para a compreensão da metodologia aplicada no desenvolvimento do mesmo. Serão abordados os principais conceitos do Transtorno Depressivo Maior e definições relacionadas à Processamento de Linguagem Natural e Inteligência Artificial com alguns de seus subtemas: aprendizado de máquina e *Cognitive Computing*. Além disso, serão abordados também conceitos relacionados ao IBM Watson, um dos principais *frameworks* para desenvolvimento de *chatbots*, e conceitos relativos ao sistema operacional Android e a plataforma de serviços Firebase.

2.1 Transtorno Depressivo Maior

De acordo com o Manual Diagnóstico e Estatístico de Transtornos Mentais AMERICAN PSYCHIATRIC ASSOCIATION (2014), o Manual Diagnóstico e Estatístico de Transtornos Mentais V (DSM V), a depressão caracteriza-se em sua essência por humor depressivo e perda de interesse ou prazer em coisas que antes a pessoa gostava de fazer. Isso podendo ocorrer durante quase todo o tempo, por pelo menos duas semanas seguidas ou ainda se estender por meses e até por mais tempo. Entretanto, é importante notar que esse humor depressivo se difere da simples tristeza, que é um humor passageiro, momentâneo. Outra grande característica de quem sofre com depressão é a redução na produção de um neurotransmissor chamado Serotonina, que é o responsável pela transmissão da sensação de bem-estar, fazendo com que as áreas do cérebro responsáveis pelo prazer nas coisas se tornem cada vez mais inativas resultando na dificuldade do indivíduo em se sentir bem.

Os sintomas específicos da depressão, segundo o DSM V, são:

- Humor deprimido na maioria dos dias, quase todos os dias (p. ex.: sente-se triste, vazio ou sem esperança) por observação subjetiva ou realizada por terceiros (Nota: em crianças e adolescentes pode ser humor irritável);
- Acentuada diminuição do prazer ou interesse em todas ou quase todas as atividades na maior parte do dia, quase todos os dias (indicado por relato subjetivo ou observação feita por terceiros);
- Perda ou ganho de peso acentuado sem estar em dieta (p.ex. alteração de mais de 5% do peso corporal em um mês) ou aumento ou diminuição de apetite quase todos os dias (Nota:

em crianças, considerar incapacidade de apresentar os ganhos de peso esperado);

- Insônia ou hipersônia quase todos os dias;
- Agitação ou retardo psicomotor quase todos os dias (observável por outros, não apenas sensações subjetivas de inquietação ou de estar mais lento); Fadiga e perda de energia quase todos os dias;
- Sentimento de inutilidade ou culpa excessiva ou inadequada (que pode ser delirante), quase todos os dias (não meramente auto recriminação ou culpa por estar doente);
- Capacidade diminuída de pensar ou concentrar-se ou indecisão, quase todos os dias (por relato subjetivo ou observação feita por outros);
- Pensamentos de morte recorrentes (não apenas medo de morrer), ideação suicida recorrente sem um plano específico, ou tentativa de suicídio ou plano específico de cometer suicídio;

Um outro critério de diagnóstico apresentado no DSM V traz que algumas áreas importantes da vida do indivíduo como o funcionamento social, profissional, etc, sofrem prejuízo devido à depressão. Habilidades sociais como manifestar opinião, fazer, aceitar e recusar pedidos, compartilhar emoções, lidar com críticas, etc, podem ser gravemente alteradas de acordo com o nível de depressão do indivíduo.

Outra questão que deve ser avaliada quando se fala de depressão é o risco de suicídio. Segundo o DSM V, esse risco está presente durante todo período em que a pessoa está com depressão e aumenta quando o indivíduo é do sexo masculino, solteiro ou mora sozinho e tem sentimentos derivados da desesperança. Além disso, se a pessoa já teve episódios de tentativa de suicídio o risco de suicídio completado aumenta, mesmo que na maioria dos casos de suicídio o indivíduo não tenha histórico de tentativas.

2.2 Processamento de Linguagem Natural

Processamento de Linguagem Natural (PLN), de acordo com Liddy (1998), é uma gama de técnicas computacionais para analisar e representar textos em linguagem natural, em um ou mais níveis de análise linguística, com a finalidade de obter processamento de linguagem semelhante ao humano para uma série de tarefas ou aplicações. Este conceito fica ainda mais esclarecido na visão de Vieira e Lopes (2010, p. 184) que afirma que Processamento de Linguagem Natural é “uma área de Ciência da Computação que estuda o desenvolvimento de programas de computador que analisam, reconhecem e/ou geram textos em linguagens humanas, ou linguagens naturais”. Sendo assim, Processamento de Linguagem Natural está relacionado

como programa de computador que compreendem textos em linguagem natural.

A maioria dos sistemas de PLN possuem todos ou alguns dos seguintes níveis de análise linguística relatados por Liddy (1998):

- **Fonológico:** Interpretação da pronúncia das palavras. Este nível só é importante para sistemas de reconhecimento de voz, não tendo sentido ser empregado em aplicações que só utilizam texto.
- **Morfológico:** Análise dos componentes das palavras, incluindo prefixos, sufixos e radical. Neste nível, as palavras são reduzidas à sua forma mais simples para evitar que palavras que estão no plural, por exemplo, não sejam consideradas morfologicamente diferentes da mesma palavra só que no singular. Além disso, este nível se mostra bem mais recompensador quando aplicado a idiomas que possuem morfologia mais rica, se comparados ao Inglês.
- **Léxico:** Envolve análise do sentido léxico das palavras e marcações de *part of speech*. Na análise linguística, este nível é útil pois levanta informações sobre partes do texto como por exemplo, se a parte analisada é um verbo, um substantivo, etc
- **Sintático:** Análise das palavras em uma sentença a fim de descobrir a estrutura gramatical da mesma. É neste nível que, através da saída gerada pelo nível anterior, é possível descobrir a relação entre as palavras, como por exemplo o substantivo ‘bolsa’ e o adjetivo ‘azul’ que, combinados, se tornam uma chave de busca mais precisa do que se utilizadas as duas palavras individualmente
- **Semântico:** Determina os possíveis significados de uma frase, já que uma palavra pode ter mais de um significado, através da desambiguação de palavras.
- **Discursivo:** Examina a estrutura e o significado de um texto no todo, e não apenas de uma sentença. Este nível é importante pois nele pode ser descoberto a importância de uma frase no documento todo, por exemplo esta frase é uma opinião? é uma conclusão? ou será um fato?
- **Pragmático:** Busca entender o propósito do uso da linguagem em determinada situação, especialmente os aspectos da linguagem que envolveram algum conhecimento de mundo. O resultado desta análise pode ser incorporado à interface do usuário para facilitar a comunicação e a identificação do usuário com o sistema.

O processamento de Linguagem Natural pode ser aplicado a muitos sistemas que envolvem qualquer tipo de textos em linguagem natural como sistemas de recuperação de

informação, *chatbots* e outros sistemas que respondem perguntas, sistemas de reconhecimento de fala, entre outros.

2.3 Inteligência Artificial

O campo da Inteligência Artificial surgiu após a Segunda Guerra Mundial e desde então tem avançado de forma a se tornar uma das principais áreas da computação moderna. Mas afinal, o que é Inteligência Artificial? De acordo com Russell e Norvig (2014), esta área de estudo diz respeito à compreensão e, principalmente, à construção de entidades computacionais inteligentes. Os autores também descrevem que IA possui duas abordagens principais: uma relacionada aos processos de pensamento e comportamento humano e outra voltada para os processos de pensamento e comportamento em uma visão racionalista. Essas abordagens são melhor definidas de acordo com o quadro de conceitos de Inteligência Artificial demonstrado na Figura 9.

Figura 9 – Quadro de conceitos de Inteligência Artificial

Pensando como um humano	Pensando racionalmente
<p>“O novo e interessante esforço para fazer os computadores pensarem (...) <i>máquinas com mentes</i>, no sentido total e literal.” (Haugeland, 1985)</p> <p>“[Automatização de] atividades que associamos ao pensamento humano, atividades como a tomada de decisões, a resolução de problemas, o aprendizado...” (Bellman, 1978)</p>	<p>“O estudo das faculdades mentais pelo uso de modelos computacionais.” (Charniak e McDermott, 1985)</p> <p>“O estudo das computações que tornam possível perceber, raciocinar e agir.” (Winston, 1992)</p>
Agindo como seres humanos	Agindo racionalmente
<p>“A arte de criar máquinas que executam funções que exigem inteligência quando executadas por pessoas.” (Kurzweil, 1990)</p> <p>“O estudo de como os computadores podem fazer tarefas que hoje são melhor desempenhadas pelas pessoas.” (Rich and Knight, 1991)</p>	<p>“Inteligência Computacional é o estudo do projeto de agentes inteligentes.” (Poole <i>et al.</i>, 1998)</p> <p>“AI... está relacionada a um desempenho inteligente de artefatos.” (Nilsson, 1998)</p>

Fonte: Russell e Norvig (2013)

A partir desses conceitos surgem vários subcampos da área de Inteligência Artificial como aprendizado de máquina, computação cognitiva, entre outros, onde são estudadas e desenvolvidas soluções para uma grande variedade de problemas.

2.3.1 *Aprendizado de máquina*

O conceito de aprendizagem de máquina, do inglês *Machine Learning*, está ligado diretamente à capacidade de uma máquina aprender sobre dados que lhe forem apresentados a partir de dados com padrões semelhantes. Em (ALPAYDIN, 2009) o autor explica que, em geral, existe um processo que explica os dados observados e que apesar de a máquina não poder determinar todo o processo, pode determinar uma ótima aproximação dele através dos padrões e regularidades dos dados coletados. Ele afirma também que aprendizado de máquina nada mais é do que programar computadores para melhorar seu comportamento usando sua experiência passada.

Quanto aos tipos, Russell e Norvig (2013) classificam em quatro:

- **Aprendizagem não supervisionada:** não existem exemplos prévios
- **Aprendizagem por reforço:** o aprendizado acontece a partir de uma série de reforços (recompensas ou punições). O autor cita como exemplo disso um agente de táxi onde “falta de gorjeta ao final de uma corrida dá ao agente do táxi a indicação de que algo saiu errado”
- **Aprendizagem supervisionada:** a partir de alguns exemplos prévios de entradas e suas consequentes saídas, aprende-se uma função geral que mapeia entrada para saída
- **Aprendizagem semi supervisionada:** esse tipo de aprendizagem está entre a supervisionada e não supervisionada e ocorre quando em parte dos exemplos fornecidos a saída não é conhecida.

As técnicas de aprendizado de máquina são amplamente empregadas em jogos, reconhecimento de fala e de texto, sistemas de recomendação, entre outros.

2.3.2 *Cognitive Computing*

Um dos desdobramentos mais relevantes da área de Inteligência Artificial é com certeza a Computação Cognitiva, ou do inglês, *Cognitive Computing*. Seu principal objetivo e significado é simular em serviços computacionais o pensamento humano, utilizando técnicas de aprendizado de máquina e processamento de linguagem natural, processando dados previamente coletados para auxiliar no processo de tomada de decisão inteligente em uma espécie de representação computacional do cérebro. Esses conceitos, que já foram até mesmo eternizados no cinema em filmes como “*Her*” e “*Ex-Machina*”, têm aplicação em inúmeras áreas de conhecimento

como sistemas de recomendação, sistemas médicos relacionados à diagnóstico, e muitos outros que vão surgindo à medida que as tecnologias e *frameworks* evoluem. De acordo com Kelly 2015, sistemas cognitivos englobam cinco capacidades principais:

1. Criar um envolvimento humano mais profundo;
2. Dimensionar e elevar a experiência
3. Infundir produtos e serviços com cognição
4. Permitir processos e operações cognitivas
5. Incentivar a exploração e a descoberta

2.4 IBM Watson

Desde que os *chatbots* começaram a ser uma grande ferramenta de influência na forma de atendimento ao cliente e em geral, a forma de desenvolvê-los também evoluiu. Em vez de ter que programar códigos enormes e complexos, e que às vezes precisam ser específicos para algum canal (como Facebook, Whatsapp, Slack, etc), é possível desenvolver um *bot* com *frameworks* que permitem que o mesmo *bot* desenvolvido possa ser integrado a qualquer uma destes canais, para ser utilizado pelo usuário. Assim, de forma a facilitar o desenvolvimento desta pesquisa, buscou-se um *framework* de desenvolvimento dentre os mais populares do mercado, como por exemplo Dialogflow¹ da Google (antigo API.ai), Microsoft Bot Framework², o IBM Watson³, entre outros (DAVYDOVA, 2017) . O *framework* escolhido foi o IBM Watson pois, diferente do Dialogflow, por exemplo, que até Fevereiro/2018 não tinha suporte para criar *bots* que interagem em Português do Brasil, ele possui suporte para conversas nesta linguagem desde 2015 (STENBECK, 2015). Além disso, se comparado a outros *frameworks* populares como o Microsoft Bot Framework, o IBM Watson possui uma integração muito simples com aplicações para Android, além de ter um material bem maior de estudo e pesquisa sobre este *framework* disponível na internet.

O IBM Watson é resultado de uma pesquisa realizada pela IBM, que se iniciou em 2007, com o objetivo de criar um sistema de computador que pudesse concorrer com outros jogadores no jogo de perguntas e respostas *Jeopardy*. Após anos de pesquisa, em 2011, o Watson fez sua primeira aparição competindo no programa de televisão *Jeopardy!* e ganhando dos dois melhores jogadores do jogo (FERRUCCI, 2012). Logo após, o Watson se tornou uma

¹ <https://dialogflow.com/>

² <https://dev.botframework.com/>

³ <https://www.ibm.com/cloud/watson-assistant/>

série de API's que estão disponibilizadas no IBM Cloud (MAZON, 2018a) incluindo o Watson Conversation que em 2018 passou-se a se chamar IBM Watson Assistant (MAZON, 2018b).

2.4.1 IBM Watson Assistant

O Watson Conversation, ou como é conhecido atualmente Watson Assistant, é uma API para criação de *chatbots*, disponível no IBM Cloud, e que tem, de acordo com Mazon (2018a), "uma interface simples para que até mesmo uma pessoa que não seja de TI consiga desenvolver e ensinar conteúdo ao *bot*". Esta API possui ainda modelos de Processamento de Linguagem Natural próprios para vários idiomas, além de se utilizar de alguns conceitos comuns à maioria das APIs de desenvolvimento de *chatbots* como os conceitos de Intenções (*Intents*) e Entidades (*Entities*), e de apresentar outros conceitos, como Diálogo, *Slots* e Digressão.

Para Mazon (2018a), uma *intent* ou intenção "trata-se da ação atrelada às perguntas realizadas pelo usuário. Isto é, o que o usuário procura ao falar algo;". Sendo assim, considerando o exemplo citado por Mazon (2018a) de um usuário que interage com um *chatbot* para pedir uma pizza, uma vez que o usuário falasse "Eu quero uma pizza", o *chatbot* entenderia que a intenção por trás do que ele falou é "pedir pizza". Esta intenção é cadastrada no Watson Assistant de acordo com a Figura 10. Porém, apesar de existirem várias frases onde uma mesma intenção poderia ser identificada, no Watson Assistant só é necessário dar algumas frases de exemplo para que o sistema consiga identificar outras com intenções comuns (MAZON, 2018a). Assim, se o usuário falasse, como cita Mazon (2018a), "tô querendo uma pizza aí meu", o *chatbot* identificaria a intenção #pedir_pizza mesmo que este exemplo não estivesse nos exemplos de treinamento desta *intent*. O Assistant também disponibiliza uma lista de *intents* pré treinadas na aba 'Content Catalog', para que o desenvolvedor não 'perca tempo' criando *intents* comuns à maioria dos *chatbots* desenvolvidos atualmente (MAZON, 2018b).

De acordo com Faria (2017), "Se as intenções expressam funcionalidades, entidades expressam parâmetros para a execução de uma funcionalidade". Assim, *entities*, ou entidades, são variáveis chave no texto que contém detalhes sobre a intenção do usuário. Ao considerar o exemplo anterior, do *bot* que recebe um pedido de pizza, Mazon (2018a) dá como exemplo de entidades o sabor da pizza, o tipo da massa, a data da entrega, entre outros. Ao cadastrar uma entidade, pode se colocar tanto sinônimos quanto expressões regulares, como no caso do CEP, por exemplo, onde ao cadastrar seria colocado '[0-9]{5}-[0-9]{3}'. No Watson Assistant também é possível utilizar entidades pré definidas, as *system entities*, que não precisam ser

Figura 10 – Exemplo da *intent* "pedir pizza"

The screenshot shows the configuration page for the intent '#pedir_pizza'. It includes fields for the intent name, a description, and a list of user examples. The examples listed are: 'quero pedir uma pizza', 'eu gostaria de fazer um pedido', 'iniciar pedido de pizza', 'me vê uma pizza', and 'por favor quero fazer meu pedido'. There is an 'Add example' button and a list of 'User examples (4)'.

Fonte: Mazon (2018a)

treinadas e só precisam ser habilitadas para começarem a serem usadas no desenvolvimento do *bot*, como é o caso da entidade @sys-date que, no exemplo do pedido de pizza, garante que o *bot* vai entender a data mesmo se o usuário falar que quer a pizza 'amanhã', ou 'no natal' ou 'Sexta-Feira' (MAZON, 2018a). No Watson Assistant é possível habilitar as *system entities* de acordo com a Figura 11.

Figura 11 – Onde habilitar as *system entities* no Watson Assistant

The screenshot shows the 'System entities' configuration page in Watson Assistant. It lists several system entities with their descriptions and status toggles. The '@sys-date' entity is currently turned on.

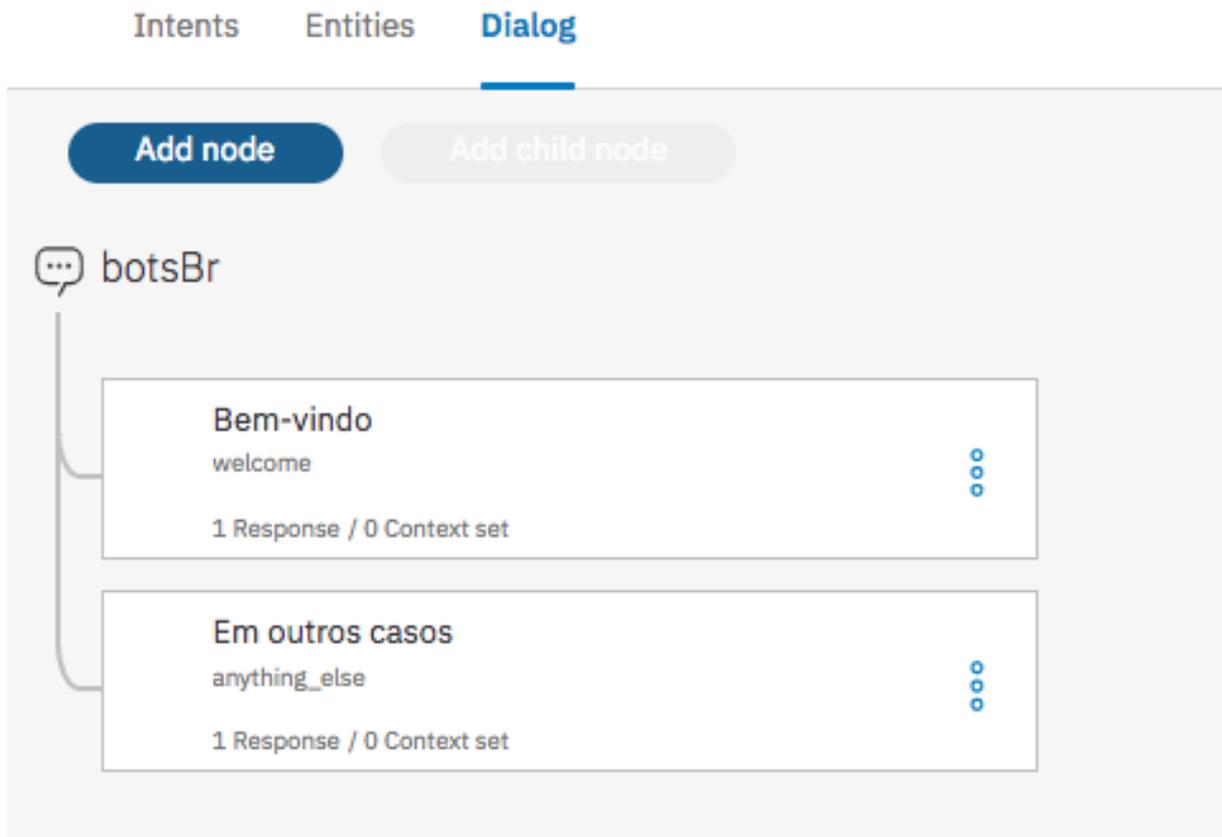
Name (5)	Description	Status
> @sys-currency	Extracts currency values from user examples including the amount and the unit. (20 cents)	Off
> @sys-date	Extracts date mentions (Sexta-feira)	On
> @sys-number	Extracts numbers mentioned from user examples as digits or written as numbers. (21)	Off
> @sys-percentage	Extracts amounts from user examples including the number and the % sign. (15%)	Off
> @sys-time	Extracts time mentions (em 10)	Off

Fonte: Mazon (2018a)

Outro conceito do Watson Assistant é o Diálogo, que nada mais é do que uma espécie de 'árvore' com nós interconectados onde é montado o fluxo de conversa do *bot*, como mostra a Figura 12. Cada nó só é visitado caso uma condição, que no Assistant pode ser uma intenção, entidade, ou alguma outra variável ou teste lógico, for reconhecida. Em cada nó pode existir ainda várias opções de resposta que vão ser exibidas, sequencialmente ou aleatoriamente, cada

vez que aquele nó for visitado e a condição dele for reconhecida. A estrutura de um nó é definida como na Figura 13. Cada nó pode possuir também nós filhos, que só vão ser visitados caso o nó 'pai' seja visitado primeiro. Este tipo de estrutura é o que faz com que vários fluxos de conversa possam ser criados em um mesmo diálogo (MAZON, 2018a).

Figura 12 – Exemplo de diálogo no Watson Assistant



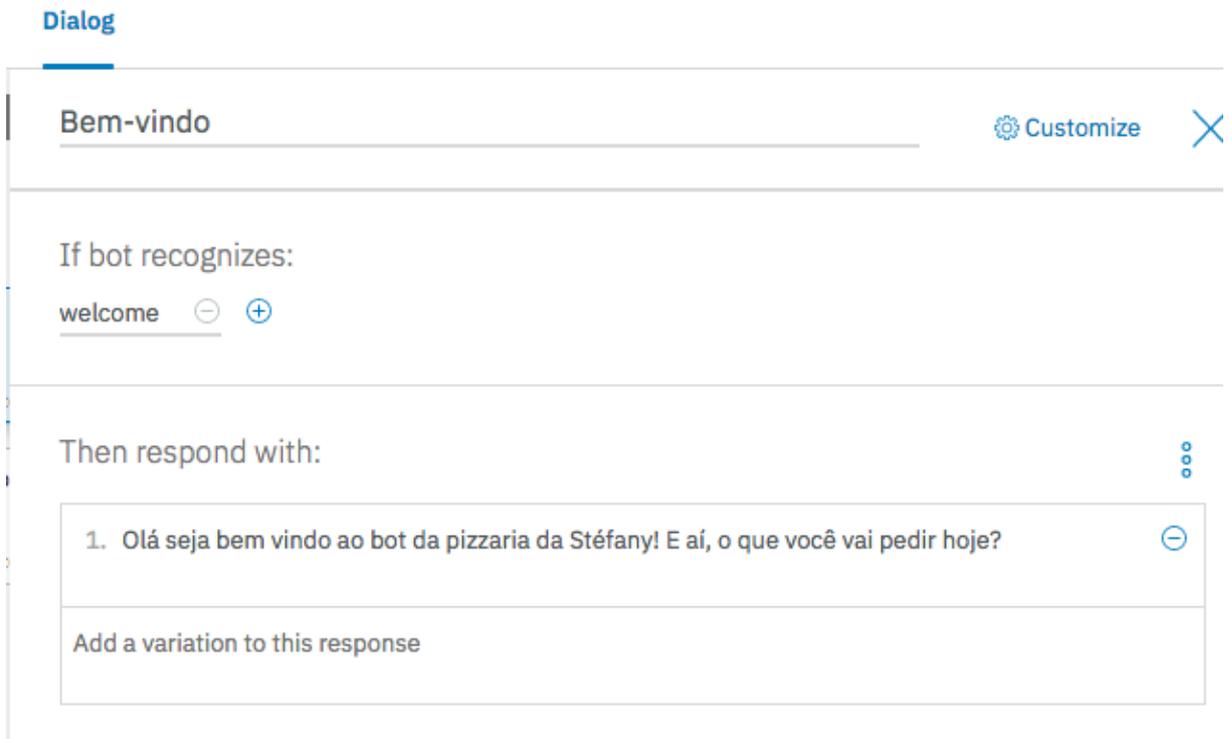
Fonte: Mazon (2018a)

Em casos onde várias variáveis diferentes precisam ser coletadas, é possível fazer isso no mesmo nó através do uso de *Slots*. De acordo com Mazon (2018a)

Slots são utilizados para coleta de informações ao longo da conversa, de um modo alternativo à formulários. O mais legal é que ao interagir com o bot o usuário não precisa enviar as informações em uma ordem específica pois o slot trata variável por variável.

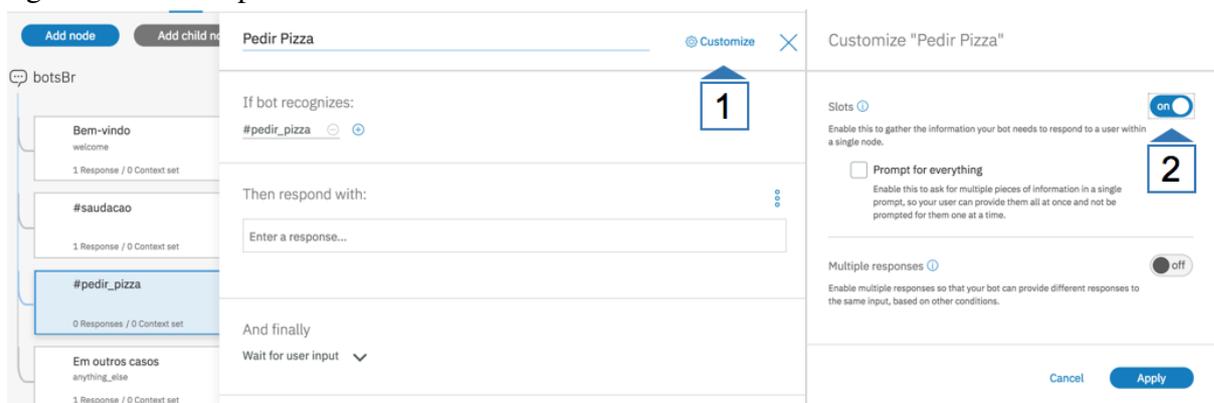
Para inserir *slots* em um nó, deve-se ser habilitada a utilização dos mesmos, como mostra a Figura 14, e então incluídos um por um de acordo com a variável que está sendo procurada. Uma vez inseridos os *slots*, se o usuário falar algo em que o Watson Assistant não consiga identificar todas as variáveis dos *slots*, o *chatbot* vai perguntar variável por variável até ter todas as respostas (MAZON, 2018a). Este é o caso do exemplo dado por Mazon (2018a), mostrado na Figura 15,

Figura 13 – Exemplo de um nó de diálogo no Watson Assistant



Fonte: Mazon (2018a)

onde se usuário inserir somente 'eu quero uma pizza', o *chatbot* perguntará pelas outras variáveis de acordo com a pergunta definida no campo '*If not present, ask*' de cada *slot*, respectivamente. Por outro lado, neste mesmo exemplo, se o usuário inserir 'me vê uma pizza de queijo com massa fina para amanhã' a única pergunta exibida vai ser a do *slot* correspondente ao CEP, pois as outras variáveis já terão sido encontradas.

Figura 14 – Exemplo de como habilitar *slots* em um nó no Watson Assistant

Fonte: Mazon (2018a)

Um outro conceito importante para a construção de um *chatbot* no Watson Assistant é o de Digressão. Segundo Mazon (2018b), "Uma digressão ocorre quando o usuário está no

Figura 15 – Exemplo de *slots* para o nó #pedir_pizza

Pedir Pizza Customize X

If bot recognizes:
#pedir_pizza ⊖ ⊕

Then check for: Manage handlers

	Check for	Save it as	If not present, ask	Type		
1	@sabor	\$sabor	Qual sabor vc quer?	Required	⚙️	🗑️
2	@tipo_massa	\$tipo_massa	Qual o tipo de massa?	Required	⚙️	🗑️
3	@informacao_entrega	\$informacao_entrega	Informe o CEP de entrega	Required	⚙️	🗑️
4	@sys-date	\$date	Informe a data de entrega	Required	⚙️	🗑️

⊕ Add slot

Fonte: Mazon (2018a)

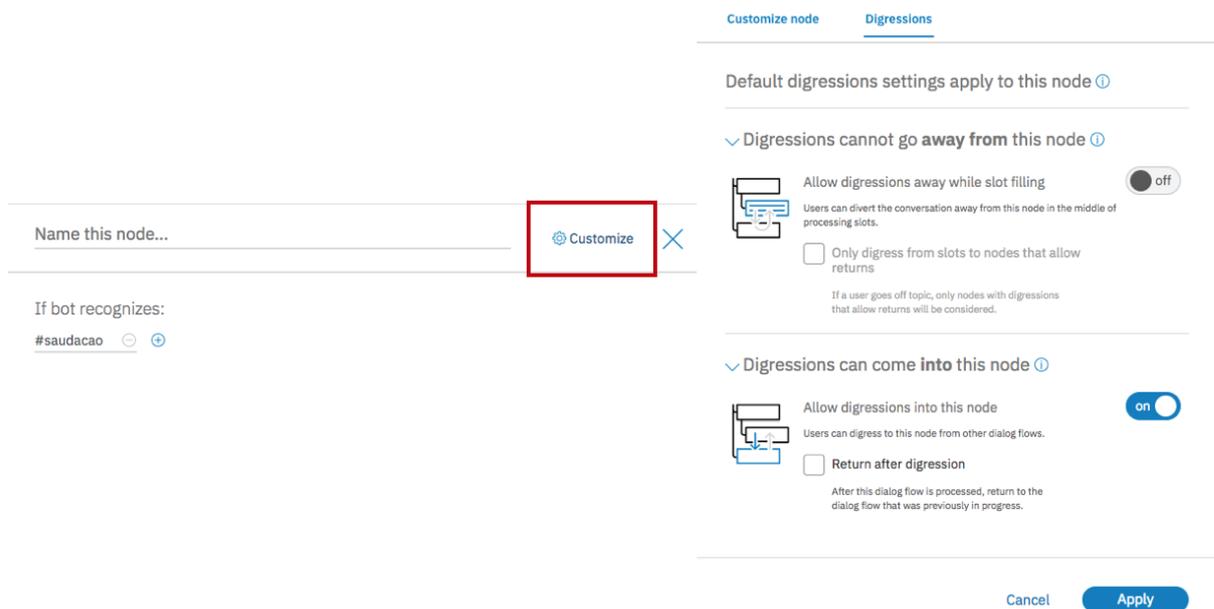
meio de um fluxo de nós de diálogo e muda abruptamente para outro conteúdo de outro fluxo". Logo, considerando novamente o exemplo do pedido de pizza citado anteriormente, se o usuário estiver no meio do nó de pedido de pizza, informando uma das variáveis de algum dos *slots* e de repente perguntar algo relativo à forma de pagamento, o *chatbot* responde de acordo com o nó relacionado à intenção #pagamento e depois volta pro nó que ele estava para terminar de verificar os *slots* que ainda faltam (MAZON, 2018b). A Figura 16 mostra como é possível permitir digressões em um nó.

Por fim, uma vez que estes conceitos básicos são estudados, é possível entender e desenvolver um *chatbot* através do IBM Watson Assistant.

2.5 Android

O Android é uma plataforma para dispositivos móveis que foi desenvolvido inicialmente pela Android Inc., empresa que foi comprada pela Google em 2005 e que, desde 2007, esta

Figura 16 – Exemplo de como habilitar digressão em um nó no Watson Assistant



Fonte: Mazon (2018b)

plataforma vem sendo mantida, aprimorada e desenvolvida pela *Open Handset Alliance*⁴, grupo formado por grandes empresas do setor de tecnologia, lideradas pela própria Google (DEITEL; DEITEL; DEITEL, 2015; LECHETA, 2013). Atualmente, esta plataforma está presente em vários tipos de dispositivos que vão desde smartphones, tablets, relógio inteligentes (*smartwatches*) a televisões, câmeras, satélites da NASA, dentre muitos outros (DEITEL; DEITEL; DEITEL, 2015).

Uma das principais características do Android é ele ser *open source*, ou seja, possui o código-fonte aberto e gratuito (DEITEL; DEITEL; DEITEL, 2015). Esta característica é uma grande vantagem pois, como pontuado por Lecheta (2013), "desenvolvedores de todos os lugares do mundo podem contribuir para o seu código-fonte, adicionando novas funcionalidades ou simplesmente corrigindo falhas". Além disso, esta vantagem se estende às fabricantes de celular, pois permite que as mesmas possam utilizar o sistema operacional do Android nos celulares que fabricam sem ter que pagar por isso, além de permitir, através da licença de software *Apache Software Foundation*, que o código-fonte possa ser alterado para criar novos produtos sem que as alterações precisem ser compartilhadas com ninguém (LECHETA, 2013).

Outra característica importante do Android é que seu sistema operacional é baseado em Linux. De acordo com Lecheta (2013), "o sistema operacional do Android foi baseado no *kernel 2.6* do Linux, e é responsável por gerenciar a memória, os processos, *threads* e a segurança

⁴ <https://www.openhandsetalliance.com/index.html>

dos arquivos e pastas, além de redes e *drivers*". Esta característica é vista como uma vantagem para os desenvolvedores de aplicações para Android, que não precisam tratar diretamente destes serviços gerenciados pelo *kernel*.

No que diz respeito à construção de aplicativos para Android, utiliza-se o *Software Development Kit* (SDK), que nada mais é do que um conjunto de ferramentas dentre as quais estão um emulador para simular dispositivos, bibliotecas, depurador, entre outras (DEITEL; DEITEL; DEITEL, 2015). Além disso, as linguagens de programação que podem ser utilizadas são Java ou Kotlin, por meio do Ambiente de Desenvolvimento Integrado (*Integrated Development Environment*) (IDE) de preferência do desenvolvedor, como por exemplo, o Eclipse, Android Studio, IntelliJ IDEA, entre outros. Por fim, uma vez desenvolvido, o aplicativo já pode então ser disponibilizado na Play Store⁵, site da Google por meio de onde é possível baixar aplicativos para o seu dispositivo Android, de acordo com os critérios necessários para instalação do aplicativo no mesmo.

Sendo assim, com esses conceitos e informações em mente, é possível desenvolver um aplicativo para Android da melhor forma possível.

2.5.1 *Android Studio*

O Android Studio é um Ambiente de Desenvolvimento Integrado (IDE) baseado no IDE *JetBrains IntelliJ IDEA*, para desenvolvimento de aplicações para Android (DEITEL; DEITEL; WALD, 2016). Este IDE foi anunciado em 2013 e, a partir de Novembro de 2016, com o lançamento da versão Android Studio 2.2, se tornou o IDE oficial para Android no lugar do Eclipse Android Developer Tools, que desde então não recebeu mais atualizações (EASON, 2017).

Segundo Caldas (2017), os principais diferenciais do Android Studio em relação aos outros IDEs é que ele organiza melhor o projeto, colocando cada parte essencial em pastas separadas e favorecendo as buscas por arquivos, e que ele permite a utilização do *Gradle* dentro do projeto, auxiliando no processo de compilação, o que resulta em uma melhora na configuração dentro do aplicativo.

Os principais recursos do Android Studio, ainda de acordo com Caldas (2017) são:

- **Gradle**: contribui para facilitar a definição de configurações de compilação, definição de bibliotecas externas que serão utilizadas no projeto, entre outros;

⁵ <https://play.google.com/store>

- **Layout:** o Android Studio possui um editor de layout, onde é possível criar layouts arrastando componentes ou alterando o código *Extensible Markup Language* (XML), e que tem uma janela de pré-visualização para uma melhor antecipação de como vai ficar o layout;
- **Renderização dos Layouts:** ao renderizar a tela, o Android Studio consegue identificar possíveis erros no layout e sugerir a melhor forma para corrigi-los;
- **Análise de código:** o Android Studio possui algumas ferramentas de boas práticas de código, como o 'Android Lint', que varre o código sugerindo soluções para possíveis erros encontrados de forma que o código fique mais limpo e organizado;
- **Integração com Repositórios de Versionamento:** o Android Studio tem integração com diversos sistemas de versionamento, como *Github*, *Mercurial*, entre outros;
- **Debug:** o Android Studio traz ferramentas de monitoramento de memória e de utilização de recursos de que se tornam de grande importância no desenvolvimento de aplicações;
- **Keymap:** o Android Studio possui uma grande variedade de atalhos de teclado que facilitam o desenvolvimento e a utilização do IDE;
- **Plugins:** permite a instalação de *plugins* para auxiliar no desenvolvimento.

2.5.2 SQLite

O SQLite é uma biblioteca de código aberto que implementa um banco de dados Linguagem de Consulta Estruturada (SQL). Ele é utilizado principalmente para dispositivos móveis devido à sua leveza e simplicidade, tanto de desenvolvimento quanto de armazenamento, já que o banco de dados inteiro é armazenado como um único arquivo no disco. Neste banco de dados, a biblioteca compacta faz com que, mesmo com todos os recursos ativados, o seu tamanho fique menor do que 600kB, fazendo com que este banco de dados seja utilizado em um smartphone, por exemplo, que não possui tanta memória disponível (SQLITE, 2018?).

O Android possui integração com o SQLite permitindo que as aplicações utilizem a biblioteca. É permitido a cada aplicação criar um ou mais banco de dados que serão salvos na pasta 'database' da aplicação que criou, respectivamente. A criação do banco de dados pode ser feita principalmente utilizando a API do Android, via código, ou externamente em um cliente do SQLite (LECHETA, 2013). Na API do Android as classes que gerenciam a persistência de dados no SQLite são as classes do pacote **android.database**. A classe *SQLiteOpenHelper*, por exemplo, é utilizada para criar e recuperar um objeto do tipo *SQLiteDatabase*, que é a

representação do banco de dados. Já a classe `Cursor` é utilizada para gerenciar os resultados de uma busca no banco de dados, que é feita com SQL (DEITEL; DEITEL; DEITEL, 2015).

2.5.3 *Conceitos de um aplicativo Android*

Quando se fala em desenvolver um aplicativo para Android o desenvolvedor se depara com um bom suporte no que diz respeito a material de estudo e ferramentas que o ajudem a desenvolver algo que satisfaça a proposta do aplicativo. Entretanto, antes de somente reproduzir códigos é necessário compreender conceitos que vão servir como base do desenvolvimento, como o funcionamento de um aplicativo no sistema, os principais componentes de um aplicativo e como garantir que todas as bibliotecas, recursos e permissões estejam de acordo com o que o aplicativo necessita.

O sistema Android implementa o que (ANDROID DEVELOPERS, 2018?) chama de ‘princípio do privilégio mínimo’, onde cada aplicativo só pode acessar os recursos e componentes necessários para ele rodar, e nada mais. Isso evita possíveis erros e complicações que poderiam ser causados caso o aplicativo tivesse acesso a qualquer parte do sistema deliberadamente. Isto acontece porque o sistema gera as permissões de acesso aos arquivos baseado no ID de usuário do Linux do aplicativo que gerou estes mesmos arquivos, fazendo com que só o aplicativo que tem aquele ID de usuário possa acessar aqueles arquivos. Apesar disso, o desenvolvedor pode fazer com que o aplicativo compartilhe o mesmo ID de usuário do Linux com outro *app* para que se compartilhe dados entre eles, além de que, uma vez que o usuário autorize, é possível compartilhar também serviços do sistema como câmera, Bluetooth, etc.(ANDROID DEVELOPERS, 2018?)

Outro conceito base diz respeito aos componentes dentro de um aplicativo Android. Para Android Developers (2018?), um componente é um bloco de construção que influenciam o comportamento geral de um aplicativo. Os principais componentes, de acordo com Android Developers (2018?), são:

- **Atividade (*Activity*):** representa uma tela do aplicativo. Cada atividade ajuda a compor uma experiência coesa para o usuário mas é independente de outras atividades do aplicativo. Num aplicativo de email, por exemplo, existe uma atividade que mostra uma lista de novos emails, outra atividade que escreve um novo e-mail, outra ainda que lê emails, etc, porém a atividade de escrever um novo email pode ser iniciada por algum outro aplicativo que queria compartilhar algo via email. Além disso, cada atividade em um aplicativo é implementada

como uma subclasse de *Activity*.

- **Serviço (*Service*):** é um componente, geralmente executado em segundo plano, para realizar operações de execução longa ou para realizar trabalho para processos remotos. Eles não apresentam uma interface do usuário. Por exemplo, um serviço pode tocar música em segundo plano enquanto o usuário está em um aplicativo diferente ou buscar dados na rede sem bloquear a interação do usuário com uma atividade. Uma vez iniciado, por uma atividade por exemplo, um serviço pode ser deixado executando ou outro componente pode vincular-se a ele para interagir. No que diz respeito à implementação, um serviço é uma subclasse de *Service*.
- **Provedores de conteúdo (*ContentProvider*):** gerenciam um conjunto de dados de um aplicativo que pode ser compartilhado com outros aplicativos. Independente do local onde são armazenados os dados, se em um banco de dados SQLite ou em algum outro local de armazenamento que o aplicativo pode acessar, se o provedor de conteúdo permitir, outros aplicativos podem consultar ou até modificar esses dados. Por exemplo, qualquer aplicativo com as permissões adequadas pode consultar um provedor de conteúdo para ler e gravar informações sobre uma pessoa específica. Em relação à implementação, um provedor de conteúdo é uma subclasse de *ContentProvider* e é necessário que seja implementado também um conjunto de APIs que permitem a outros aplicativos realizar transações com aquele *ContentProvider*.
- **Receptores de transmissão (*BroadcastReceiver*):** O sistema pode gerar transmissões, como uma transmissão que diz que a bateria está acabando, que houve uma captura de tela, entre outros, e os aplicativos também, como uma transmissão que comunica a outros dispositivos que estão disponíveis para usar dados que foram baixados no dispositivo. Assim, um receptor de transmissão é um componente que responde a essas transmissões gerando uma notificação que vai ser exibida ao usuário, por exemplo, de forma que o usuário tome conhecimento de que a transmissão ocorreu. A classe que implementa um receptor de transmissão é a *BroadcastReceiver* e cada transmissão resulta num objeto *Intent*.

Um objeto *Intent* define uma mensagem para ativação de um componente ou de um tipo de componente específico. Neste objeto é especificada a ação a ser executada e a URI dos dados usados para esta ação, no caso do objeto está ativando uma atividade ou um serviço, ou uma string representando a transmissão a ser recebida, no caso de o componente ativado

ser um receptor de transmissão. Além disso, é possível também retornar o resultado de uma atividade (um contato que foi solicitado, por exemplo, ou uma foto que foi tirada, etc) através do objeto *Intent*. É importante salientar que, por segurança, para ativar um provedor de conteúdo não se utiliza uma *intent* e sim um *ContentResolver*. Este objeto funciona como uma camada de abstração para que o componente que solicita as informações do provedor de conteúdo não precise lidar diretamente com ele. Sendo assim, a ativação de cada tipo de componente ocorre da seguinte maneira:

- para iniciar uma atividade um objeto *Intent* é passado ao *startActivity()* ou *startActivityForResult()* (no caso de ser necessário retornar um resultado).
- para iniciar um novo serviço ou dar novas instruções a um serviço que já está em andamento passando um objeto *Intent* a *startService()* ou vinculando este objeto a um serviço usando *bindService()*.
- no caso de iniciar uma transmissão, basta passar um objeto *Intent* para os métodos *sendBroadcast()*, *sendOrderedBroadcast()* ou *sendStickyBroadcast()*.
- para iniciar uma consulta a um provedor de conteúdo é necessário chamar *query()* em um *ContentResolver*.(ANDROID DEVELOPERS, 2018?)

Para que o sistema identifique que esses componentes existem e precisam ser ativados, eles têm que ter sido declarados no arquivo de manifesto, o *AndroidManifest.xml*. Este arquivo contém todas as informações referente aos componentes além de outras informações essenciais para a execução do aplicativo como permissões de usuário que o aplicativo vai precisar (como leitura de contatos, acesso a internet, etc), recursos de hardware e software que o aplicativo vai precisar (câmera, Bluetooth, etc), as bibliotecas que o aplicativo vai precisar mas não são da estrutura do Android e o nível de API mínimo que o aplicativo necessita para ser executado de acordo com as APIs que ele usa (ANDROID DEVELOPERS, 2018?).

Uma vez que o código do aplicativo está pronto para ser executado e sem erros, o Android SDK compila juntamente o código e todos os arquivos e recursos do aplicativo transformando tudo isso em um arquivo de sufixo *.apk*, que depois é usado para a instalação do aplicativo no dispositivo Android (ANDROID DEVELOPERS, 2018?).

2.6 Firebase

O Firebase é uma plataforma de ferramentas em nuvem para auxiliar no desenvolvimento de aplicações para dispositivos móveis Android ou iOS, e aplicações web. Esta plataforma

foi criada em 2011 pela Firebase, Inc. e foi adquirida em 2014 pela Google (TAMPLIN, 2014). Ela possui uma interface simples, além de ter suporte para desenvolvimento nas linguagens de programação C++, Java, Javascript, Node.js, Objective-C e Swift (BATTISTELLI, 2017?).

Os serviços oferecidos pelo Firebase estão divididos em duas categorias: serviços para auxiliar a desenvolver e testar uma aplicação e serviços para expandir e melhorar negócios, melhorando o engajamento dos mesmos com uma aplicação. Na primeira categoria estão enquadrados serviços como o de banco de dados, o de autenticação, o serviço de armazenamento de dados, o Cloud Functions, e vários outros. Já na segunda categoria se encaixam os serviços de *insights* sobre a aplicação, o serviço de notificações Cloud Messaging, um serviço de customização de conteúdo por cliente, o de exibição de propagandas, entre outros. É importante mencionar que nem todos os serviços do Firebase são gratuitos e ilimitados. Alguns deles seguem o sistema *'pay to scale'* onde a utilização do serviço é gratuita somente até certo número de requisições, passando depois a cobrar de acordo com cada serviço oferecido (FIREBASE, 2018?a). Os serviços do Firebase podem ser monitorados e gerenciados pelo Firebase Console ou via Firebase SDK.

Dentre toda esta variedade de serviços do Firebase, nesta pesquisa foram utilizados somente o Firebase Cloud Functions e o Firebase Cloud Messaging.

2.6.1 *Firebase Cloud Functions*

De acordo com Firebase (2018d?b), o Cloud Functions é um produto conjunto do Google Cloud Platform e do Firebase criado para atuar como solução de computação sem servidor no desenvolvimento de aplicativos. Através deste serviço, um código ou *'função'*, escrito em JavaScript ou TypeScript, é executado de forma automática como resposta a eventos acionados pelos recursos do Firebase ou por solicitações HTTPS. Dessa forma, a principal vantagem do uso do Cloud Functions é que as funções criadas ficam armazenadas na nuvem do Google, fazendo com que não seja necessário gerenciar servidores próprios (FIREBASE, 2018c). Além disso, por ser um serviço integrado ao Firebase, o Cloud Functions está disponível tanto pelo SDK do Firebase, caso seja acessado via código, quanto no Firebase Console (FIREBASE, 2018?c).

No que diz respeito às funções, uma vez que uma função é criada, os servidores do Google passam a detectar se ela foi disparada, gerenciando as instâncias de servidor virtual necessárias para executar a função, de acordo com a carga produzida pela execução da função.

Além disso, uma função é "executada isoladamente, no próprio ambiente e com a própria configuração", explica Firebase (2018?b).

Assim, em resumo, os principais recursos oferecidos pelo Cloud Functions, segundo Firebase (2018?b), são que:

- **O Cloud Functions integra a plataforma do Firebase**, o que permite que as funções sejam acionadas por outros serviços do Firebase como o *Realtime Database*, *Firebase Authentication*, entre outros.
- **O custo de manutenção para o desenvolvedor é zero**, pois ele não precisa se preocupar com credenciais, configuração e provisionamento de novos servidores ou desativação dos antigos.
- **Mantém a privacidade e segurança do código**, uma vez que o Cloud Functions é totalmente isolado do cliente, trazendo a certeza de que o código é privado e sempre fará exatamente o que o desenvolvedor quer.

2.6.2 *Cloud Messaging*

Um dos serviços mais comuns em um dispositivo móvel é o serviço de notificações. Logo, é importante ao desenvolvedor ter disponível ferramentas que possam ajudar no envio destas notificações. Assim, o Firebase Cloud Messaging é o serviço do Firebase que permite que mensagens de notificação referente à um aplicativo sejam enviadas para um dispositivo, estimulando desta forma, a interação do usuário com o aplicativo. Este serviço é totalmente grátis e pode ser adicionado a um aplicativo Android, iOS ou Web, além de poder mandar mensagens instantâneas de até 4KB (FIREBASE, 2018?d).

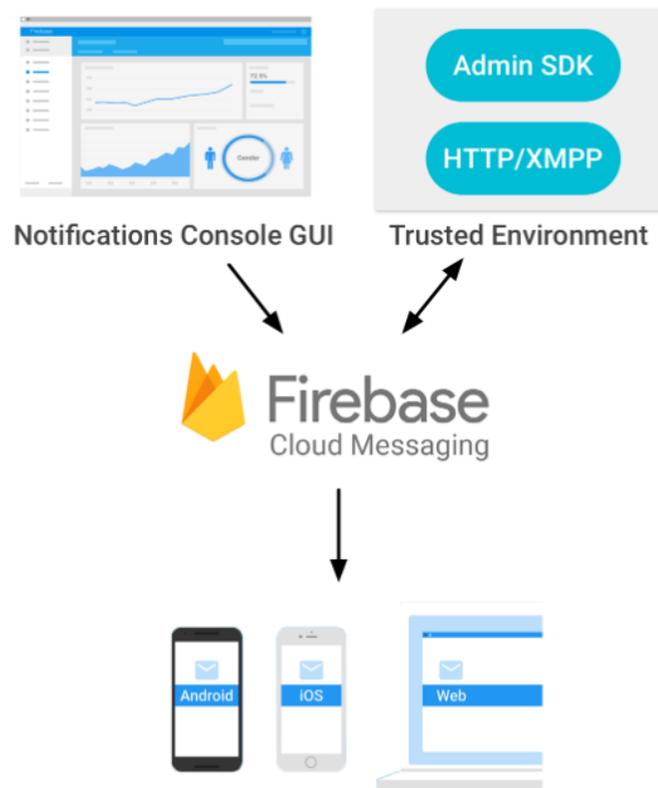
Dentre os principais recursos que este serviço oferece, Firebase (2018?d) destaca: o fato do Firebase Cloud Messaging poder enviar mensagens tanto de notificação quanto de dados; a segmentação versátil de mensagens, onde uma mensagem pode ser enviada para dispositivos únicos, para grupos de dispositivos ou para dispositivos inscritos em tópicos: e a possibilidade de enviar mensagens de *apps* cliente de volta para o servidor de forma confiável e com o mínimo de consumo de bateria possível.

No que diz respeito aos tipos de mensagens enviadas pelo Firebase Cloud Messaging, é possível que se envie dois tipos de mensagens: de notificação, que também são conhecidas como "mensagens de exibição", e mensagens de dados, que ao serem recebidas, vão ter algum tipo de tratamento pelo *app* cliente. Sobre isso, é possível afirmar que:

As mensagens de notificação contêm um conjunto predefinido de chaves visíveis pelo usuário. As mensagens de dados, por outro lado, contêm somente pares de chave-valor personalizados definidos pelo usuário. As mensagens de notificação podem conter um *payload* de dados opcional. O *payload* máximo para ambos os tipos de mensagem é de 4 KB, exceto ao enviar mensagens do Console do Firebase, que impõe um limite de 1.024 caracteres (FIREBASE, 2018?e)

Um exemplo geral de como funciona a integração do Firebase Cloud Messaging a uma aplicação é demonstrado na Figura 17, onde em uma ponta tem o Firebase Console ou um ambiente confiável HTTP/XMPP, e na outra ponta, recebendo as mensagens, um *app* cliente iOS, Android ou Web (JavaScript).

Figura 17 – Esquema geral de integração do Firebase Cloud Messaging



Fonte: Firebase (2018?d)

3 METODOLOGIA DE DESENVOLVIMENTO

3.1 Tecnologias Utilizadas

Para cada etapa de desenvolvimento desta pesquisa foram utilizados softwares diferentes mas que juntos, geraram a aplicação final. Na etapa de desenvolvimento do protótipo de *chatbot* foi utilizado o *framework* IBM Watson, mais especificamente, o serviço Watson Assistant. Para utilização do serviço Watson Assistant foi necessário criar uma conta no IBM Cloud. Uma vez que foi criada a conta, selecionou-se o serviço Assistant e logo após foi criado um *workspace* em Português do Brasil. Já na etapa de desenvolvimento de uma aplicação de teste para Android, foi utilizado o Android Studio como ambiente de desenvolvimento. O hardware utilizado para o desenvolvimento em ambas as etapas foi um *ultrabook* DELL, com processador core Intel i7 de quarta geração, 16 GB de memória RAM, 1TB de HD e sistema operacional Windows 10.

3.2 Métodos no Desenvolvimento do Protótipo de *Chatbot*

3.2.1 Fluxo de diálogo proposto

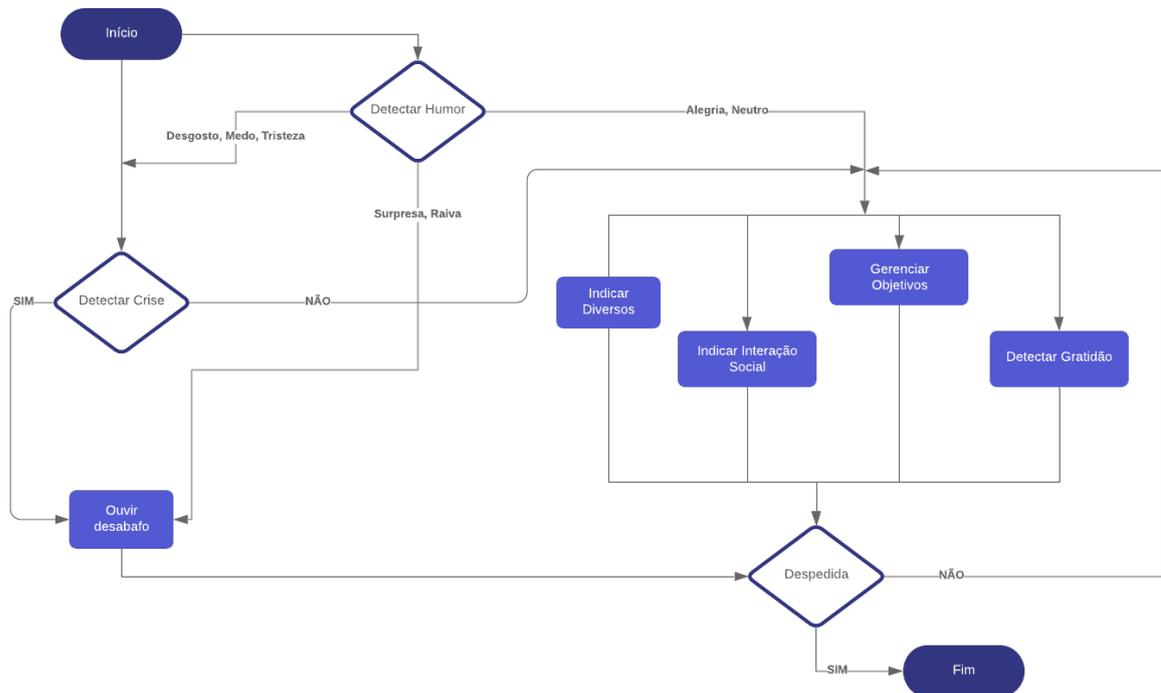
Quando se trata do planejamento do fluxo da conversa para um *chatbot*, muita coisa está envolvida e não só a abordagem computacional em si. “Quais são as pessoas da conversa?”, “Qual o problema que se busca resolver?”, “Quais cenários de conversa podem ocorrer?” e “Como o *chatbot* reage à determinada situação?” são perguntas básicas que devem ser respondidas antes mesmo de algum código-fonte ser escrito.

Assim, depois de já terem sido respondidas as duas primeiras perguntas, como demonstrado nos capítulos anteriores, foi feita uma análise sobre os possíveis cenários da conversa de um *chatbot* com uma pessoa com sintomas da depressão, e chegou-se ao diagrama exibido na Figura 18.

Ao iniciar uma conversa o *chatbot* busca detectar humor do usuário de forma a decidir que ação tomar baseada no que foi detectado. Para esse propósito, foi considerado humor bom = [alegria, neutro], humor ruim = [desgosto, medo, tristeza], e as emoções [surpresa, raiva] foram tratadas separadas das demais.

Quando o *chatbot* detecta ‘humor bom’, passa-se à uma das metodologias escolhidas para incentivar um possível aumento do interesse do usuário em atividades que favoreçam

Figura 18 – Diagrama do fluxo de conversa do protótipo de *chatbot*



Fonte: Autoria Própria.

uma melhoria na qualidade de vida: “Indicação de diversos”(indicação de músicas ou fatos e informações sobre combate à depressão), “Gerenciar objetivos”, “Indicar interação social” e “Detectar gratidão”.

Quando em “Indicação de diversos”(indicação de músicas ou fatos e informações sobre combate à depressão), o usuário recebe do *chatbot* dicas de músicas e informações relevantes no combate à depressão e na diminuição do estresse e da ansiedade.

Já em “Gerenciar objetivos”, o *chatbot* ajuda o usuário a pensar em objetivos pra semana ou a repensar objetivos já estabelecidos mas que o usuário sente dificuldade em cumprir.

“Indicar interação social” por si só já descreve seu objetivo. O *chatbot* dá dicas de atividades que podem ser feitas baseado no clima da cidade do usuário. Aqui, o *chatbot* também incentiva o usuário a falar sobre alguma atividade que ele tinha prazer em fazer e agora não faz mais. Uma vez que o *chatbot* “toma conhecimento” desta atividade, ele pode indicá-la pro usuário em interações futuras, fazendo com que o usuário seja incentivado a fazer atividades personalizadas.

Em “Detectar gratidão”, o usuário é encorajado pelo *chatbot* a pensar em 3 coisas boas que possam ter acontecido com ele. Uma vez que o usuário vai digitando e recebendo a confirmação de que suas opções estão sendo “ouvidas”, o *chatbot* pede para que ele escolha

entre uma das 3 coisas boas, fazendo com que o usuário reflita e reconheça o que houve de bom no seu dia.

Quando o *chatbot* detecta “humor ruim”, acredita-se que o usuário está em um momento difícil e ele é redirecionado para a “Detecção de Crise” logo depois, “Ouvir desabafo”.

Uma vez em “Detecção de Crise”, o *chatbot* pergunta ao usuário se ele está em um momento crítico e se a resposta for positiva, exibe uma mensagem com informações e inscrições de contato com o Centro de Valorização da Vida e outros telefones de emergência. Logo após, o *chatbot* oferece a opção de ligar para o contato de emergência cadastrado ou, se o usuário não possui contato, dá a opção de cadastro. Essa rotina se repete não somente quando é detectado humor ruim, como também quando é detectado palavras gatilho como “suicídio/suicidar, sos, matar/morrer”.

Na etapa de “Ouvir desabafo”, o usuário é incentivado pelo *chatbot* a digitar seu “desabafo” de forma que quando ela termine, se sinta mais leve. Através dessa interação o *chatbot* também incentiva o usuário a procurar alguém de confiança, como um amigo, familiar ou terapeuta, para conversar. Esta interação também é iniciada quando o *chatbot* detecta as emoções “raiva” e “surpresa”.

Por fim, na “Despedida” o usuário é encorajado a escolher livremente entre as funções anteriormente descritas caso ele queira continuar conversando. Uma vez que ele negue, o *chatbot* se despede e encerra a interação.

Após ter sido projetado, o fluxo de diálogo desta pesquisa foi analisado por uma psicóloga clínica que não encontrou nenhuma inconsistência no mesmo.

3.2.2 *Intents e entities do projeto*

O mapeamento das *intents* básicas a serem detectadas no diálogo foi feito de acordo com a Figura 19.

As *intents* com prefixo #Bot_Control e #General foram incorporadas do catálogo de *intents* do Watson Assistant. Já as que não possuem prefixo foram criadas e alguns exemplos de treino adicionados foram:

- **#crise** - ‘Estou tão solitário que queria morrer’, ‘Eu queria poder dormir e nunca mais acordar’, etc;
- **#mood** - ‘não tem nada de bom acontecendo’, ‘tudo de boas’, ‘as coisas têm sido difíceis...’, etc;

Figura 19 – *Intents* do protótipo de *chatbot*

<input type="checkbox"/> Intent (14) ▼	Description (optional)
<input type="checkbox"/> #Bot_Control_Change_Subject	Mude para um tópico diferente.
<input type="checkbox"/> #Bot_Control_Clarification	Repita ou esclareça a última declaração.
<input type="checkbox"/> #Bot_Control_Confirm_Presence	Peça ao robô para indicar que está disponível.
<input type="checkbox"/> #Bot_Control_Reject_Response	Indique que a resposta do robô não aborda uma questão.
<input type="checkbox"/> #Bot_Control_Unsure	Indique não haver uma resposta definitiva a uma pergunta.
<input type="checkbox"/> #crise	Bot detecta sinal de crise/pensamento suicida
<input type="checkbox"/> #General_About_You	Solicite atributos pessoais genéricos.
<input type="checkbox"/> #General_Ending	Termine a conversa.
<input type="checkbox"/> #General_Greetings	Cumprimente o robô.
<input type="checkbox"/> #General_Negative_Feedback	Expresse feedback desfavorável.
<input type="checkbox"/> #General_Positive_Feedback	Exprima sentimento positivo ou gratidão.
<input type="checkbox"/> #gratidao	Coisas boas que podem ter acontecido
<input type="checkbox"/> #mood	Indique humor
<input type="checkbox"/> #saudacao	Responde mensagem de saudação do bot

Fonte: Autoria Própria.

- **#gratidao** - ‘minha mãe me abraçou’, ‘fui ao supermercado e comprei muitas comidas gostosas’, etc;
- **#saudacao** - ‘Boa noite’, ‘hey’, ‘e aí’, etc.

Já as *entities* criadas são exibidas na Figura 20. Cada entidade possui a opção *Fuzzy*

Figura 20 – *Entities* do protótipo de *chatbot*

My entities System entities	
<input type="checkbox"/> Entity (4) ▼	Values
<input type="checkbox"/> @contatoEmergencial	telefone
<input type="checkbox"/> @crise_trigger	suicídio, socorro, sos
<input type="checkbox"/> @emocao	tristeza, alegria, desgosto, medo, neutro, raiva, surpresa
<input type="checkbox"/> @resposta	negativa, positiva

Fonte: Autoria Própria.

Matching ativada, recurso do Watson Assistant que faz com que os valores das entidades possam ser reconhecidos mesmo que sejam mal digitados. Essa função ajuda também com a variação de gênero e número de algumas palavras.

3.2.3 Nós principais do diálogo

Os nós raiz do *chatbot* foram criados de acordo com o que é mostrado na Figura 21 e seguem a mesma estrutura do fluxo de diálogo proposto descrito anteriormente.

Figura 21 – Nós do diálogo



Fonte: Autoria Própria.

3.3 Métodos no Desenvolvimento da Aplicação de Teste

3.3.1 Formação das bases de dados necessárias

Para fornecer dados base para as funções de “Diversos” e “Indicar Interação Social”, foi necessário a criação de um *dataset* próprio que atendesse a cada uma.

Para a função de “Diversos” o *dataset* criado é dividido entre músicas e informações. As músicas estão de acordo com um estudo do instituto de neurociência britânico *Mindlab International* (CURTIN, 2017) que divulga um ranking com 10 melhores músicas para redução de ansiedade, onde a primeira colocada diminui a ansiedade do ouvinte em até 65%. Quanto às informações, foi pesquisado 4 atitudes básicas que em geral ajudam a combater sintomas da depressão com embasamento de pelo menos um artigo científico relevante. Cada entrada consiste em:

- TEXTO, INFO, CONTRIBUICAO, TIPO, LINK, EXIBICAO

É necessário esclarecer que para entradas do tipo música, o campo TEXTO é o nome do artista, INFO é a informação sobre o artista e LINK armazena o link de um vídeo daquela música no youtube. Para entradas do tipo informação, o campo TEXTO é uma versão resumida da informação, INFO é a informação literal e LINK armazena uma referência a um artigo científico relacionado à informação dada. As entradas do *dataset* de “Diversos” ficaram no mesmo formato exemplificado na Figura 22.

Figura 22 – Exemplo de entrada do *dataset* “Diversos”

```
"MUSICA",
"Marconi Union",
"Marconi Union é um trio da Inglaterra composto por Richard Talbot, Jamie
Crossley and Duncan Meadows que foi formado em meados de 2002/2003",
"https://www.youtube.com/watch?v=UfcAVEjslrU",
"Em 16 de outubro de 2011, o trio criou esta música
em colaboração com a British Academy of Sound Therapy(Academia Britânica de Terapia de Som).
Na música eles usaram violão, piano e outros sons que juntos
podem induzir um estado de calma.", 0
```

Fonte: Autoria Própria.

Já para a função de “Indicar Interação Social” o *dataset* base criado foi feito de acordo com 12 interações sociais populares entre pessoas adultas. Cada entrada, como exemplificado na Figura 23, possui a seguinte estrutura:

- TEXTO, CLIMA, EXIBICAO

Figura 23 – Exemplo de entrada do *dataset* “Interações sociais”

```
"começar a frequentar aulas de dança de salão",
"FRIO",
0
```

Fonte: Autoria Própria.

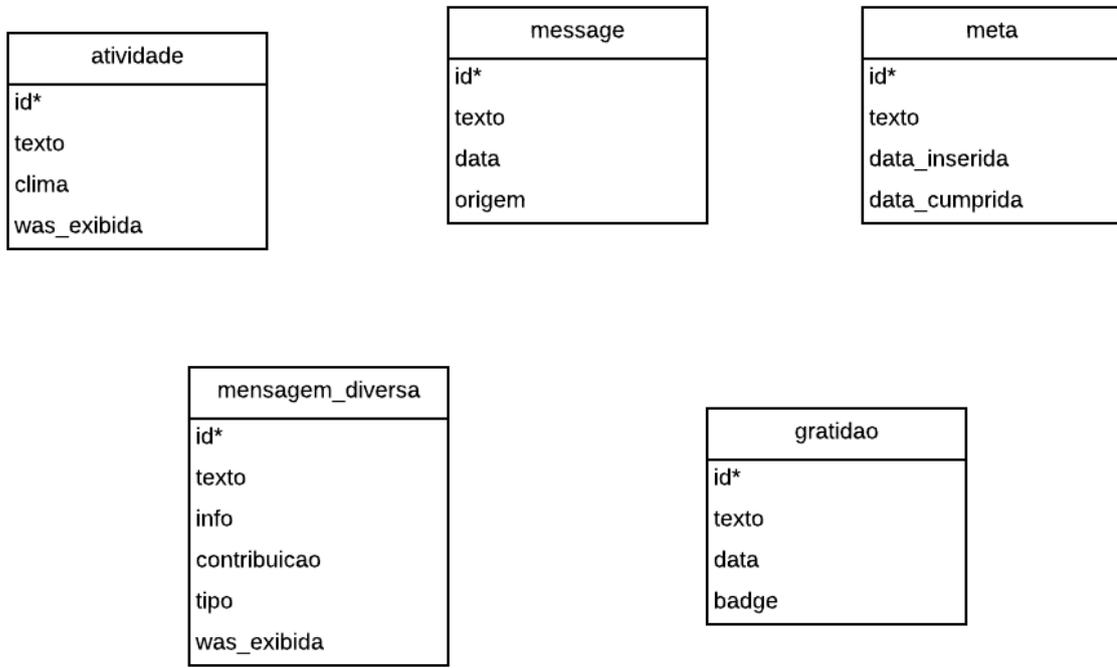
Para os dados de ambas as funções, o campo EXIBICAO armazena *true* (1) ou *false* (0) caso o dado já tenha sido exibido ou não. Isto foi implementado para evitar de repetir dados durante a conversa com o usuário, tornando mais dinâmica a interação do mesmo com a aplicação.

3.3.2 Modelagem do banco de dados

Conforme o diagrama da Figura 24, uma base de dados foi desenvolvida para salvar os dados da aplicação de forma que a aplicação não precise de nenhum serviço externo para

manipulação de dados. Apesar de não se relacionarem, as tabelas guardam dados essenciais para as interações do *chatbot*.

Figura 24 – Diagrama de entidade relacionamento do banco de dados da aplicação



Fonte: Autoria Própria.

A tabela de 'atividade' guarda as indicações de atividades, enquanto que a de 'gratidao', como o nome sugere, guarda as informações que são exibidas na tela de "Galeria". A tabela 'meta' guarda os objetivos cadastrados, a tabela 'mensagem_diversa' guardas as informações que são exibidas em "Diversos" e a tabela 'message' guarda todas as mensagens enviadas e recebidas pela aplicação.

Para cada uma dessas tabelas, foram criadas seus respectivos modelos e classes de acesso a dados.

3.3.3 Incorporação do chatbot na aplicação

Inicialmente, para a utilização do serviço, basta informar as credenciais do *workspace* a ser utilizado, como mostra o Código-fonte 1:

```

1 myConversationService = new ConversationService (
2     "2018-04-26", appCtx.getString(R.string.username),
3     appCtx.getString(R.string.password)
  
```

```
4 );
```

Código-fonte 1 – Credenciais do serviço do Watson Assistant no Android

Para enviar a mensagem, é preciso informar o texto que o usuário digitou e o contexto da conversa. Assim, a primeira coisa feita na aplicação é recuperar as variáveis importantes pro contexto (dados do contato emergencial e fuso horário) do arquivo XML no qual elas foram salvas, e passar esse contexto juntamente com a mensagem que o usuário digitou como parâmetro no objeto *MessageRequest*, como visto no Código-fonte 2.

```
1 MessageRequest request = new MessageRequest.Builder()
2     .inputText(inputText).context(ctxConversation).build();
```

Código-fonte 2 – Objeto *MessageRequest*

Em geral o jeito mais comum de fazer chamadas à API do Watson Assistant é através do método *execute()*. Entretanto, como as mensagens são mostradas no componente *RecyclerView* do Android, é necessário fazer a chamada à API em uma *thread* diferente da *thread* principal da aplicação. Assim, a chamada à API é feita utilizando o método assíncrono *enqueue()* e recebe no método *onResponse* um objeto *MessageResponse* que contém os dados de resposta do serviço de conversação, dos quais os principais são as mensagens de resposta do *chatbot* e o contexto atualizado. É neste método que as mensagens e variáveis de contexto são tratadas e depois, assincronamente, o componente *RecyclerView* é atualizado.

```
1 myConversationService.message(getString(R.string.workspace),
2     request)
3
4     .enqueue(new ServiceCallback<MessageResponse>() {
5
6         //recebe a resposta do watson
7         @Override
8         public void onResponse(final MessageResponse response){
9             //Tratamento da resposta
10        }
11    }
12 );
```

Código-fonte 3 – Chamada à API do Watson Assistant

3.3.4 Tratamento das mensagens de “Indicação de interação social”

Quando a mensagem é recebida, a aplicação procura nas variáveis de contexto requisição para um dos 4 tipos de ações seguintes:

- Indicações
- Diversos
- Objetivos
- Galeria

Para a maioria dessas ações, apenas operações básicas de persistência como incluir, ou atualizar, etc são realizadas. Entretanto, quando a requisição é para a ação “Indicações”, se aliada a ela estiver a variável ‘show’ (exibir), a aplicação faz chamada à API *OpenWeatherMap*¹ para consultar o clima baseado na informação de localização obtida pelo GPS do dispositivo. Esta função é realizada pelo método *accessLocation()* mostrado no Código-fonte 4.

```

1 private void accessLocation () {
2     GPSManager mGPS = new GPSManager(MainActivity.this);
3     mGPS.getLocation ();
4     // caso nao tenha conseguido recuperar a localiza o do dispositivo
5     if ((mGPS.getLatitude ()==0)&&(mGPS.getLongitude ()==0)) {
6         ctxConversation.put ("responseClima", false);
7         ctxConversation.put ("indicacaoClimaFrio", atividadeRepository.findLatest ("FRIO").
8             getTexto ());
9         ctxConversation.put ("indicacaoClimaQuente", atividadeRepository.findLatest ("QUENTE").
10            getTexto ());
11     } else {
12         Weather.placeIdTask asyncTask = new Weather.placeIdTask (new Weather.AsyncResponse () {
13             public void processFinish (String weather_city, String weather_description,
14                 String weather_temperature, String weather_humidity,
15                 String weather_pressure,
16                 String weather_updatedOn) {
17
18                 ctxConversation.put ("responseClima", true);
19                 ctxConversation.put ("temperaturaClima", weather_temperature);
20                 ctxConversation.put ("descricaoClima", weather_description.toLowerCase ());
21                 // recupera atividade que ainda nao foi exibida
22                 ctxConversation.put ("atividadeClima", getAtividadeClima (weather_temperature));

```

¹ <https://openweathermap.org/>

```

20
21     }
22     });
23     asyncTask.execute( String.valueOf(mGPS.getLatitude()),String.valueOf(mGPS.getLongitude
24         ());
25 }

```

Código-fonte 4 – Método *accessLocation()*

Foi criada a classe *GPSManeger* para capturar a localização do GPS do dispositivo. Depois, se uma latitude e uma longitude válida foram retornadas, a classe *Weather*, que instancia os serviços e funções da API *OpenWeatherMap*, foi chamada para performar a detecção do clima. Uma vez que concluída a detecção, uma atividade relacionada com clima recuperado é trazida do banco de dados e incluída no contexto para ser exibida ao usuário de acordo com o *chatbot*.

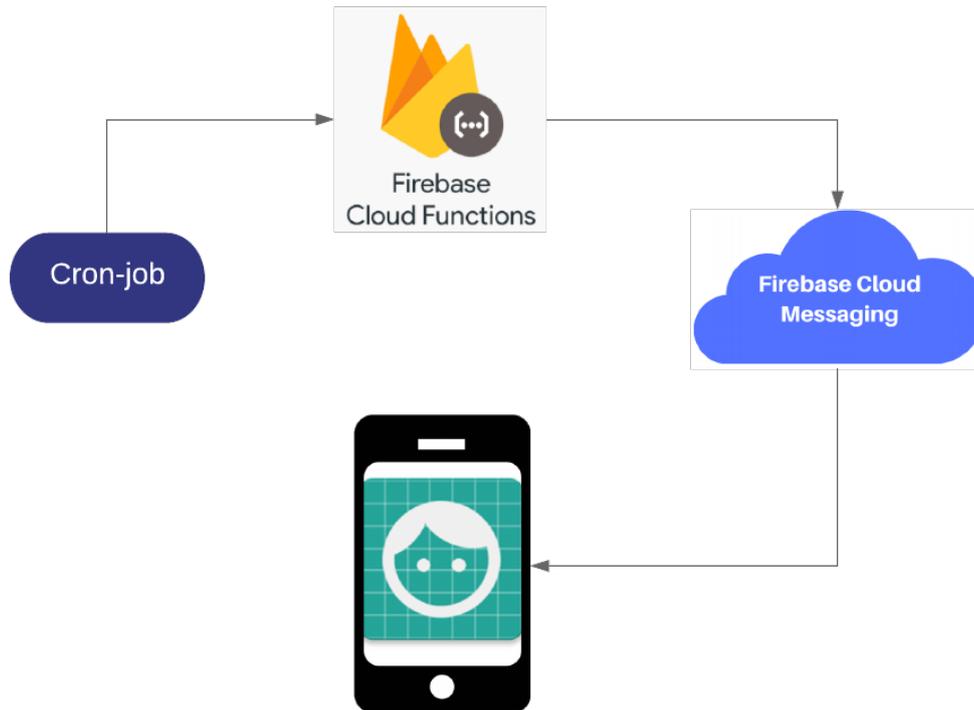
3.3.5 Serviço de notificação

Com a finalidade de estimular a interação do usuário com a aplicação, foi criado um serviço que notifica o usuário a fim de que ele converse com o *chatbot* pelo menos uma vez por dia. As notificações são enviadas e recebidas pelos dispositivos de acordo com a arquitetura exibida na Figura 25.

Foram criados serviços de agendamento de tarefas, popularmente conhecidos como *cron-jobs*. Esse tipo de serviço permite a realização de alguma requisição HTTP em dias e horários determinados pelo criador do *cron-job*. Assim, foram criados 3 *cron-jobs* diferentes onde cada um executa uma função do Firebase Functions que, por sua vez, envia uma mensagem padrão para um canal(*topic*) diferente do Firebase Cloud Messaging. Esses *cron-jobs* são ativados diariamente em 3 horários diferentes, de forma que as interações do usuário com a aplicação não ocorram exatamente ao mesmo tempo. O serviço de agendamento de tarefas utilizado foi o *cron-job.org* que é grátis e simples de ser utilizado.

Quando o usuário recebe a notificação, e clica nela, a aplicação é aberta e, ao ser verificado que a chamada à aplicação foi feita por uma notificação, uma chamada ao método *enviarMensagem()* é realizada e a mensagem exibida para o usuário é uma das mensagens do nó “Saudação” do serviço de conversação. Esse processo é realizado no método *onCreate()* da aplicação de forma que o resultado da notificação só aconteça quando a aplicação não esteja em

Figura 25 – Arquitetura do envio de notificação



Fonte: Autoria Própria.

execução.

3.3.6 Design da interface

Conquistar o interesse do usuário na interação com o chatbot, sendo que o mesmo tem predisposição falta de interesse devido ao transtorno, é um desafio. Além disso, o estigma associado à transtornos mentais e como ele afeta as pessoas que sofrem desses transtornos, fazendo com que elas sintam vergonha de demonstrar que precisam buscar ajuda (LUCAS *et al.*, 2017), também é algo que deve ser levado em conta. Logo, um dos objetivos para a interface da aplicação foi lembrar o máximo possível os aplicativos de troca de mensagem mais populares de forma a não chamar atenção, gerando um possível constrangimento ao usuário.

Sendo assim, um protótipo das principais telas do sistema foi desenvolvido e ao serem implementados, buscou-se atender ao máximo as diretrizes de design para aplicativos para Android do Google Material Design e os fatores citados anteriormente. É possível observar através da Figura 26 o protótipo das telas. Além disso, de forma a tornar a aplicação um pouco mais amigável, foi atribuído o nome francês *Jean* ao *chatbot*, de forma que a aplicação passou a ser identificada como *JeanBot*.

Após concebido o protótipo, passou-se à implementação do layout e o resultado final

Figura 26 – Protótipo das telas do aplicativo



(a) Tela principal

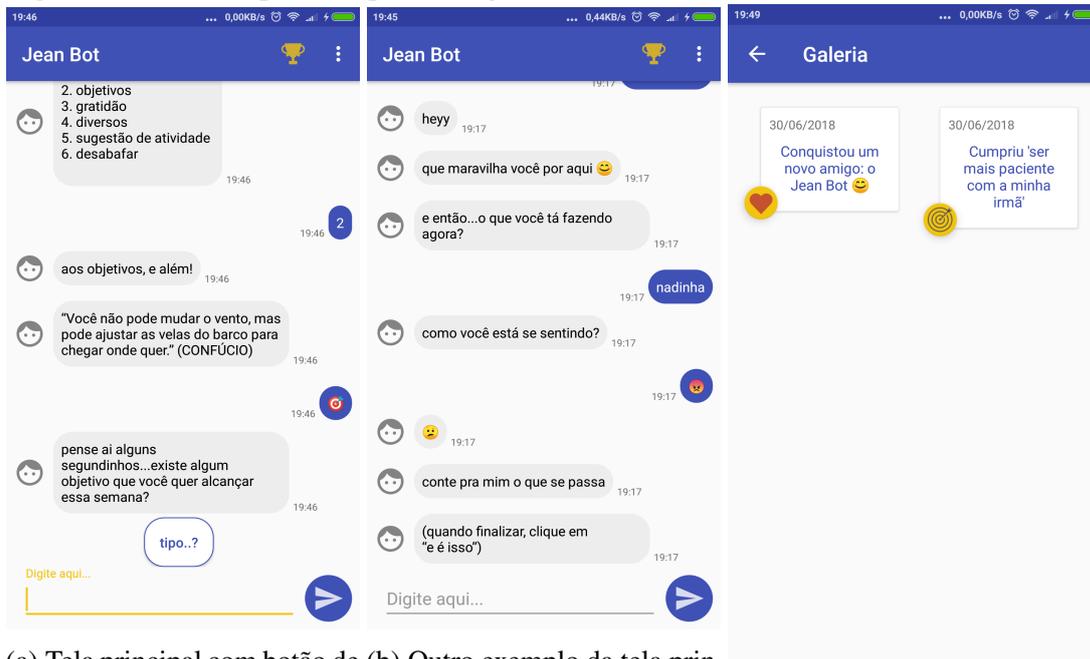
(b) Menu da tela principal

(c) Galeria

Fonte: Autoria própria.

está demonstrado na Figura 27.

Figura 27 – Telas após a implementação



(a) Tela principal com botão de sugestão de resposta

(b) Outro exemplo da tela principal

(c) Galeria

Fonte: Autoria própria.

4 EXPERIMENTOS

4.1 Visão Geral

Após divulgação em uma rede social, aplicação foi testada por 10 pessoas que apresentavam alguns dos sintomas de Transtorno Depressivo Maior descritos no DSM-V (AMERICAN PSYCHIATRIC ASSOCIATION, 2014) dentre as quais algumas já tinham histórico de terem sido diagnosticadas com a doença. Elas interagiram com a aplicação durante uma semana utilizando dispositivos com sistema operacional Android igual ou superior ao Android 5.0 (*API level 21*). Antes de realizar o teste, elas assinaram um Termo de Consentimento Livre e esclarecido que se encontra no Apêndice A deste trabalho. Além disso, foram informadas sobre o projeto e lembradas de que a aplicação não substitui terapia convencional. Logo, o teste da aplicação deveria ser concomitante à terapia, caso já fizessem. Tendo finalizado o período de testes, aplicou-se o questionário exposto no Apêndice B.

4.2 Resultados e Discussão

O questionário foi dividido em três partes: uma avaliação básica de redução de sintomas baseada no PHQ- 9 (KROENKE; SPITZER; WILLIAMS, 2001); uma avaliação sobre o *chatbot* e suas funções cognitivas; uma avaliação sobre aspectos visuais da aplicação de teste. Por meio dele, buscou-se avaliar os seguintes aspectos da aplicação do teste:

- Influência do *chatbot* na diminuição de sintomas
- Funções cognitivas e relevância do *bot* para o participante
- Utilidade das informações
- Performance
- Design da interface

4.2.1 *Influência do chatbot na diminuição de sintomas*

Para avaliar a influência do *chatbot* na diminuição dos sintomas foi inserido no questionário uma avaliação básica baseada no PHQ- 9 (KROENKE; SPITZER; WILLIAMS, 2001), onde o usuário respondeu sobre os sintomas e a frequência em que se apresentavam antes e durante a interação com o aplicativo nas semanas de testes. Nas respostas é possível observar que, durante a interação com a aplicação, houve uma leve redução na observação de alguns

sintomas em algumas pessoas de acordo com a tabela 1:

Tabela 1 – Tabela que representa a quantidade de usuários onde houve redução, por sintoma

Sintoma	Quantidade de usuários onde houve redução/Total
Tive pouco interesse ou prazer em fazer as coisas	4/10
Senti desânimo, desalento ou falta de esperança	5/10
Tive dificuldade em adormecer ou em dormir sem interrupções, ou dormi demais	6/10
Senti cansaço ou falta de energia	3/10
Tive falta ou excesso de apetite	2/10
Senti que não gosto de mim própria(o) - ou que sou um falho(a) ou me desiludi a mim próprio(a) ou à minha família	4/10
Tive dificuldade em concentrar-me nas coisas, como ao ler o jornal ou ver televisão	1/10
Movimentei-me ou falei tão lentamente que outras pessoas poderão ter notado. Ou o oposto: estive agitado(a) a ponto de andar de um lado para o outro muito mais do que é habitual	4/10
Pensei que seria melhor está morto(a), ou magoar-me a mim próprio(a) de alguma forma	3/10

Fonte: Autoria própria.

Assim, é possível comprovar que a utilização do protótipo de *chatbot* proposto nesta pesquisa auxilia na redução de sintomas da depressão.

4.2.2 Funções cognitivas e relevância do bot para o participante

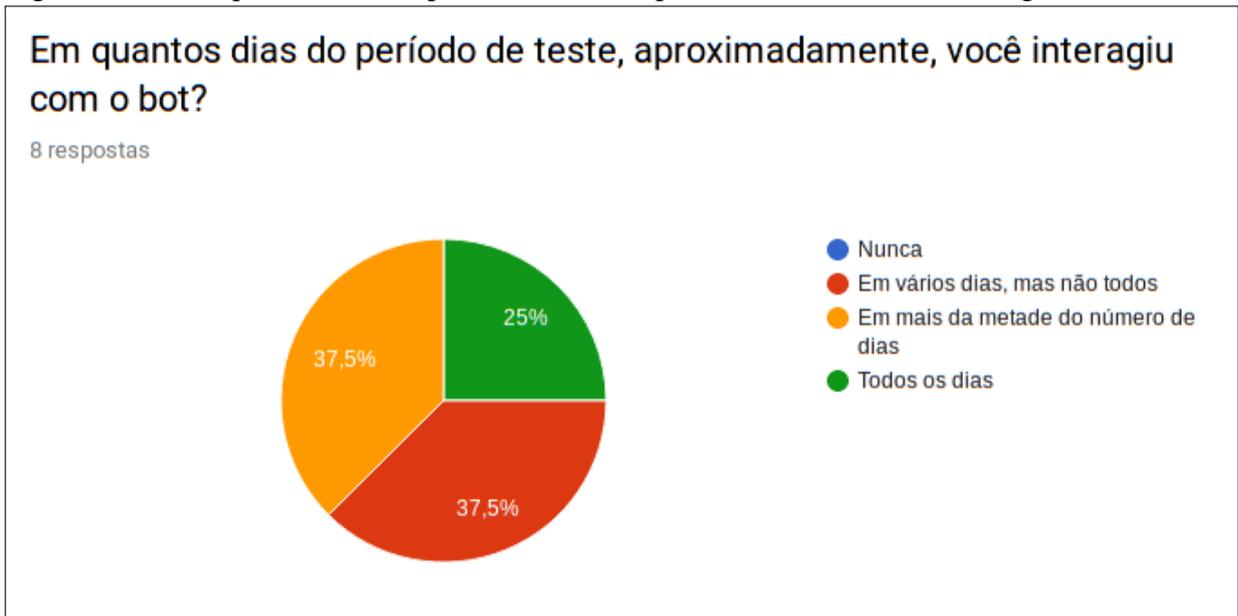
A primeira pergunta que contribuiu para avaliar estes aspectos foi "Em quantos dias do período de teste, aproximadamente, você interagiu com o *bot*?". Com o resultado é possível perceber, de acordo o gráfico na Figura 28, que apesar de duas pessoas terem se absterido de responder essa pergunta, a maioria dos que a responderam interagiram com a aplicação em mais da metade dos dias do período de teste.

Outra pergunta foi "Como você avaliaria as estratégias de acompanhamento do *Bot*?". É possível observar através do gráfico da Figura 29 que apenas 1 pessoa, dentre as 10 que participaram dos testes, considerou as estratégias de acompanhamento do *bot* abaixo de Regular.

A terceira pergunta direcionada a este aspecto foi "Quanto você considera que o *Bot* te ajudou no combate aos sintomas que você apresenta?". Através do gráfico mostrado na Figura 30 é possível ver que apenas 1 pessoa, de 10, considerou que, numa escala de 1 (Ajudou em Nada) a 5 (Ajudou Muito), o bot não havia ajudado em nada (nota 1) no combate aos sintomas que a pessoa apresentava.

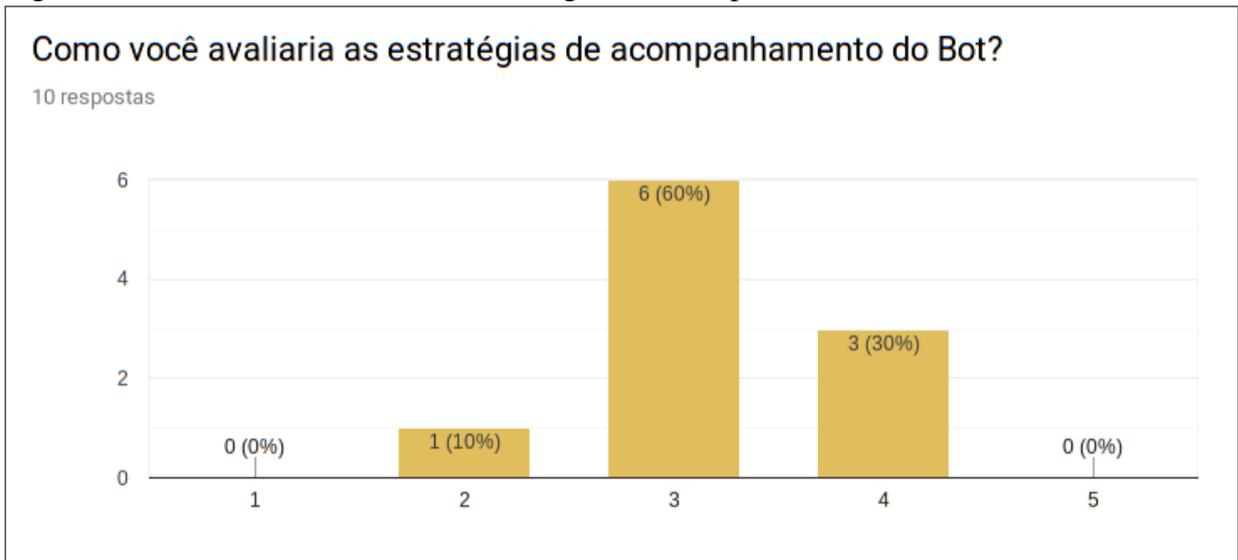
A última pergunta que avaliou as funções cognitivas e relevância do *bot* para o

Figura 28 – "Em quantos dias do período de teste, aproximadamente, você interagiu com o bot?"



Fonte: Autoria própria.

Figura 29 – "Como você avaliaria as estratégias de acompanhamento do Bot?"

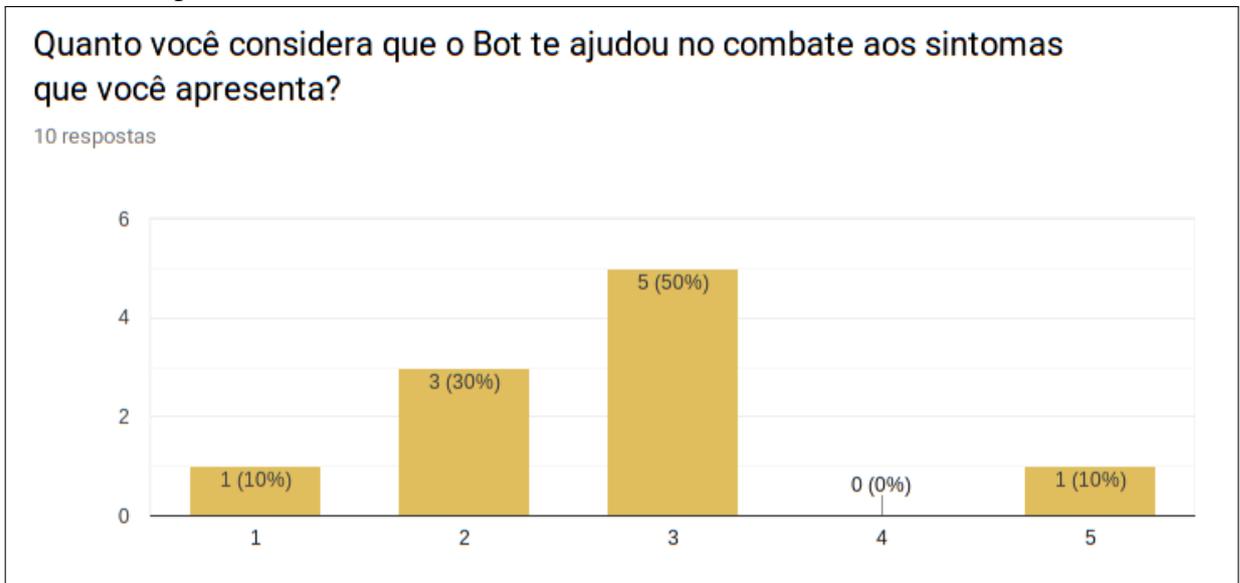


Fonte: Autoria própria.

participante foi "Quanto você considera que o *Bot* entendeu suas mensagens?". De acordo com as respostas demonstradas no gráfico na Figura 31, em relação à compreensão do *bot* e identificação correta das mensagens do usuário, numa escala de 1 (Entendeu Nada) a 5 (Entendeu tudo), a maioria das pessoas (6/10) respondeu abaixo de Regular.

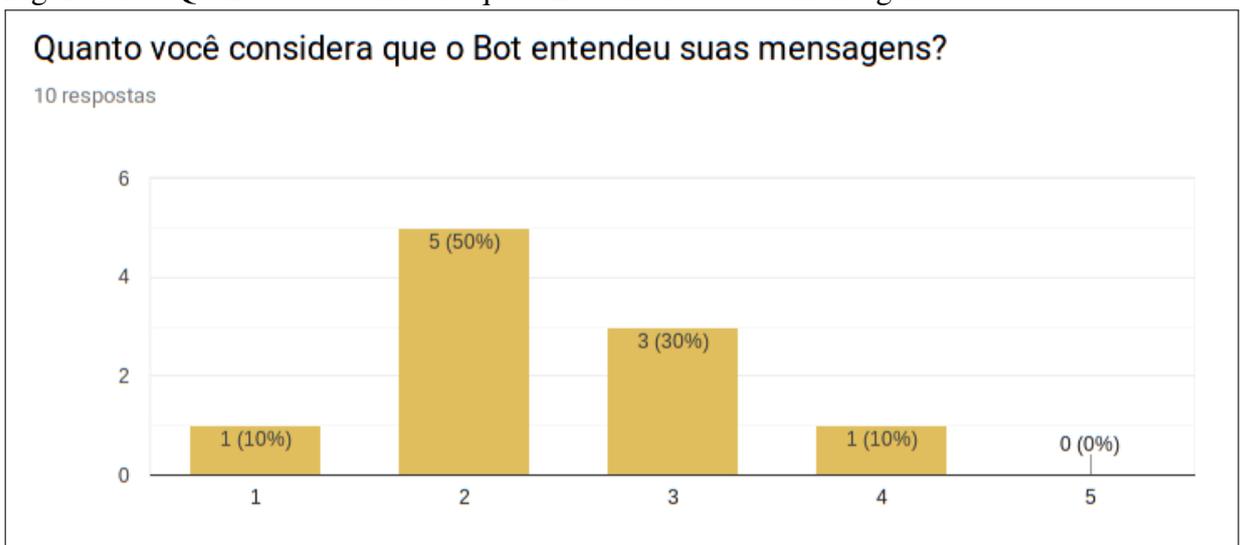
Por fim, comprova-se que o protótipo proposto possui funções cognitivas de nível regular que, juntamente com suas estratégias de acompanhamento, tornam a utilização do mesmo relevante para o usuário.

Figura 30 – "Quanto você considera que o Bot te ajudou no combate aos sintomas que você apresenta?"



Fonte: Autoria própria.

Figura 31 – "Quanto você considera que o Bot entendeu suas mensagens?"



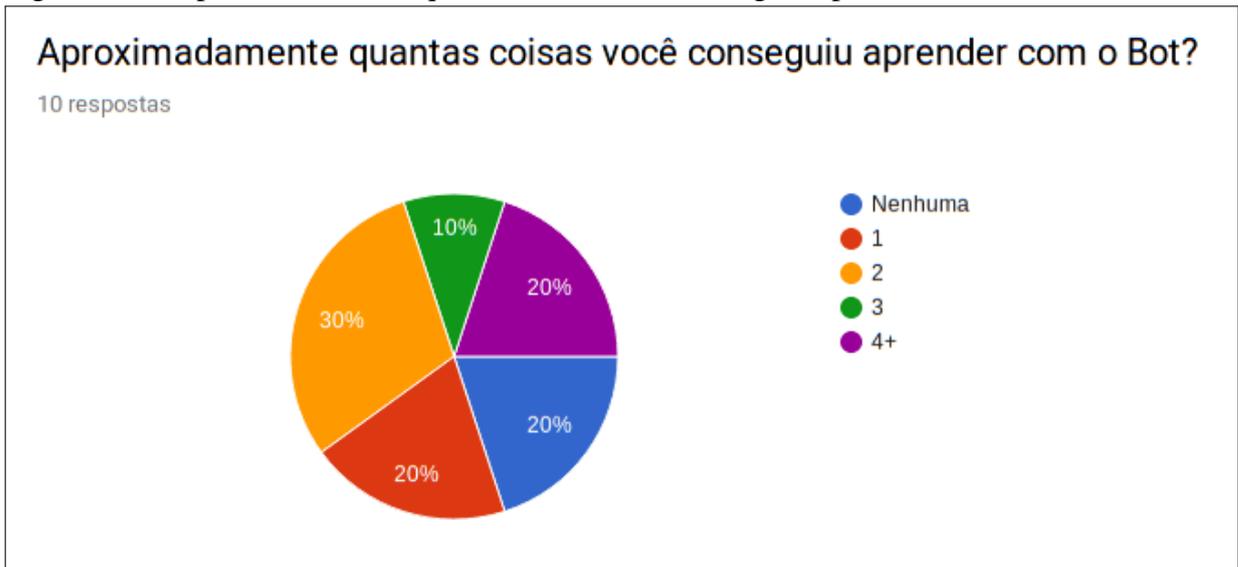
Fonte: Autoria própria.

4.2.3 Utilidade das informações

A primeira pergunta para avaliar o aspecto da utilidade das informações apresentadas pelo *bot* foi "Aproximadamente quantas coisas você conseguiu aprender com o *Bot*?". Pelas respostas apresentadas no gráfico na Figura 32 é possível observar que, dentre as informações apresentadas ao usuário pelo *bot* (músicas, informações sobre clima, etc), 6 pessoas informaram que aprenderam 2 coisas ou mais com o *bot*.

Outro questionamento respondido pelos participantes relacionado a esse aspecto foi "Quão útil para sua vida foi o que você aprendeu com o *Bot*?", onde, de acordo com o gráfico

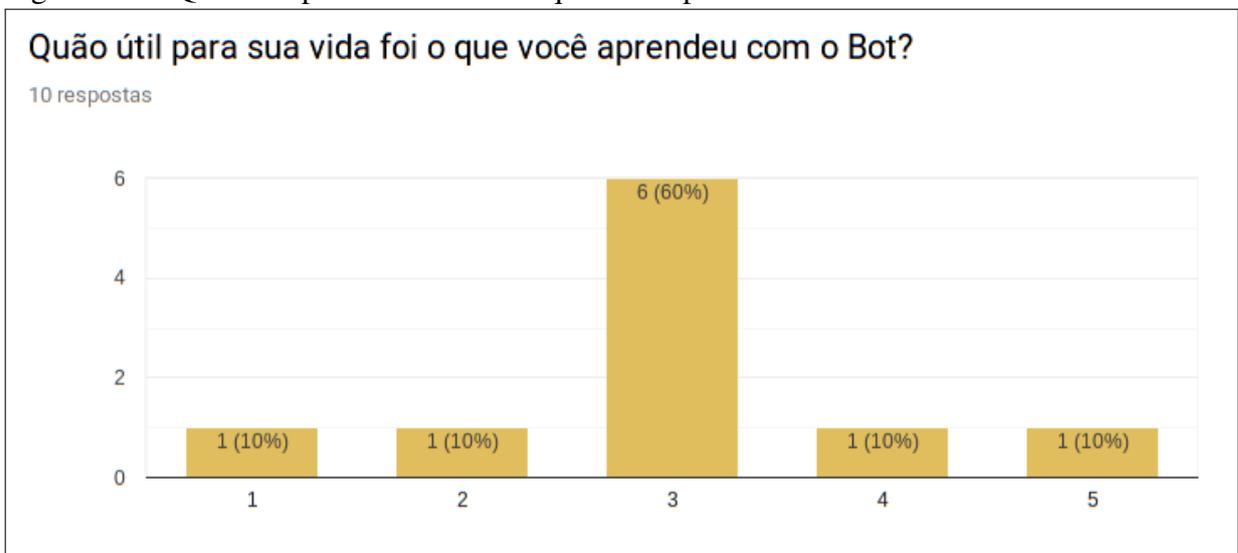
Figura 32 – "Aproximadamente quantas coisas você conseguiu aprender com o Bot?"



Fonte: Autoria própria.

demonstrado na Figura 33, 8 pessoas consideraram o aplicativo útil (nota 3), numa escala de 1 (Sem Utilidade) a 5 (Muito Útil).

Figura 33 – "Quão útil para sua vida foi o que você aprendeu com o Bot?"



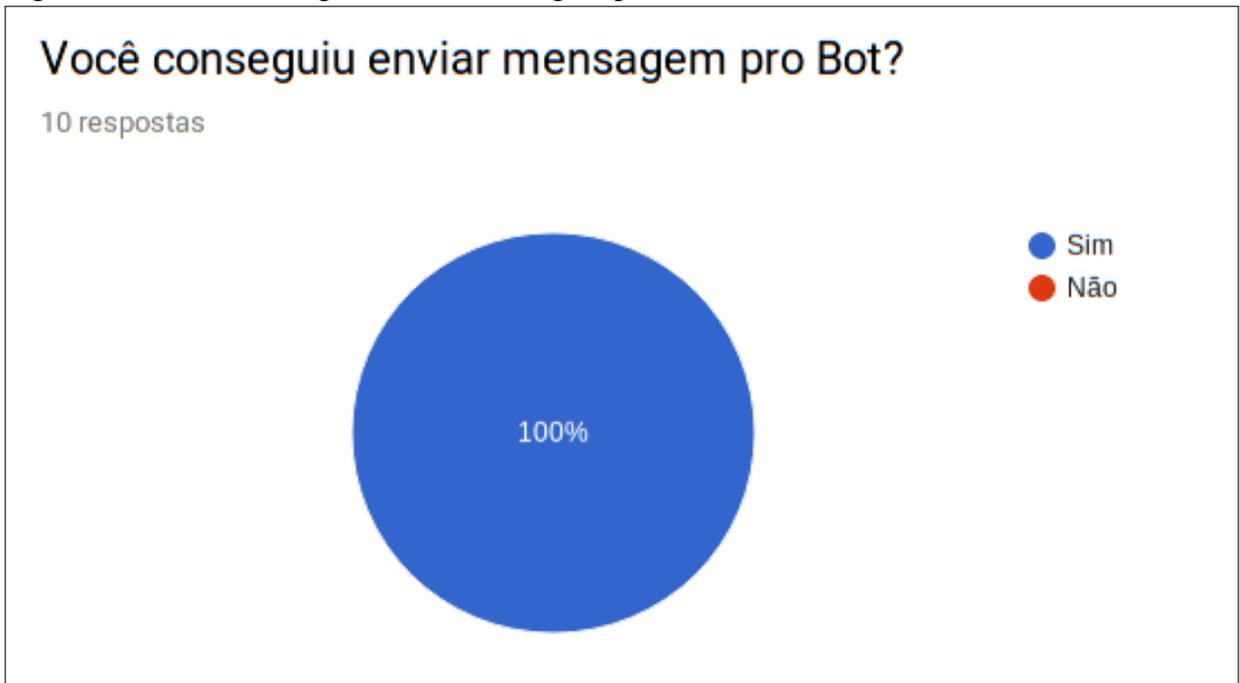
Fonte: Autoria própria.

Em resumo, é possível comprovar que as informações sobre atitudes básicas que em geral ajudam a combater sintomas da depressão, dicas de músicas para ouvir a fim de diminuir a ansiedade, dicas de interação social e informações de contatos para situações de emergência são práticas e úteis no dia a dia do usuário.

4.2.4 Performance

No que tange à performance da aplicação testada, é possível perceber através das respostas às perguntas "Você conseguiu enviar mensagem pro *Bot*?" e "Você teve algum problema na hora de instalar o aplicativo?", apresentadas nos gráficos exibidos nas Figuras 34 e 35, respectivamente, que durante o período de testes todos conseguiram enviar mensagens para o *bot* e nenhuma das pessoas teve problemas ao instalar a aplicação no seu dispositivo.

Figura 34 – "Você conseguiu enviar mensagem para o Bot?"



Fonte: Autoria própria.

Outra pergunta respondida relacionada quanto ao aspecto performance foi "Você percebeu algum problema ao receber as mensagens (ex: receber mensagens fora de contexto)?". De acordo com as respostas apresentadas no gráfico exibido na Figura 36, observa-se que apenas 3 dentre as 10 pessoas não tiveram algum problema ao receber mensagem do *bot*, como mensagens fora de contexto, por exemplo.

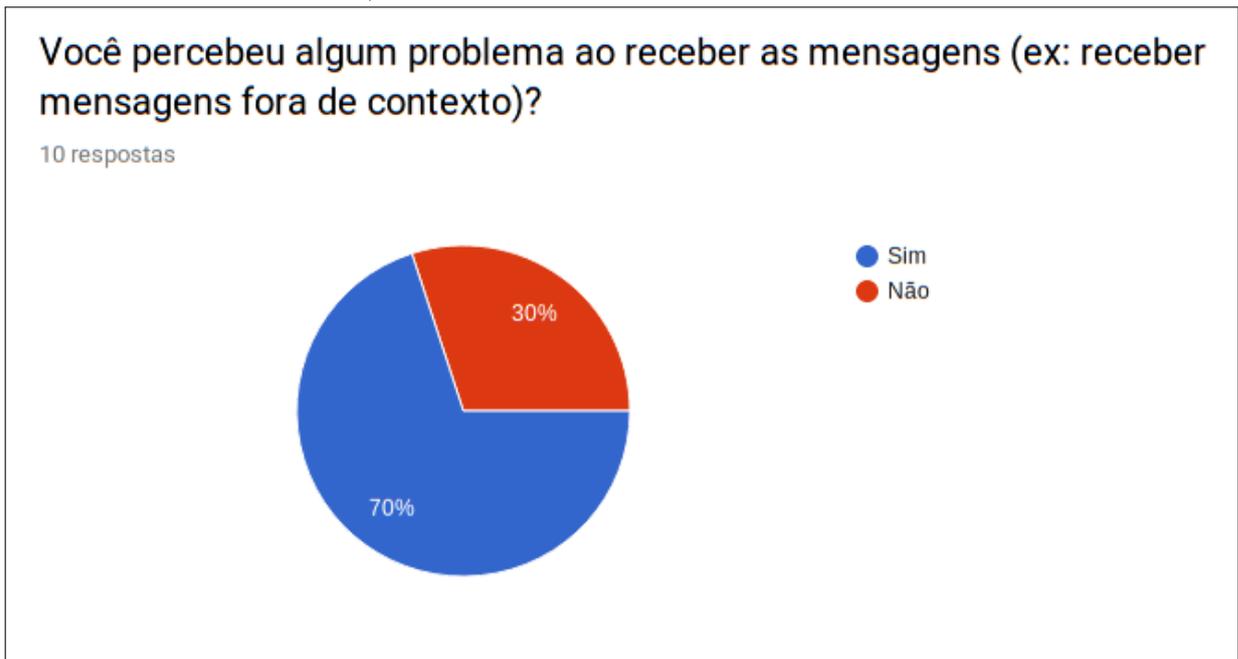
Ainda nesse aspecto, conforme o gráfico na Figura 37, ao se depararem com a pergunta "Alguma vez a aplicação "travou" no seu dispositivo?", apenas 2 pessoas reportaram que a aplicação "travou" alguma vez nos seus respectivos dispositivos. Além disso, ao responderem a pergunta "Quão demorado era pra receber mensagens do *Bot*?", mostrada no gráfico mostrado na Figura 38, numa escala de 1 (Muito demorado) a 5 (Rápido), a maioria (8 pessoas) atribuiu a nota máxima.

Figura 35 – "Você teve algum problema na hora de instalar o aplicativo?"



Fonte: Autoria própria.

Figura 36 – "Você percebeu algum problema ao receber as mensagens (ex: receber mensagens fora de contexto)?"



Fonte: Autoria própria.

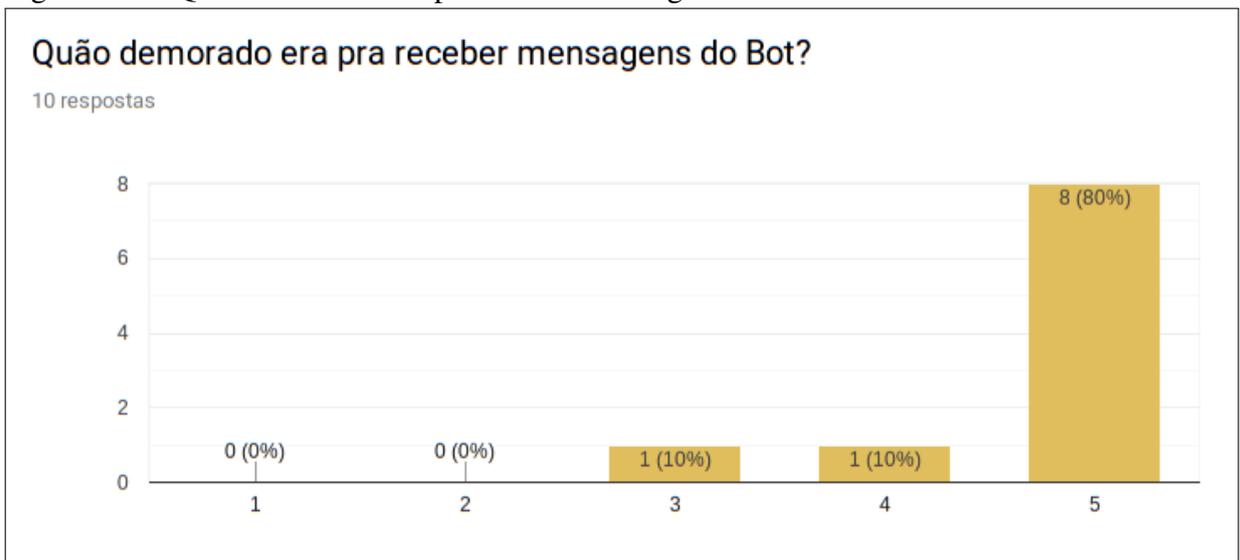
Tendo em vista essas informações, comprova-se que o protótipo de *chatbot* proposto possui performance regular pois apesar de enviar mensagens de forma rápida, ainda ocorrem momentos em que há problemas no recebimento de mensagens do *bot*, como reportado pela maioria dos usuários.

Figura 37 – "Alguma vez a aplicação "travou" no seu dispositivo?"



Fonte: Autoria própria.

Figura 38 – "Quão demorado era pra receber mensagens do Bot?"

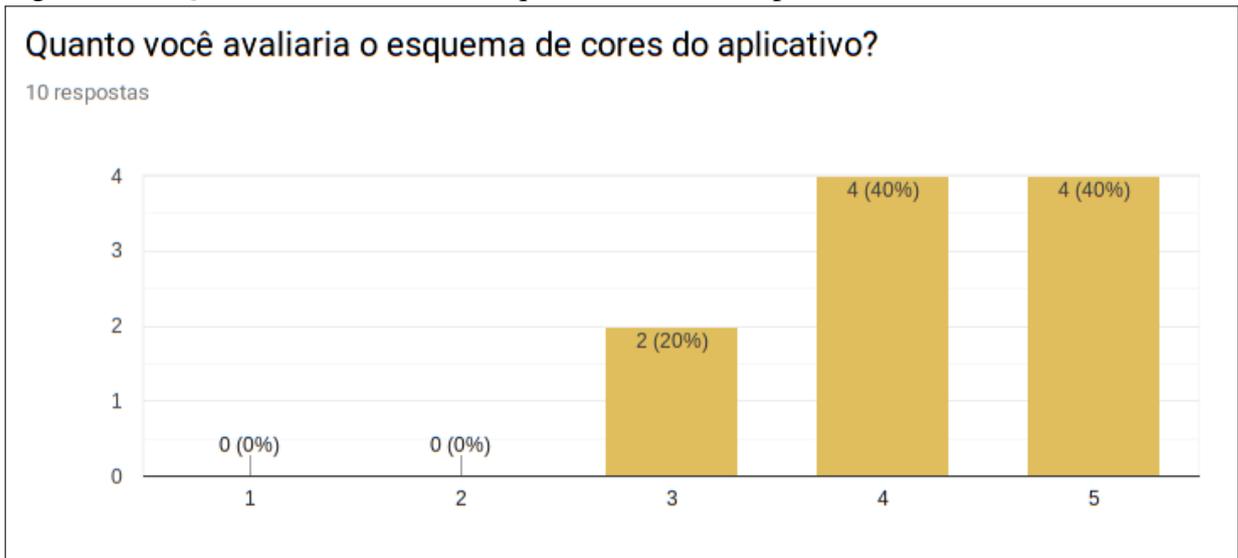


Fonte: Autoria própria.

4.2.5 Design da interface

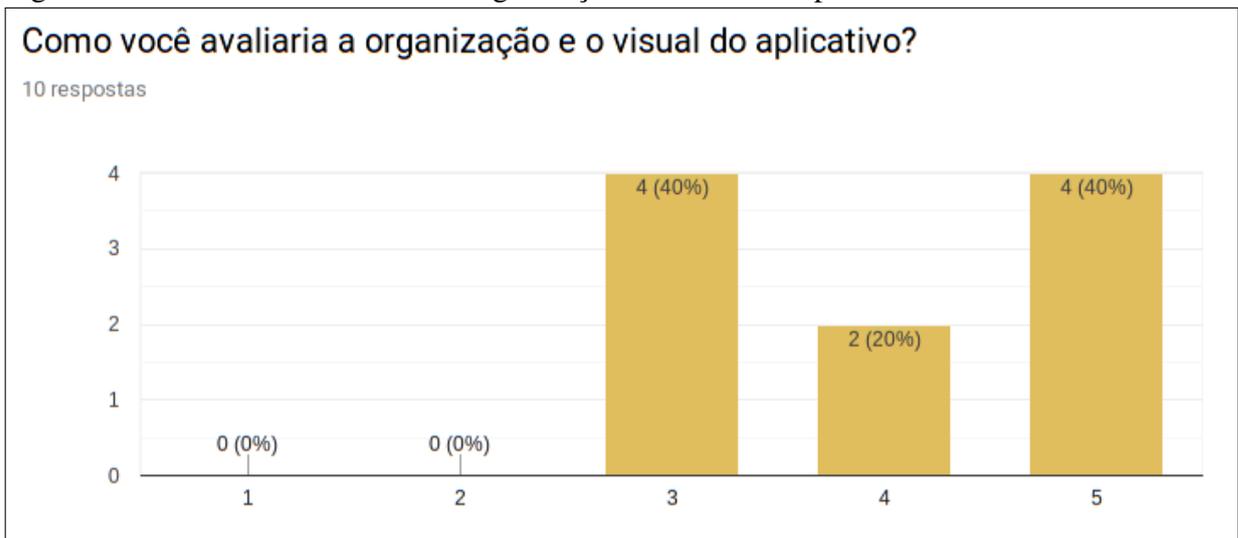
O último aspecto avaliado de forma bem geral foi design da interface. Assim, quando perguntados "Quanto você avaliaria o esquema de cores do aplicativo?" e "Como você avaliaria a organização e o visual do aplicativo?" os participantes avaliaram, de acordo com as respostas mostradas nos gráficos nas Figuras 39 e 40, respectivamente, o esquema de cores do aplicativo, a organização e a parte visual em geral do aplicativo. Sobre o esquema de cores (gráfico na Figura 39), 4 pessoas avaliaram como Bom (nota 4) e 4 pessoas como Muito Bom (nota 5). Quando

Figura 39 – "Quanto você avaliaria o esquema de cores do aplicativo?"



Fonte: Autoria própria.

Figura 40 – "Como você avaliaria a organização e o visual do aplicativo?"



Fonte: Autoria própria.

perguntados sobre como avaliavam a organização e a parte visual do aplicativo (gráfico na Figura 40), todos deram notas maiores ou iguais a 3, dentre os quais 4 responderam com Muito Bom (nota 5).

Assim, comprova-se que o protótipo de *chatbot* supre a necessidade de uma interface gráfica agradável e discreta para o usuário do mesmo.

4.3 Análise de Trabalhos Relacionados

Todos os trabalhos relacionados citados previamente foram de grande importância para fundamentar o desenvolvimento desta pesquisa, apresentando vantagens e desvantagens

que podem ser observadas se: considerarmos o idioma para o qual foram desenvolvidos; o quão discreto é o aplicativo de forma a evitar qualquer tipo de constrangimento da parte do usuário; o anonimato do usuário; possui alguma funcionalidade que sirva como estratégia de prevenção ao suicídio; o aplicativo possui funcionalidades voltadas para a redução dos sintomas da depressão.

Quanto ao idioma, é possível observar que a maioria dos trabalhos é em inglês, exceto o aplicativo Ajuda Já, que foi originalmente desenvolvido em português do Brasil. Isso revela uma grande vantagem desta pesquisa em relação à esses trabalhos, pois o protótipo que resulta da mesma se comunica em português do Brasil, facilitando o acesso ao usuário brasileiro.

No que diz respeito à discrição, ou seja, a quão discreto é o aplicativo de forma a evitar qualquer tipo de constrangimento da parte do usuário em relação ao uso de um aplicativo relacionado à saúde mental, é possível perceber que o *app Operation Reach Out* (MILITARY COMMUNITY AWARENESS, 2016) peca neste quesito, pois na tela de controle da exibição do vídeo há botões que diferem do padrão de tamanho estabelecido pelos mais populares *apps* de reprodução de vídeos, podendo despertar o interesse de pessoas ao redor ao chamar atenção para a sua interface, o que poderia constranger o usuário que utiliza esse tipo de aplicativo.

Em contrapartida, o protótipo resultante desta pesquisa deve se assemelhar ao máximo visualmente com os aplicativos de troca de mensagens mais populares para que o usuário possa interagir com o app da maneira mais discreta possível. Quando o aspecto observado é a privacidade do usuário, é possível perceber a maioria dos trabalhos relacionados possuem base própria para armazenar os dados do usuário, exceto *Woebot* ((FITZPATRICK; DARCY; VIERHILE, 2017), 2017). Isto acontece porque o *Woebot* é integrado ao Facebook Messenger, fazendo com que os dados das mensagens trocadas, incluindo imagens dos gráficos de evolução do usuário, a própria identidade do usuário por meio de seu perfil na rede social completo, entre outras coisas, sejam possível acesso ao Facebook.

Assim, tendo em vista essa informação, é possível perceber que o protótipo que resulta desta pesquisa segue o quesito privacidade, pois a aplicação deve armazenar as mensagens e os dados relacionados a elas num banco de dados próprio.

Outro aspecto observado ao analisar-se os trabalhos relacionados é se eles possuem alguma funcionalidade que auxilie na prevenção ao suicídio. Nesse quesito, somente o trabalho de Lucas et al. (2017) não possui, pois apesar de ter sido utilizado para sua pesquisa um modelo de *bot* aplicado a saúde mental, esse *bot* apenas detecta se o usuário possui sintomas de Transtorno do Estresse Pós Traumático e não contém nenhuma funcionalidade relacionada

diretamente à prevenção de suicídio. Essa situação já não se aplica ao protótipo resultante desta pesquisa, pois o mesmo deve armazenar um contato emergencial e no momento em que detectar uma crise, que pode originar uma possível tentativa de suicídio, o *chatbot* deve dar a opção do usuário ligar pro contato de emergência, além de exibir as informações de contato do Centro de Valorização da Vida e outros telefones úteis.

O último aspecto observado nos trabalhos relacionados é se possui-se alguma funcionalidade que estimule a redução de sintomas. Considerando esse quesito, é possível perceber que o trabalho de Lucas et al. (2017) não possui esta funcionalidade pois, como dito anteriormente, o *bot* utilizado apenas detecta os sintomas e sintomas de Transtorno do Estresse Pós Traumático. O *app* Ajuda Já foi desenvolvido com foco em pessoas que estão prestes a cometer suicídio, porém também não possui alguma funcionalidade focada em auxiliar na diminuição de sintomas da depressão.

Já o protótipo que resulta desta pesquisa leva vantagem nesse aspecto, pois pode exibir sugestões de atividades baseadas no clima da localidade do usuário, sugestões de atividades que antes eram prazerosas personalizadas de acordo com informações coletadas na conversa com o usuário, entre outras atividades que podem auxiliar na diminuição dos sintomas da depressão.

Logo, a comparação dos trabalhos relacionados com o protótipo de *chatbot* que foi desenvolvido como resultado desta pesquisa, segundo os aspectos anteriormente citados, pode ser percebida conforme com o que é mostrado na tabela 2.

Tabela 2 – Comparação dos trabalhos relacionados

	Operation Reach		Reporting Mental Health Symptoms: Breaking Down Barriers to Care with Virtual Human Interviewers		
	Out	Ajuda Já	Interviewers	Woebot	Presente trabalho
Idioma Português do Brasil	✗	✓	✗	✗	✓
Discrição	✗	✓	✓	✓	✓
Privacidade	✓	✓	✓	✗	✓
Funcionalidade relacionada à prevenção ao suicídio	✓	✓	✗	✓	✓
Funcionalidade de estímulo à redução de sintomas	✗	✗	✗	✓	✓

Fonte: Autoria própria.

5 CONCLUSÕES E TRABALHOS FUTUROS

Ao longo deste trabalho foram demonstrados vários aspectos do desenvolvimento de um protótipo de *chatbot*, em Português do Brasil, e da disponibilização do mesmo na forma de um aplicativo para dispositivos Android, a fim de que ajudasse no combate aos sintomas da depressão e na prevenção de suicídios. Concomitantemente, o trabalho apresentou uma pesquisa que buscou avaliar o impacto e a relevância do *chatbot* desenvolvido na vida de pessoas com sintomas ou que já foram diagnosticadas com depressão.

Para a autora do presente trabalho, o desenvolvimento do mesmo foi altamente importante para ampliar seus conhecimentos sobre *chatbots*, desenvolvimento de aplicações para Android e transtorno depressivo maior, contribuindo para seu crescimento acadêmico e profissional.

O aumento do número de casos de pessoas com depressão no Brasil fomenta a criação de ferramentas que usam tecnologia para auxiliar no combate a essa doença. Desse modo, um *chatbot* no idioma deste país e que ajude tanto no combate aos sintomas da depressão como na prevenção de suicídios pode ser um grande aliado da psicoterapia tradicional no tratamento dessa doença. Assim, reforça-se a importância do protótipo de *chatbot* resultante desta pesquisa não só para as pessoas que sofrem de transtorno depressivo maior como também para os profissionais de saúde que as tratam.

A abordagem de desenvolvimento do protótipo de *chatbot* utilizando IBM Watson Assistant se mostrou bastante satisfatória devido à facilidade na utilização desta API e outras vantagens já apresentadas no decorrer desta pesquisa. Já no que diz respeito ao desenvolvimento da aplicação de teste, concluiu-se que a abordagem utilizando o Android Studio, SQLite e Firebase foi satisfatória, apesar de que a rotina de gerar as notificações cumpriu seu propósito mas ainda não foi o ideal de desenvolvimento desse tipo de serviço.

Após os testes da aplicação para Android desenvolvida, a análise dos resultados do questionário conseguiu mostrar que o *chatbot* desenvolvido contribuiu para a redução de sintomas da depressão, além de possuir funções cognitivas de nível regular, o que contribui, juntamente com suas estratégias de acompanhamento, para que o mesmo seja relevante para o usuário. Para mais, também foi evidenciado que as informações apresentadas pelo *chatbot* durante a utilização do aplicativo se mostraram práticas e úteis na vida do usuário. Assim, a partir destas considerações, verifica-se que o objetivo de desenvolver um protótipo de *chatbot* para dispositivos móveis que pudesse ser utilizado como ferramenta de auxílio e prevenção à

depressão e ao suicídio foi cumprido.

No que diz respeito às contribuições tecnológicas e científicas, é possível perceber que este trabalho apresentou o uso *chatbot* como ferramenta numa área onde assistentes virtuais são pouco explorados, que é a área da saúde mental. Além disso, apresentou o IBM Watson Assistant como uma ferramenta completa para o desenvolvimento desse tipo de *chatbots*. É possível perceber também que, além deste trabalho ter sido somado aos poucos trabalhos no Brasil que propõe o uso de um *chatbot* que conversa em Português do Brasil no combate à depressão, ele abriu precedente para novas pesquisas de forma que incentiva mais pesquisadores a explorarem essa área.

5.1 Limitações e Trabalhos Futuros

Dentre as limitações deste trabalho observou-se que tanto os *datasets* de sugestões de interação social quanto o de músicas ou fatos e informações sobre combate à depressão possuem poucas amostras, o que faz com que após um certo tempo de interação, o *chatbot* comece a sugerir e exibir informações repetidas. Outra limitação que contribui para que a interação com o *chatbot* torne-se repetitiva é o fato de possuir poucos fluxos de diálogo, se levada em consideração a infinidade de possibilidades de interação do usuário.

Foi percebido também que a exibição das respostas do *chatbot* na aplicação são muito rápidas, o que faz com que haja pouca naturalidade e fluidez na interação com o usuário. Outra observação que foi feita é que o período de testes, de uma semana, foi menor do que de o período de testes de outras pesquisas e que a quantidade de usuários foi pequena, o que possivelmente interferiu na riqueza de dados coletados. Observou-se ainda que o questionário de avaliação aplicado ao final do período de testes possui algumas perguntas muito abrangentes, o que dificulta uma análise de resultados mais precisa principalmente em relação à utilidade das informações apresentadas pelo *chatbot* na interação e à qualidade do design da interface gráfica do aplicativo.

Em vista dessas observações e com o objetivo de no futuro desenvolver melhor esta pesquisa, sugere-se que um *dataset* mais completo seja construído levando em conta, por exemplo, sites de conteúdo turístico de acordo com a localidade do usuário. O mesmo vale para o *dataset* de fatos e informações sobre combate à depressão - que podem conter informações relacionadas à vários tipos de abordagens da psicologia e como elas veem o combate à depressão - e para o *dataset* de músicas, que pode ser construído baseado em *playlists* populares de serviços

de *streaming* de músicas relacionadas ao combate à depressão e ansiedade, podendo ser estudado também uma possível integração com a API destes tipos de serviço. Além disso, sugere-se o aperfeiçoamento dos fluxos de diálogo já existentes e criação de novos fluxos de diálogo, além de aumentar a quantidade de possíveis respostas em cada nó do diálogo para que o mesmo abranja mais contextos, tudo isso com acompanhamento de uma equipe de psicólogos de diferentes abordagens.

Outro aspecto sugerido a ser desenvolvido futuramente é agregar *intents* semelhantes em uma só *intent* para diminuir, mesmo que de forma leve, o tempo de resposta à chamada da API do Watson.

Sugere-se também um estudo mais aprofundado do desenvolvimento de uma interface gráfica mais amigável e fluída, possibilitando também formulação de perguntas mais específicas a serem incluídas no questionário final para avaliar tanto esse aspecto quanto os outros aspectos citados anteriormente (influência do *chatbot* na diminuição de sintomas, performance, etc).

Quanto aos testes, sugere-se que os mesmos sejam realizados por um tempo maior e sejam direcionados a diferentes grupos etários, por região do Brasil, entre outros grupos específicos aos quais a utilização de um *chatbot* como ferramenta de auxílio ao combate à depressão e ao suicídio sejam relevantes.

REFERÊNCIAS

- ALPAYDIN, E. **Introduction to machine learning**. [S.l.]: MIT press, 2009. 584 p.
- AMERICAN PSYCHIATRIC ASSOCIATION. **DSM-5: manual diagnóstico e estatístico de transtornos mentais**. 5. ed. Porto Alegre: Artmed, 2014. 992 p.
- ANDROID DEVELOPERS. **Fundamentos de aplicativos**. 2018. Disponível em: <<https://developer.android.com/guide/components/fundamentals.html?hl=pt-BR>>. Acesso em: 16 abr. 2018.
- BATTISTELLI, J. **Google Firebase for dummies: o que é e como funciona a plataforma**. 2017. Disponível em: <<https://blog.mastertech.com.br/tecnologia/google-firebase-for-dummies-o-que-e-e-como-funciona-plataforma/>>. Acesso em: 15 mar. 2018.
- CURTIN, M. **Neuroscience Says Listening to This Song Reduces Anxiety by Up to 65 Percent**. 2017. Disponível em: <<https://www.inc.com/melanie-curtin/neuroscience-says-listening-to-this-one-song-reduces-anxiety-by-up-to-65-percent.html>>. Acesso em: 12 nov. 2017.
- DAVYDOVA, O. chatbot platforms: A comparative table. **Chatbots Journal chatbotsjournal.com/25-chatbot-platforms-a-comparative-table-aeefc932eaff**, 2017.
- DEITEL, H.; DEITEL, P.; DEITEL, A. **Android: Como programar**. [S.l.]: Bookman Editora, 2015.
- DEVAULT, D.; ARTSTEIN, R.; BENN, G.; DEY, T.; FAST, E.; GAINER, A.; GEORGILA, K.; GRATCH, J.; HARTHOLT, A.; LHOMET, M. *et al.* Simsensei kiosk: A virtual human interviewer for healthcare decision support. In: INTERNATIONAL FOUNDATION FOR AUTONOMOUS AGENTS AND MULTIAGENT SYSTEMS. **Proceedings of the 2014 international conference on Autonomous agents and multi-agent systems**. [S.l.], 2014. p. 1061–1068.
- EASON, J. **Support Ended for Eclipse Android Developer Tools**. 2017. Disponível em: <<https://android-developers.googleblog.com/2016/11/support-ended-for-eclipse-android.html?m=1>>. Acesso em: 15 mar. 2018.
- FARIA, L. C. **Desenvolvendo Chatbots com Watson Conversation**. 2017. Disponível em: <<https://gago.io/blog/anatomia-de-um-chatbot/>>. Acesso em: 17 abr. 2018.
- FERRUCCI, D. A. Introduction to “this is watson”. **IBM Journal of Research and Development**, IBM, v. 56, n. 3.4, p. 1–1, 2012.
- FIREBASE. **Cloud Functions para Firebase**. 2018. Disponível em: <<https://firebase.google.com/docs/functions/?hl=pt-br>>. Acesso em: 18 mar. 2018.
- FIREBASE. **Documentação**. 2018. Disponível em: <<https://firebase.google.com/docs>>. Acesso em: 18 mar. 2018.
- FIREBASE. **Firestore Cloud Messaging**. 2018. Disponível em: <<https://firebase.google.com/docs/cloud-messaging/?hl=pt-br>>. Acesso em: 18 mar. 2018.

- FIREBASE. **Google Cloud Functions e Firebase**. 2018. Disponível em: <<https://firebase.google.com/docs/functions/functions-and-firebase?hl=pt-br>>. Acesso em: 18 mar. 2018.
- FIREBASE. **Sobre as mensagens do FCM**. 2018. Disponível em: <https://firebase.google.com/docs/cloud-messaging/concept-options.html?hl=pt-br#notifications_and_data_messages>. Acesso em: 18 mar. 2018.
- FITZPATRICK, K. K.; DARCY, A.; VIERHILE, M. Delivering cognitive behavior therapy to young adults with symptoms of depression and anxiety using a fully automated conversational agent (woebot): a randomized controlled trial. **JMIR mental health**, JMIR Publications Inc., Toronto, Canada, v. 4, n. 2, p. e19, 2017.
- KELLY, J. E. Computing, cognition and the future of knowing. **Whitepaper, IBM Research**, v. 2, 2015.
- KROENKE, K.; SPITZER, R. L.; WILLIAMS, J. B. The phq-9: validity of a brief depression severity measure. **Journal of general internal medicine**, Wiley Online Library, v. 16, n. 9, p. 606–613, 2001.
- LECHETA, R. R. **Google Android-3ª Edição: Aprenda a criar aplicações para dispositivos móveis com o Android SDK**. [S.l.]: Novatec Editora, 2013.
- LIDDY, E. D. Enhanced text retrieval using natural language processing. **Bulletin of the American Society for Information Science and Technology**, Wiley Online Library, v. 24, n. 4, p. 14–16, 1998.
- LUCAS, G. M.; RIZZO, A.; GRATCH, J.; SCHERER, S.; STRATOU, G.; BOBERG, J.; MORENCY, L.-P. Reporting mental health symptoms: breaking down barriers to care with virtual human interviewers. **Frontiers in Robotics and AI**, Frontiers, v. 4, p. 51, 2017.
- MAZON, S. **Desenvolvendo Chatbots com Watson Conversation**. 2018. Disponível em: <<https://www.ibm.com/developerworks/br/library/desenvolvendo-chatbots-com-watson-conversation/index.html>>. Acesso em: 17 abr. 2018.
- MAZON, S. **Seja bem-vindo: Watson Assistant!** 2018. Disponível em: <<https://www.ibm.com/developerworks/br/library/bem-vindo-ao-watson-assistant/index.html>>. Acesso em: 17 abr. 2018.
- MILITARY COMMUNITY AWARENESS INC. **SUICIDE PREVENTION IN THE DIGITAL AGE: Operation Reach Out: A Suicide Prevention App**. 2016. Disponível em: <http://www.4mca.com/catalog/MCA_white_paper.pdf>. Acesso em: 20 nov. 2017.
- NORVIG, P.; RUSSELL, S. **Inteligência Artificial: Tradução da 3ª Edição**. [S.l.]: Elsevier Brasil, 2014. v. 1.
- SQLITE. **About SQLite**. 2018. Disponível em: <<https://www.sqlite.org/about.html>>. Acesso em: 03 jun. 2018.
- STENBECK, F. **What languages does IBM Watson support?** 2015. Disponível em: <<https://fredrikstenbeck.com/what-languages-does-ibm-watson-support/>>. Acesso em: 16 mar. 2018.

TAMPLIN, J. Firebase is joining google! **Firestore Blog**, 2014. Disponível em: <<https://firebase.googleblog.com/2014/10/firebase-is-joining-google.html?m=1>>. Acesso em: 09 abr. 2018.

TOROUS, J.; FRIEDMAN, R.; KESHAVAN, M. Smartphone ownership and interest in mobile applications to monitor symptoms of mental health conditions. **JMIR mHealth and uHealth**, JMIR Publications Inc., Toronto, Canada, v. 2, n. 1, p. 1–8, 2014. Disponível em: <https://mhealth.jmir.org/article/viewFile/mhealth_v2i1e2/2>. Acesso em: 23 out. 2017.

VIEIRA, R.; LOPES, L. Processamento de linguagem natural e o tratamento computacional de linguagens científicas. **EM CORPORA**, p. 183–201, 2010. Disponível em: <<http://www.pucrs.br/edipucrs/linguagensespecializadasemcorpora.pdf>>. Acesso em: 16 dez. 2017.

WEIZENBAUM, J. Eliza—a computer program for the study of natural language communication between man and machine. **Communications of the ACM**, v. 9, n. 1, p. 1–6, 1966. Disponível em: <http://www.universelle-automation.de/1966_Boston.pdf>. Acesso em: 13 out. 2017.

WORLD HEALTH ORGANIZATION. **Depression: let's talk**. 2017. Disponível em: <http://www.who.int/mental_health/management/depression/en/>. Acesso em: 20 out. 2017.

WORLD HEALTH ORGANIZATION. **Depression**. 2018. Disponível em: <<https://www.who.int/en/news-room/fact-sheets/detail/depression>>. Acesso em: 18 maio 2018.

APÊNDICE A – TERMO DE CONSENTIMENTO LIVRE E ESCLARECIDO



UFMA – Universidade Federal do Maranhão
CCET – Centro de Ciências Exatas e Tecnológicas
Curso de Ciência da Computação

Chatbot como ferramenta no auxílio ao combate a depressão e prevenção ao suicídio

TERMO DE CONSENTIMENTO LIVRE E ESCLARECIDO

(TERMINOLOGIA OBRIGATÓRIO EM ATENDIMENTO A RESOLUÇÃO 510/16 –CNS-MS)

A pesquisa “**Chatbot como ferramenta no auxílio ao combate a depressão e prevenção ao suicídio**” tem como objetivo analisar o desenvolvimento de um chatbot que ajude pessoas com sintomas de depressão, avaliando a eficácia das suas funções cognitivas no desenvolvimento da habilidade de comunicação do usuário, no aumento do interesse do mesmo em atividades que favoreçam uma melhoria na qualidade de vida e na redução da incidência de comportamento suicida. Nenhum dos procedimentos será invasivo e não causará nenhum desconforto ou risco à sua saúde. Em caso de dúvidas, você será totalmente esclarecido pelos responsáveis da pesquisa antes e durante a realização do experimento, além da possibilidade de entrar em contato por um dos meios divulgados abaixo. É importante salientar que o produto desta pesquisa NÃO SUBSTITUI psicoterapia ou qualquer outro tipo de acompanhamento profissional especializado.

Este “Termo de Consentimento Livre e Esclarecido” atende a Resolução 510/16-CNS-MS.

Eu, _____,

RG _____ - SSP/_____, estando ciente das informações acima lidas, concordo em participar da pesquisa “**Chatbot como ferramenta no auxílio ao combate a depressão e prevenção ao suicídio**” e entendo que as informações cedidas por mim são confidenciais, autorizando a sua divulgação no meio científico e acadêmico de forma anônima e global, tendo a minha identidade totalmente preservada. Estou ciente de que sou voluntário e, portanto, não receberei nenhum benefício por participar desta pesquisa, bem como não terei ônus algum. Tenho total liberdade para aceitar ou recusar fazer parte deste estudo e sei que a minha recusa, em qualquer momento do experimento, não acarretará nenhum prejuízo para mim.

_____, _____ de _____ de _____.

Assinatura do voluntário

Prof. Me. Débora Rodrigues Stefanello.

Êmile Cunha Lopes.

Êmile Cunha Lopes
 Pesquisadora
 emile.clopes@gmail.com
 (98) 9 8158-3666

Prof. Me. Débora Rodrigues Stefanello
 Pesquisadora
 deborasi92@gmail.com
 (98) 9 9158-1100

APÊNDICE B – QUESTIONÁRIO

Feedback sobre o Jean Bot

Obrigado por ter participado

Queremos saber seu feedback para continuar melhorando a logística e o conteúdo. Responda a esta pesquisa rápida e conte-nos sua opinião. As respostas serão anônimas.

*Obrigatório

Sintomas 1/2

Nas duas semanas ANTES do uso do Jean Bot, em quantos dias você foi afetado(a) por algum dos seguintes problemas?

1. Tive pouco interesse ou prazer em fazer as coisas *

Marcar apenas uma oval.

- Nunca
- Em vários dias
- Em mais da metade do número de dias
- Em quase todos os dias

2. Senti desânimo, desalento ou falta de esperança *

Marcar apenas uma oval.

- Nunca
- Em vários dias
- Em mais da metade do número de dias
- Em quase todos os dias

3. Tive dificuldade em adormecer ou em dormir sem interrupções, ou dormi demais *

Marcar apenas uma oval.

- Nunca
- Em vários dias
- Em mais da metade do número de dias
- Em quase todos os dias

4. Senti cansaço ou falta de energia *

Marcar apenas uma oval.

- Nunca
- Em vários dias
- Em mais da metade do número de dias
- Em quase todos os dias

5. Tive falta ou excesso de apetite *

Marcar apenas uma oval.

- Nunca
- Em vários dias
- Em mais da metade do número de dias
- Em quase todos os dias

6. Senti que não gosto de mim própria(o) - ou que sou um falho(a) ou me desiludi a mim próprio(a) ou à minha família *

Marcar apenas uma oval.

- Nunca
- Em vários dias
- Em mais da metade do número de dias
- Em quase todos os dias

7. Tive dificuldade em concentrar-me nas coisas, como ao ler o jornal ou ver televisão *

Marcar apenas uma oval.

- Nunca
- Em vários dias
- Em mais da metade do número de dias
- Em quase todos os dias

8. Movimentei-me ou falei tão lentamente que outras pessoas poderão ter notado. Ou o oposto: estive agitado(a) a ponto de andar de um lado para o outro muito mais do que é habitual *

Marcar apenas uma oval.

- Nunca
- Em vários dias
- Em mais da metade do número de dias
- Em quase todos os dias

9. Pensei que seria melhor está morto(a), ou magoar-me a mim próprio(a) de alguma forma *

Marcar apenas uma oval.

- Nunca
- Em vários dias
- Em mais da metade do número de dias
- Em quase todos os dias

Sintomas 2/2

DURANTE o uso do Jean Bot em quantos dias você foi afetado(a) por algum dos seguintes problemas?

10. **Tive pouco interesse ou prazer em fazer as coisas ***

Marcar apenas uma oval.

- Nunca
- Em vários dias
- Em mais da metade do número de dias
- Em quase todos os dias

11. **Senti desânimo, desalento ou falta de esperança ***

Marcar apenas uma oval.

- Nunca
- Em vários dias
- Em mais da metade do número de dias
- Em quase todos os dias

12. **Tive dificuldade em adormecer ou em dormir sem interrupções, ou dormi demais ***

Marcar apenas uma oval.

- Nunca
- Em vários dias
- Em mais da metade do número de dias
- Em quase todos os dias

13. **Senti cansaço ou falta de energia ***

Marcar apenas uma oval.

- Nunca
- Em vários dias
- Em mais da metade do número de dias
- Em quase todos os dias

14. **Tive falta ou excesso de apetite ***

Marcar apenas uma oval.

- Nunca
- Em vários dias
- Em mais da metade do número de dias
- Em quase todos os dias

15. **Senti que não gosto de mim própria(o) - ou que sou um falho(a) ou me desiludi a mim próprio(a) ou à minha família ***

Marcar apenas uma oval.

- Nunca
- Em vários dias
- Em mais da metade do número de dias
- Em quase todos os dias

16. **Tive dificuldade em concentrar-me nas coisas, como ao ler o jornal ou ver televisão ***

Marcar apenas uma oval.

- Nunca
- Em vários dias
- Em mais da metade do número de dias
- Em quase todos os dias

17. **Movimentei-me ou falei tão lentamente que outras pessoas poderão ter notado. Ou o oposto: estive agitado(a) a ponto de andar de um lado para o outro muito mais do que é habitual ***

Marcar apenas uma oval.

- Nunca
- Em vários dias
- Em mais da metade do número de dias
- Em quase todos os dias

18. **Pensei que seria melhor está morto(a), ou magoar-me a mim próprio(a) de alguma forma ***

Marcar apenas uma oval.

- Nunca
- Em vários dias
- Em mais da metade do número de dias
- Em quase todos os dias

Sobre o bot

19. **Em quantos dias do período de teste, aproximadamente, você interagiu com o bot? ***

Marcar apenas uma oval.

- Nunca
- Em vários dias, mas não todos
- Em mais da metade do número de dias
- Todos os dias

20. **Como você avaliaria as estratégias de acompanhamento do Bot? ***

Marcar apenas uma oval.

	1	2	3	4	5	
Péssimo	<input type="radio"/>	Muito bom				

21. **Quanto você considera que o Bot te ajudou no combate aos sintomas que você apresenta? ***

Marcar apenas uma oval.

	1	2	3	4	5	
Ajudou em nada	<input type="radio"/>	Ajudou muito				

22. **Quanto você considera que o Bot entendeu suas mensagens? ***

Marcar apenas uma oval.

1 2 3 4 5

Entendeu nada Entendeu tudo

23. **Aproximadamente quantas coisas você conseguiu aprender com o Bot? ***

Marcar apenas uma oval.

- Nenhuma
- 1
- 2
- 3
- 4+

24. **Quão útil para sua vida foi o que você aprendeu com o Bot? ***

Marcar apenas uma oval.

1 2 3 4 5

Sem utilidade Muito útil

Aplicativo 1/2

25. **Você conseguiu enviar mensagem pro Bot? ***

Marcar apenas uma oval.

- Sim
- Não

26. **Você teve algum problema na hora de instalar o aplicativo? ***

Marcar apenas uma oval.

- Sim
- Não

27. **Você percebeu algum problema ao receber as mensagens (ex: receber mensagens fora de contexto)? ***

Marcar apenas uma oval.

- Sim
- Não

28. **Alguma vez a aplicação “travou” no seu dispositivo? ***

Marcar apenas uma oval.

- Sim
- Não

29. **Quão demorado era pra receber mensagens do Bot? ***

Marcar apenas uma oval.

	1	2	3	4	5	
Muito demorado	<input type="radio"/>	Rápido				

Aplicativo 2/2

30. **Quanto você avaliaria o esquema de cores do aplicativo? ***

Marcar apenas uma oval.

	1	2	3	4	5	
Péssimo	<input type="radio"/>	Muito bom				

31. **Como você avaliaria a organização e o visual do aplicativo? ***

Marcar apenas uma oval.

	1	2	3	4	5	
Péssimo	<input type="radio"/>	Muito bom				

Muito obrigada!

Mais uma vez, obrigada por participar desta pesquisa. Espero que tenha tido alguma influência na sua vida.

32. **Deixe aqui algum comentário, crítica ou sugestão**
