

**CENTRO UNIVERSITÁRIO DE JOÃO PESSOA - UNIPÊ  
PRÓ-REITORIA ACADÊMICA  
CURSO DE GESTÃO DA TECNOLOGIA DA INFORMAÇÃO**

**ANTONIO LEONARDO FALCÃO FILHO  
GYAN PATRICK LIMA ALVES  
MÁRCIO HERIBERTO DE BRITO**

**DESENVOLVIMENTO DE CHATBOT PARA A COORDENAÇÃO DOS CURSOS  
DE PÓS-GRADUAÇÃO NA ÁREA DE DIREITO**

**JOÃO PESSOA - PB  
2017**

**ANTONIO LEONARDO FALCÃO FILHO  
GYAN PATRICK LIMA ALVES  
MÁRCIO HERIBERTO DE BRITO**

**DESENVOLVIMENTO DE CHATBOT PARA A COORDENAÇÃO DOS CURSOS  
DE PÓS-GRADUAÇÃO NA ÁREA DE DIREITO**

Relatório Técnico apresentado ao Centro  
Universitário de João Pessoa, como requisito para  
a conclusão do curso de Gestão da Tecnologia da  
Informação.

Orientador: Prof. Me. Ricardo Roberto de Lima

**JOÃO PESSOA - PB  
2017**

A474d Falcão Fh, Antônio Leonardo.

Desenvolvimento de Chatbot para a Coordenação de cursos de pós-graduação na área de direito / Antônio Leonardo Falcão Filho, Gyan Patrick Lima Alves, Márcio Heriberto de Brito. – João Pessoa, 2017.  
79f.

Orientador: Prof. Me. Ricardo Roberto de Lima.  
Relatório Técnico Aplicado (Curso de Gestão da Tecnologia da Informação) – Centro Universitário de João Pessoa - UNIPÊ.

1. ChatBot. 2. Inteligência Artificial. 3. Processamento de Linguagem Natural. I. Título.

**ANTONIO LEONARDO FALCÃO FILHO  
GYAN PATRICK LIMA ALVES  
MÁRCIO HERIBERTO DE BRITO**

**DESENVOLVIMENTO DE CHATBOT PARA A COORDENAÇÃO DOS CURSOS  
DE PÓS-GRADUAÇÃO NA ÁREA DE DIREITO**

**IMPORTANTE:** ESSE É APENAS UM  
TEXTO DE EXEMPLO DE FOLHA DE  
APROVAÇÃO. O ALUNO DEVERÁ  
SOLICITAR UMA FOLHA DE  
APROVAÇÃO PARA SEU TRABALHO  
NA SECRETARIA DO SEU CURSO (OU  
DEPARTAMENTO).

Trabalho aprovado. JOÃO PESSOA, \_\_\_\_ / \_\_\_\_ / \_\_\_\_.

---

**Prof. Msc. Ricardo Roberto de Lima**  
Orientador

---

**Prof. Pedro de Araújo Amorim Fernandes**  
Professor da Disciplina

---

**Prof. Thatyana Carla Dias Guerra**  
Coordenadora

**JOÃO PESSOA - PB  
2017**

Dedicamos este trabalho a todos aqueles que contribuíram para sua realização; em especial, aos nossos pais, que sempre acreditaram e apoiaram os nossos estudos.

## **AGRADECIMENTOS**

Em primeiro lugar, a Deus, por estar sempre nos guiando pelos caminhos certos e nos dando forças para continuar, mesmo diante dos obstáculos;

Aos nossos pais, por todo o esforço que fizeram e fazem para nos dar a força necessária para seguir a vida acadêmica;

Aos professores Ricardo Roberto de Lima e Pedro de Araújo Amorim Fernandes, pela inestimável ajuda e orientação neste Relatório Técnico;

E aos nossos colegas de curso, pelas contribuições e compartilhamento dos conhecimentos conosco.

## RESUMO

Este trabalho teve como objetivo desenvolver um sistema de comunicação eletrônico para utilização pelas turmas de pós-graduação do Centro Universitário de João Pessoa - UNIPÊ, com o intuito de ser utilizado como repositório para sanar dúvidas referentes às diversas atividades dos alunos atendidos. O sistema (conhecido como ChatBot) facilitará a interação entre os alunos e a coordenação, e tem uma gama de respostas a respeito de preços de mensalidades, descontos, grade curricular, entre outros. Foi desenvolvido através da API Dialogflow, e hospedado em alguns app's de mensagens já existentes no mercado, servindo como assistente virtual para o usuário, podendo ser acessado via texto ou voz. Utiliza Inteligência Artificial e Processamento de Linguagem Natural para interpretação das solicitações do usuário, e Sistema Operacional Android. Foi integrado às aplicações de mensagens através da API REST, utilizando linguagem Python e Banco de Dados NoSQL. Os resultados obtidos foram verificados por meio de análise estatística, de acordo com a resposta dos usuários em um formulário on-line e estatísticas de uma ferramenta disponível em um app de mensagens, e apresentados em gráficos. Na conclusão do trabalho, foram apresentados os pontos positivos e as dificuldades enfrentadas, os objetivos atingidos e expectativas de trabalhos futuros.

**Palavras-chave:** ChatBot. Inteligência Artificial. Processamento de Linguagem Natural.

## **ABSTRACT**

The objective of this work was to develop an electronic communication system for use by the graduate classes of the University Center of João Pessoa - UNIPÊ, with the aim of being used as a repository to solve doubts regarding the different activities of the students attended. The system (known as ChatBot) will facilitate interaction between students and coordination, and has a range of answers regarding prices of tuition, discounts, curriculum, among others. It was developed through the API Dialogflow, and hosted in some messaging app already existing in the market, serving as a virtual assistant for the user, and can be accessed via text or voice. Uses Artificial Intelligence and Natural Language Processing for interpretation of user requests, and Android Operating System. It was integrated into messaging applications through the REST API, using Python language and NoSQL database. The results obtained were verified by means of statistical analysis, according to the response of the users in an online form and statistics of a tool available in a messaging app, and presented in graphs. At the conclusion of the study, the positive points and difficulties faced, the objectives achieved and the expectations of future work were presented.

**Keywords:** ChatBot. Artificial Intelligence. Natural Language Processing.



## LISTA DE FIGURAS

Figura 1 - Arquitetura do Dialogflow _____	25
Figura 2 - Criação da conta no Google _____	26
Figura 3 - Página de acesso à interface _____	26
Figura 4 - Console _____	27
Figura 5 - Criando Novo <i>Agent</i> _____	27
Figura 6 - Definição de <i>Intents</i> _____	28
Figura 7 - Definição de <i>Entities</i> _____	28
Figura 8 - Fluxo de preços _____	29
Figura 9 - Fluxo de matrícula _____	30
Figura 10 - Integrations _____	30
Figura 11 - Configuração de WebDemo _____	31
Figura 12 - Endereço eletrônico da aplicação _____	31
Figura 13 - Integração com o Twitter _____	32
Figura 14 - Criando a aplicação no Twitter _____	32
Figura 15 - Twitter - gerando access token _____	33
Figura 16 - Integração com o Facebook Messenger _____	34
Figura 17 - Facebook Developers _____	34
Figura 18 - Criando novo app no Facebook Messenger _____	35
Figura 19 - Chave Secreta do Aplicativo _____	35
Figura 20 - Configurando Webhooks _____	36
Figura 21 - Tela da aplicação no Facebook _____	36
Figura 22 - Mensagem de saudação _____	37
Figura 23 - Facebook Developers – Revisão de aplicativo _____	37
Figura 24 - Facebook Developers – Adicionar itens _____	38
Figura 25 - Facebook Developers – pages_messaging _____	38
Figura 26 - Facebook Developers – Verificação do aplicativo _____	39
Figura 27 - Facebook Developers – configurações _____	39
Figura 28 - Telegram - gerando access token _____	40
Figura 29 - Telegram → BotFather - iniciando o BotFather _____	40
Figura 30 - Telegram → BotFather - criando o bot para o telegram _____	41
Figura 31 - DialogFlow – webhook _____	41
Figura 32 - Heroku - tela inicial _____	42

Figura 33 - Heroku - personal apps _____	42
Figura 34 - Heroku – overview _____	43
Figura 35 - Python – webhook _____	43
Figura 36 - Postman - tela inicial _____	45
Figura 37 - Firebase – database _____	45
Figura 38 - Firebase – storage _____	46
Figura 39 - Dados referentes ao sexo das pessoas que utilizaram o agente _____	48
Figura 40 - Dados referentes às idades das pessoas que utilizaram o agente _____	48
Figura 41 - Grau de escolaridade das pessoas que utilizaram o agente _____	49
Figura 42 - Renda média das pessoas que utilizaram o agente _____	49
Figura 43 - Área de atuação das pessoas que utilizaram o agente _____	50
Figura 44 - Percentual das pessoas que demonstraram interesse na pós _____	50
Figura 45 - Interesse nos cursos da pós-graduação através do agente _____	51
Figura 46 - Dados referentes à qualidade de atendimento do agente _____	51
Figura 47 - Dados referentes à experiência das pessoas que utilizaram o Robô Virtual ____	52
Figura 48 - Percentual da quantidade de respostas inapropriadas durante a interação Humano – Robô _____	52
Figura 49 - Dados referentes à experiência das pessoas com o agente _____	53
Figura 50 - Dados referentes a opinião das pessoas quanto a uma nova interação com o agente _____	53
Figura 51 - Facebook Analytics - Versão do sistema operacional _____	54
Figura 52 - Facebook Analytics - atividade dos usuários _____	54
Figura 53 – WBS Sistema de ChatBot _____	65

## SUMÁRIO

<b>1 INTRODUÇÃO</b>	<b>12</b>
1.1 CONTEXTO DO PROJETO	12
1.2 JUSTIFICATIVA DO PROJETO	13
1.3 OBJETIVOS	14
1.4 METODOLOGIA UTILIZADA	14
1.4.1 Delineamento da Pesquisa	15
1.4.2 Definição da População/Amostra	15
1.4.3 Técnicas e procedimentos de coleta	15
1.4.4 Técnicas e procedimentos de análise	15
<b>2 TECNOLOGIAS UTILIZADAS PARA IMPLEMENTAÇÃO DE AGENTES INTELIGENTES</b>	<b>17</b>
2.1 CHATTERBOT	17
2.2 INTELIGÊNCIA ARTIFICIAL	18
2.3 INTERFACE DE PROGRAMAÇÃO DE APLICAÇÕES (API)	19
2.4 TECNOLOGIAS UTILIZADAS	21
2.4.1 Python	21
2.4.2 JSON	22
2.4.3 Firebase	23
2.4.4 Heroku	23
<b>3 SOLUÇÃO DO CHATTERBOT</b>	<b>25</b>
3.1 ESTRUTURA DA SOLUÇÃO	25
3.2 PROCESSO DE DESENVOLVIMENTO	26
3.2.1 Intents	28
3.2.2 Entities	28
3.2.3 Fluxos Guiados (BPMN)	29
3.2.4 Estratégias de Integração com redes sociais	30
3.2.5 Estratégias de Integração – backend	41
3.3 SERVIÇOS UTILIZADOS	44
3.3.1 REST	44
3.3.2 Utilização do Banco NOSQL (Firebase)	45
<b>4 RESULTADOS</b>	<b>47</b>
4.1 AVALIAÇÃO DO CHATBOT DESENVOLVIDO	47

<b>4.1.1 Identificação de Perfil</b>	<b>47</b>
<b>4.1.2 Interesse na Pós-Graduação</b>	<b>50</b>
<b>4.1.3 Experiência do Usuário</b>	<b>51</b>
<b>4.1.4 Sugestões e opiniões do usuário</b>	<b>53</b>
<b>4.2 Analysis Chatbot no Messenger</b>	<b>54</b>
<b>5 CONCLUSÃO</b>	<b>56</b>
<b>REFERÊNCIAS</b>	<b>58</b>
<b>APÊNDICES</b>	<b>61</b>
Termo de Abertura do Projeto	61
Declaração de Escopo do Projeto	64
Estrutura de desenvolvimento	70

## INTRODUÇÃO

O objetivo deste capítulo é apresentar o contexto no qual este relatório técnico se insere e a justificativa do projeto, expondo o problema a ser solucionado e propondo o desenvolvimento de uma aplicação que concretize esta solução. O capítulo encerra-se com a explanação da metodologia de estudo utilizada.

### 1.1 CONTEXTO DO PROJETO

A instituição que será contextualizada no presente trabalho é considerada atualmente uma dos melhores Centros Universitários Privados do Nordeste, segundo avaliação do Ministério da Educação e Cultura (MEC)<sup>1</sup>. Possui mais de 50 cursos de pós-graduação disponíveis<sup>2</sup>, em diversas áreas de estudo.

Devido à grande quantidade de cursos, existe uma grande demanda por informações gerais da pós-graduação, e sobre cursos específicos. Os meios disponíveis para a busca da informação, contudo, são incompletos. O site de pós-graduação da instituição é bem elaborado, com informações gerais sobre os cursos, mas sem aprofundamento dos dados disponíveis. Outros meios tradicionais (telefone, atendimento presencial e até mesmo o e-mail, ultimamente) estão caindo em desuso, sobretudo pela nova geração de estudantes, a geração Z, que preferem utilizar app's (abreviatura de applications – aplicativos para celular), redes sociais e aplicativos de mensagens instantâneas para interação (PRENSKY, 2001). Os app's, entretanto, consomem espaço de memória, necessitam de atualizações constantes e diversas versões para sistemas operacionais distintos.

Segundo o Vice-Presidente sênior de pesquisa da Gartner, Peter Sondergaard, “Em 2020, as pessoas não irão usar aplicativos em seus aparelhos. (...) As pessoas vão contar com os assistentes virtuais pra tudo.”<sup>3</sup>

A Gartner é uma das maiores empresas de Tecnologia da Informação do mundo. De acordo com o seu Vice-Presidente, algoritmos serão verdadeiros assistentes virtuais, e se tornarão a única interface de interação dos usuários. Estes algoritmos são um conjunto de regras que nos ajudam a resolver um problema. Alguns

---

<sup>1</sup> <https://www.facebook.com/unipeoficial/>

<sup>2</sup> <http://unipe.br/pos-graduacao/>

<sup>3</sup> Disponível em <http://ofuturodascoisas.com/gartner-em-2020-nao-havera-mais-aplicativos/>.

algoritmos existentes trabalham com Inteligência Artificial, isto é, eles são criados para aprender novas formas de interações por si mesmos; cada interação do usuário serve para o crescimento do algoritmo (DOMINGOS, 2017). Google Now, Siri e Cortana são exemplos práticos.

Considerando a grande quantidade de pessoas que utilizam redes sociais e aplicações de mensagens instantâneas, e com a percepção das dificuldades de interação entre app's e estas aplicações (uso de memória, versões, interfaces distintas), alguns profissionais e empresas de Tecnologia da Informação (TI) criaram um meio de hospedar assistentes virtuais nas aplicações de mensagens já utilizadas pelos usuários. Estes assistentes virtuais são os *ChatBots*.

*ChatBot* é a abreviatura em inglês para *Chatter-Robot* (robô de bate-papo, em uma tradução livre), significando a interação entre homem e máquina através de aplicações/aplicativos de mensagens; um *ChatBot* funciona hospedado em um aplicativo de mensagens, e permite que seus usuários se comuniquem com o software via texto e voz; utiliza Inteligência Artificial e Processamento de Linguagem Natural para interpretação das solicitações do usuário (TEIXEIRA; MENEZES, 2003).

Desta forma, o presente trabalho se propõe a desenvolver um assistente virtual para utilização da coordenação de pós-graduação da instituição de ensino, contribuindo para diminuir a deficiência observada pelos alunos na procura por informações específicas dos cursos existentes.

## 1.2 JUSTIFICATIVA DO PROJETO

Conforme pesquisa da Forrester Research 2015, “84% das pessoas utiliza apenas cinco aplicativos por mês, o que significa que os clientes estão mais exigentes quanto às suas necessidades de comunicação e demanda por serviços.”<sup>4</sup>

O desenvolvimento de um *ChatBot* para a pós-graduação de um Centro Universitário localizado no município de João Pessoa, além de facilitar bastante a comunicação, abre mais um canal de interação da instituição com o seu público (interno e externo), podendo ser uma das principais maneiras da mesma interagir com o seus discentes, respondendo à maioria das demandas em tempo real.

---

<sup>4</sup> Disponível em <https://startupi.com.br/2016/10/chatbots-uma-tendencia-cada-vez-mais-forte-na-tecnologia/>.

Os diversos aplicativos que são utilizados pelos discentes não têm um serviço integrado para contato com a instituição, o que significa que é interessante e atrativo criar um serviço de mensagens que possa ser integrado com o aplicativo favorito deste usuário.

Percebe-se uma crescente exigência por entrega de informação de forma rápida e simplificada, e a falta de tempo faz com que, muitas vezes, alunos ou potenciais alunos não tenham condições de comparecer presencialmente ou telefonar para tirar dúvidas acerca de assuntos relacionados ao curso.

Verifica-se também, em determinados períodos do dia, um grande fluxo de atendimento presencial, ocasionando irritação pela demora no atendimento, desistência de potenciais alunos e perda de conteúdo dos alunos já matriculados.

A criação de um *ChatBot* faz com que grande parte do atendimento se torne totalmente automatizado, fazendo com que apenas assuntos de maior complexidade exijam atendimento humano. Além de ser um repositório das dúvidas mais frequentes sobre os cursos, pode servir de base para um banco de dados de alunos interessados nos diversos cursos da instituição.

### 1.3 OBJETIVOS

Desenvolver um sistema de *ChatBot* para utilização das turmas de pós-graduação de uma Universidade Particular como repositório, para sanar dúvidas referentes a preços de mensalidades, descontos, grade curricular, horários de aulas e outros assuntos correlatos às disciplinas e cursos.

Objetivos Específicos:

- Desenvolver o levantamento dos requisitos para o sistema;
- Identificar ferramentas similares já existentes e suas carências;
- Demonstrar os benefícios do sistema de *ChatBot* para a instituição de ensino e as pessoas que vierem a utilizá-lo;
- Projetar o sistema de *ChatBot*;
- Integrar o sistema a um app de mensagens.

### 1.4 METODOLOGIA UTILIZADA

Esta seção objetiva descrever o tipo de pesquisa utilizado neste trabalho de conclusão de curso, informando fontes e instrumentos utilizados, e procedimentos seguidos para o atingimento dos objetivos específicos e do objetivo geral.

Foi utilizada como referência a obra de Gerhardt e Silveira – Métodos de Pesquisa (GERHARDT; SILVEIRA 2009) para o desenvolvimento da metodologia da pesquisa.

#### **1.4.1 Delineamento da pesquisa**

Quanto aos objetivos, foi utilizado o método exploratório, através de entrevistas com as partes interessadas, levantamento bibliográfico e documental e testes, entre outras técnicas.

Em relação à abordagem, utilizou-se a análise qualitativa, nas entrevistas iniciais com a equipe da instituição responsável pela utilização da aplicação; e a quantitativa, nos testes do *ChatBot* e nos questionários de interação com a ferramenta, a fim de aferir o grau de precisão das respostas da interface.

#### **1.4.2 Definição da população/amostra**

A amostragem não probabilística é onde existe uma escolha deliberada dos elementos que irão formar a amostra. Foi utilizada esta modalidade de amostragem por questões de conveniência, através de convites a alunos selecionados, que testaram a aplicação e responderam questionário avaliativo da mesma.

#### **1.4.3 Técnicas e procedimentos de coleta**

Nas entrevistas iniciais, com o intuito de elaborar um fluxo de trabalho da equipe de coordenação da pós-graduação, utilizamos a observação, para montagem do referido fluxo de trabalho; as entrevistas foram presenciais e, posteriormente, através de consultas por meios eletrônicos (e-mail, whatsapp etc).

Nos testes da aplicação, o método utilizado foi o de disponibilização da interface, com seleção dos usuários, que fizeram pesquisas sobre os cursos de pós-graduação através da aplicação, sem interveniência direta dos responsáveis pela criação do *ChatBot*.



#### **1.4.4 Técnicas e procedimentos de análise**

Para a pesquisa qualitativa, os dados coletados foram analisados e classificados de forma a detectar as interações mais frequentes dos alunos com a coordenação de pós-graduação, e o passo-a-passo desde o primeiro contato até a resolução/resposta à demanda destes. Após esta verificação, foram propostos modelos de fluxo de trabalho a serem utilizados pela instituição na aplicação.

Na pesquisa quantitativa, aplicou-se a análise estatística, com representações gráficas dos resultados e discussão dos resultados.

## 2 TECNOLOGIAS UTILIZADAS PARA IMPLEMENTAÇÃO DE AGENTES INTELIGENTES

Este capítulo apresenta uma revisão bibliográfica dos estudos existentes referentes à área de atuação dos *ChatBots*, algumas aplicações existentes e definições dos termos utilizados ao longo deste Trabalho de Conclusão de Curso.

### 2.1 CHATTERBOT

De acordo com a definição clássica, *chatterbots* (ou *chatbots*) são sistemas de computação programados para interagir com humanos através da Linguagem Natural, simulando uma conversação (LAVEN, 2017). As respostas são previamente elaboradas em um banco de dados, e são apresentadas após as solicitações dos usuários da aplicação.

A primeira ideia relativa à criação de um sistema de computação que pudesse interagir com humanos como se fosse um humano foi desenvolvida em 1950. Em seu artigo “*Computing machinery and intelligence*”, A. M. Turing propôs um experimento que consistia em colocar dois entrevistados (**A** e **B**) de sexos distintos em um cômodo, e o entrevistador (**C**) em outro; através do diálogo, o entrevistador deveria descobrir o sexo de **A** e o de **B**; durante o experimento, sem o conhecimento do entrevistador, **A** seria substituído por uma máquina, que simularia um humano na conversação (TURING, 1950). O teste procuraria responder a seguinte questão: “Será que o interrogador decidirá erroneamente com a mesma frequência, quando o jogo é jogado dessa forma, do que quando o fazia ao tempo em que o jogo era jogado entre um homem e uma mulher?”

O primeiro *Chatterbot* a ser criado foi o *ELIZA*, em 1965, no *Massachusetts Institute of Technology (MIT)*, e seu objetivo era simular diálogos entre pacientes e um psicólogo, utilizando as frases do paciente para elaborar novas perguntas (WEIZENBAUM, 1966). Era um algoritmo simples, com apenas 204 linhas de código fonte. *ELIZA* analisava frases e fragmentos delas, localizando palavras-chave previamente determinadas, e montava sentenças a partir destas palavras-chave.

Desde então, até meados da década de 1990 inúmeros *Chatbots* foram criados, boa parte acadêmicos, porém sua evolução foi marcada pelas respostas limitadas, incapacidade para reconhecer contextos das frases e dificuldades nas variações na formulação de perguntas, causando certa frustração dos usuários (ABREU, 2017).

A grande mudança começou com o início da utilização de Inteligência Artificial (IA) nos algoritmos. Em 1994, Michael Mauldin criou a **Julia**, uma personagem de um jogo virtual, que auxiliava os usuários, com mapeamento dos cenários e mensagens de texto (MAULDIN, 1994). Quando questionada, ela podia fornecer o caminho mais curto do usuário até determinado local. Não obstante a similaridade com os *Chatbots* anteriores, ela era capaz de lembrar das ações que havia tomado anteriormente, além de informações sobre outros jogadores, que auxiliavam na tomada de decisões. O *Chatbot* também tinha variações de humor, de acordo com o dia no mês, simulando reações femininas. O termo *Chatbot* foi criado a partir da criação de Mauldin.

Um grande sucesso da década foi o A.L.I.C.E. (*Artificial Linguistic Internet Computer Entity*), com uma grande base de conhecimento, mais de 5.000 palavras em seu vocabulário e módulos de conversação que permitem ao *Chatbot* classificar o usuário pela sua profissão, sexo, idade ou localização geográfica (WALLACE, 1995). Possuía interface gráfica e uma saudação sonora, estimulando o diálogo.

Porém, o grande salto de popularidade se deu com a entrada das gigantes da tecnologia *Google*, *Facebook*, *Apple* e *Microsoft* na área de inteligência artificial, por meio de técnicas de *Machine Learnig* (aprendizado de máquina), reconhecimento de voz e processamento de linguagem natural, que criaram assistentes virtuais como *Cortana*<sup>5</sup>, *Siri*<sup>6</sup> e *Google Assistant*<sup>7</sup>.

## 2.2 INTELIGÊNCIA ARTIFICIAL

De acordo com Coppin (2010, p. 25), “Inteligência Artificial envolve utilizar métodos baseados no comportamento inteligente de humanos e outros animais para solucionar problemas complexos”. Dispositivos (máquinas, robôs, softwares) são programados para simular a capacidade humana, e de outros animais, de raciocinar, perceber, tomar decisões, resolver problemas.

O termo “*Artificial Intelligence*” (Inteligência Artificial, em português) foi utilizado pela primeira vez em 1956, em uma conferência de especialistas celebrada no *Dartmouth College* (faculdade norte-americana), pelo cientista da computação John McCarthy. Segundo McCarthy (1956), a IA (Inteligência Artificial) é a ciência e engenharia utilizadas e necessárias para produzir máquinas inteligentes. A busca pela construção de “máquinas

---

<sup>5</sup> Endereço Microsoft Cortana: <https://www.microsoft.com/pt-br/windows/cortana>

<sup>6</sup> Endereço Apple Siri: <https://www.apple.com/br/ios/siri/>

<sup>7</sup> Endereço Google Assistant: <https://assistant.google.com/>

pensantes”, no entanto, é bem antiga, remontando à Grécia antiga, com a construção de autômatos, dispositivos mecânicos que tentavam reproduzir imagens móveis da vida (HADOT, 2006).

A partir do ano de 1950, com o lançamento do artigo “*Computing machinery and intelligence*”, de A. M. Turing, que discutiu a possibilidade de uma máquina se passar por humano sem percebermos, inicia-se a era da Inteligência Artificial moderna (TURING, 1950). Diversos dispositivos passam a ser criados com softwares integrados, que se tornaram conhecidos como **Agentes**; estes agentes realizam alguma tarefa em nome de um humano de forma autônoma.

Segundo Russel e Norvig (2004), um agente é aquele que percebe o seu ambiente por meio de sensores e age sobre ele através dos atuadores. Os **Agentes Inteligentes** mais desenvolvidos têm conhecimento adicional de domínio que os habilita a realizar as tarefas deles, mesmo quando os parâmetros da tarefa mudam ou quando surgem situações inesperadas (COPPIN, 2010).

Existem dois tipos de agentes: Os baseados em regras e os desenvolvidos com inteligência artificial; os agentes baseados em regras são aqueles que só respondem a perguntas pré-definidas, ou alguns comandos de voz, especificados na sua interface de programação, a exemplo de centrais de telemarketing, que estabelecem opções pré-definidas para a escolha do usuário; estes agentes não têm capacidade de interagir com o usuário.

Os agentes inteligentes, por sua vez, são dotados de inteligência artificial, e aprendem a partir do seu próprio desempenho, do usuário e do ambiente onde estão situados (COPPIN, 2010); muitos agentes inteligentes têm autonomia e capacidade de cooperação com outros agentes. Estes tipos de agentes são capazes de interagir com o usuário, mesmo não tendo sido programados para responder à solicitação deste: Eles procuram achar uma resposta na sua base de conhecimento<sup>8</sup>.

### 2.3 INTERFACE DE PROGRAMAÇÃO DE APLICAÇÕES (API)

API é um conjunto de rotinas e padrões de programação para acesso a um aplicativo de software ou plataforma baseado na Web<sup>9</sup>. API's possibilitam uma interface de comunicação entre aplicações distintas, como por exemplo fazer login em um site utilizando

---

<sup>8</sup>Medium - O que é um Chatbot: <https://medium.com/botsbrasil/o-que-%C3%A9-um-chatbot-7fa2897eac5d>.

<sup>9</sup>Canaltech - O que é API: <https://canaltech.com.br/software/o-que-e-api/>.

dados de sua rede social. A sigla é derivada do nome em inglês “*Application Programming Interface*”, que significa, em português, “*Interface de Programação de Aplicações*”.

Existem milhares de API's disponíveis atualmente, grande parte destas públicas, podendo ser utilizadas por desenvolvedores de aplicações livremente<sup>10</sup>. Grandes empresas possuem API's que são utilizados nas mais diversas áreas, como busca de preços em tempo real (decolar.com), pontuação em programas de milhagem (Shoptime) e até mesmo lojas de fornecimento de API's (IBM, com o Watson Bluemix, por exemplo).

Desenvolvida pela Google, a API *Dialogflow*<sup>11</sup> é uma plataforma para o desenvolvimento de agentes conversacionais de voz e texto, disponibilizada de forma gratuita. A *Dialogflow* utiliza processamento de linguagem natural (PLN) para o desenvolvimento dos agentes inteligentes, o que significa que o usuário pode fazer requisições através da fala. Sua arquitetura é formada basicamente por *Intenções*, *Entidades*, *Contextos*, *Agentes* e *Parâmetros*. Sua plataforma permite uma fácil integração com redes sociais, tais como: Facebook Messenger, Twitter, Slack, Telegram, Skype, além de ser integrada com os Assistentes do Google.

A plataforma de interface conversacional do Google *Dialogflow* permite um grande alcance de público através dos seus mais de 14 idiomas suportados, entre eles o espanhol, francês e japonês. Algumas empresas começaram a aderir a tecnologia de agentes conversacionais do *Dialogflow*, tais como: Giorgio Armani, The Wall Street Journal, Royal Dutch Airlines.

A plataforma do *Dialogflow* é composta por:

a) *Intenções (Intents)*, que são um mapeamento sobre o que o usuário diz ou sobre o que o usuário digita e a ação a ser tomada pelo software;

b) *Entidades (Entities)*, que são ferramentas poderosas usadas para extrair valores de parâmetros de entradas de linguagem natural; quando usadas em um agente específico, dependerão dos valores dos parâmetros que devem ser retornados como resultado do funcionamento do agente;

c) *Contexto (Context)*, que são *strings* (espécies de palavras-chave) que definem o contexto da conversa, observando que uma mesma *string* pode representar ações diversas, a depender do contexto do momento da interação. São úteis para diferenciar frases que são vagas ou têm significados diferentes. São definidos em dois tipos: Contextos de Entrada e Contextos de Saída. Contextos de Entrada são intenções limitadas a um determinado período,

<sup>10</sup> Vertigo - <http://blog.vertigo.com.br/entenda-o-que-e-uma-api>.

<sup>11</sup> Página da *Dialogflow*: <https://dialogflow.com/>.

por padrão 10 minutos ou 5 requisições. Contextos de Saída, quando atribuídos a uma Intenção, especificam uma ação a adotar naquela requisição;

d) Agentes (*Agents*), módulos de PLN, encarregados de interpretar os comandos e gerar as respostas; e Parâmetros (*Parameters*), utilizados para conectar as palavras faladas ou digitadas pelo usuário a entidades.

## 2.4 TECNOLOGIAS UTILIZADAS

Esta seção apresenta e define as aplicações e linguagens de programação utilizadas para o desenvolvimento do aplicativo objeto deste trabalho.

### 2.4.1 Python

Segundo Borges (2010, p.10), "Python é uma linguagem de altíssimo nível (em inglês, *Very High Level Language*) orientada a objeto, de tipagem dinâmica e forte, interpretada e interativa".

A linguagem foi criada em 1990 por Guido Van Rossum, no Instituto Nacional de Pesquisa para Matemática e Ciência da Computação da Holanda (CWI) e tinha originalmente foco em usuários como físicos e engenheiros. O Python foi concebido a partir de outra linguagem existente na época, chamada ABC (BORGES, 2010, p.10).

Hoje, a linguagem é bem aceita na indústria por empresas de alta tecnologia (BORGES, 2010, p.11), tais como:

- Google (aplicações Web).
- Yahoo (aplicações Web).
- Microsoft (Iron Python: Python para .NET).
- Nokia (disponível para as linhas recentes de celulares e PDAs).
- Disney (animações 3D).

O Python inclui diversas estruturas de alto nível (listas, dicionários, data/hora, complexos e outras) e uma vasta coleção de módulos prontos para uso, além de frameworks de terceiros que podem ser adicionados. Também possui recursos encontrados em outras linguagens modernas, tais como: geradores, introspecção, persistência, metaclasses e unidades de teste. Multiparadigma, a linguagem suporta programação modular e funcional, além da orientação a objetos. Mesmo os tipos básicos no Python são objetos (COELHO, 2007).

A linguagem é interpretada através de bytecode pela máquina virtual Python, tornando o código portátil. Com isso é possível compilar aplicações em uma plataforma e rodar em outros sistemas ou executar direto do código fonte (BORGES, 2010, p. 10).<sup>12</sup>

#### 2.4.2 JSON

JSON é um formato para intercâmbio de dados. Um formato para intercâmbio de dados é um formato-texto usado para trocar dados entre plataformas bem diferentes. Apesar de muito simples, tem sido bastante utilizado por aplicações Web devido a sua capacidade de estruturar informações de uma forma bem mais compacta do que a conseguida pelo modelo XML, tornando mais rápido a troca dessas informações<sup>13</sup>. Isto explica o fato de o JSON ter sido adotado por empresas como Google e Yahoo, cujas aplicações precisam transmitir grandes volumes de dados.

Pode-se exemplificar a função do JSON ao se imaginar um mundo composto por várias pequenas ilhas, onde cada ilha fala um idioma diferente, todas elas são dependentes do comércio exterior, e todas têm comerciantes marítimos que viajam longas distâncias entre as ilhas. Tendo em mente que cada ilha fala uma língua diferente, e que seus comerciantes não falam uma língua comum entre eles, as ilhas teriam de investir em pesquisadores para aprender diversos idiomas, e isso consumiria muito tempo. No entanto, todas as ilhas concordaram em definir uma única linguagem para lidarem com negócios, e cada ilha emprega apenas um tradutor, que entende o formato dos dados e relatórios comerciais trazidos pelos comerciantes (BASSETT, 2015).

O mundo da tecnologia é muito semelhante ao mundo das ilhas: há muitas linguagens, sistemas e arquiteturas diferentes e, caso esses sistemas não entrassem em um acordo sobre o formato de transmissão de seus dados, as comunicações estariam prejudicadas, pois consumiram uma quantidade absurda de tempo e recurso (BASSETT, 2015).

---

<sup>12</sup> Página oficial: <http://www.python.org>

<sup>13</sup> Disponível em: <<https://www.profissionaisiti.com.br/2012/03/breve-introducao-ao-json-json-e-java/>>. Acesso em: 11 out. 2017.

### 2.4.3 Firebase

Para o desenvolvimento deste projeto foi escolhido o banco de dados Firebase, que faz parte do grupo de banco de dados não relacionais (*NoSQL*). São bancos projetados para trabalhar com escalabilidade e suportar aplicações em nuvem (GONÇALVES, 2016).

Projetado pelo Google, o Firebase é uma plataforma em nuvem para armazenamento de dados, ofertado de maneira gratuita. O Firebase possui três tipos de planos: grátis, Flame, e Blaze, que funciona mediante pagamento pela utilização. A plataforma possui uma capacidade para assegurar a alta demanda dos usuários, dimensionando automaticamente sua aplicação. Além de ser um banco de dados *NoSQL*, que permite sua arquitetura seja armazenada em objetos *JSON*, diferente do banco de dados *SQL* não utiliza tabelas nem registros (FIREBASE, 2017). A plataforma, que foi fundada em 2011 por Andrew Lee e James Tamplin, possibilita que sua aplicação receba uma sincronia de dados instantaneamente com seus clientes, através da sua SDK, que permite a criação de apps para iOS, Android e JavaScript; criado o app, seus consumidores partilham uma instância para receber progresso de seus dados atuais.<sup>14</sup>

Firebase mantém seus dados em forma de nós, que permite visualizar uma árvore que contém várias ramificações. A aplicação possibilita que seus dados sejam representados em objetos *JSON*, facilitando um melhor entendimento e compreensão dos dados.

### 2.4.4 Heroku

O Heroku se enquadra na categoria de serviços da computação em nuvem conhecida como Plataforma como Serviço (Platform as a Service, ou PaaS), no qual o fornecedor entrega para o cliente um ambiente pronto para receber a aplicação. Diferente do IaaS (Infraestrutura como Serviço), no qual cliente contrata máquinas (reais ou virtuais) e é responsável pela instalação de bibliotecas, montagem das estruturas do sistema de arquivos, entre outros recursos, o PaaS é uma solução de alto nível que abstrai este tipo de preocupação<sup>15</sup>.

Foi criado em 2007 por três desenvolvedores norte-americanos: James Lindenbaum, Adam Wiggins e Orion Henry. O Heroku inicialmente tinha suporte para aplicativos

---

<sup>14</sup> Página oficial Firease: <https://firebase.google.com>

<sup>15</sup> Extraído de: <<https://www.devmedia.com.br/primeiros-passos-em-paas-com-heroku/29465>>. Acesso em 12 out. 2017.



desenvolvidos em Ruby, que é uma linguagem de programação orientada a objetos, rodando no servidor web Rack, uma abstração de requisições e respostas HTTP (MOLINARI, 2016). Em 2010, o projeto foi adquirido pela Salesforce, e em 2011 iniciou um processo para suportar outras linguagens e frameworks. Atualmente o Heroku suporta Ruby, Java, Clojure, Python, Scala e Node, e possui um parque de mais de três milhões de aplicações instaladas.<sup>16</sup>

---

<sup>16</sup> Página oficial Heroku: <https://www.heroku.com>

### 3 SOLUÇÃO DO CHATTERBOT

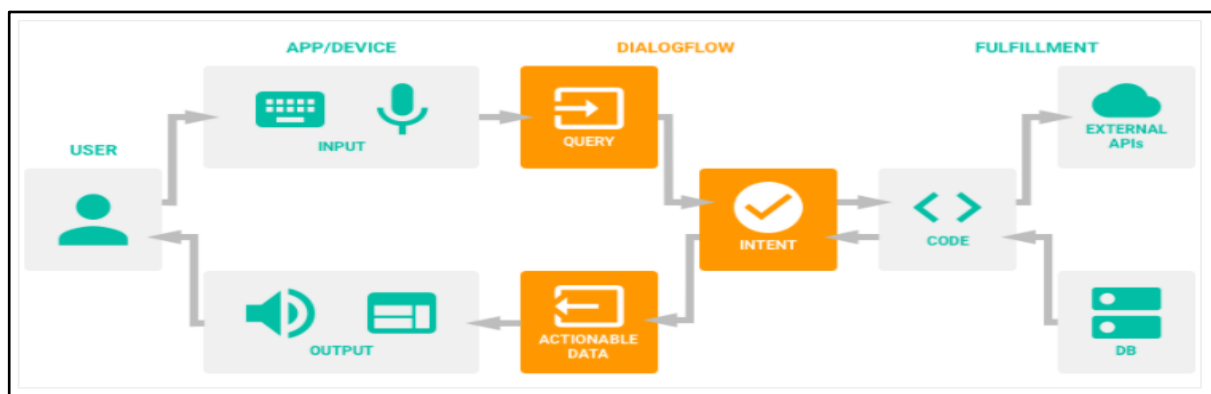
Este capítulo demonstra o processo de desenvolvimento da aplicação, a estrutura desta, estratégias de integração com redes sociais e serviços utilizados.

#### 3.1 ESTRUTURA DA SOLUÇÃO

A arquitetura deste agente inteligente é baseada em Processamento de Linguagem Natural (PLN). Segundo Gariba *et al.* (2005), o processamento de Linguagem Natural vem a facilitar a interação do software (através de sua interface) com o usuário, para que se torne mais fácil a comunicação e a passagem de conhecimento, de forma que quem fizer o uso de um software possa compreender o que ele tem a oferecer e consiga saber o que o usuário está necessitando.

Esta plataforma é fragmentada em três componentes básicos: *Speech Recogiton* (Reconhecimento de Voz), *Natural Language Understanding-NLU* (Compreensão da Linguagem Natural) e *Conversation Management-CM, User-Fulfillment* (Gerenciamento de Conversação) (Figura 1). O NLU possui funcionalidades parecidas ao PLN, só que sua utilidade transforma as requisições dos usuários deixando os dados mais acionáveis.<sup>17</sup>

Figura 1 - Arquitetura do Dialogflow



Fonte: Página oficial da *Dialogflow*, 2017.<sup>18</sup>

O processo de transformar Processamento de Linguagem Natural (PLN) para deixar os agentes mais acionáveis denomina-se NLU. PLN é uma subárea de Inteligência Artificial

<sup>17</sup> Endereço da plataforma: <https://dialogflow.com/>

<sup>18</sup> <https://dialogflow.com/docs/agents>

(IA), que tem como objetivo prover aos agentes a capacidade de entender, compreender e interpretar um texto.

### 3.2 PROCESSO DE DESENVOLVIMENTO

O processo de desenvolvimento do *Chatterbot* inicia-se com a criação da conta do usuário no Google, na página de login da empresa (Figura 2).

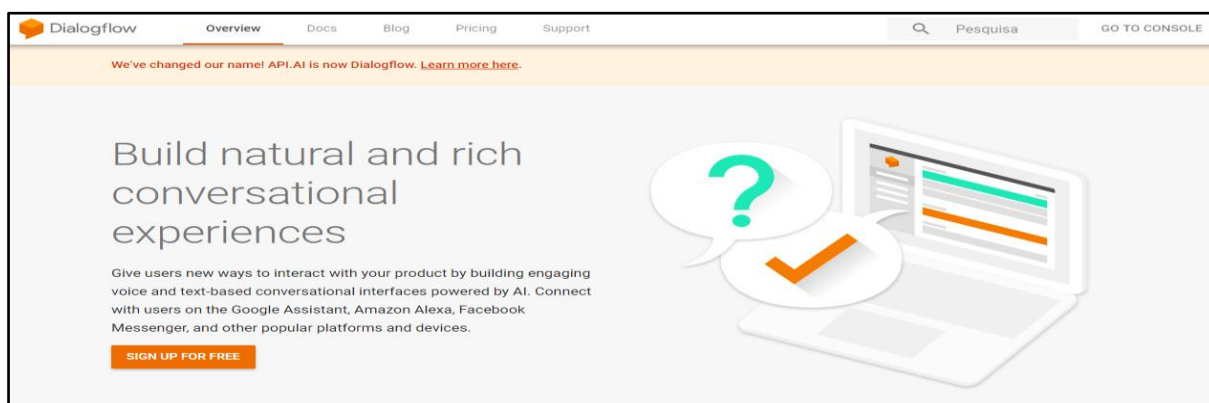
Figura 2 - Criação da conta no Google



Fonte: Página oficial do Google, 2017.<sup>19</sup>

Após a criação da conta, deve-se acessar a página da API do Google; a interface é acessada clicando em “GO TO CONSOLE” no canto superior direito da página (Figura 3).<sup>20</sup>

Figura 3- Página de acesso à interface



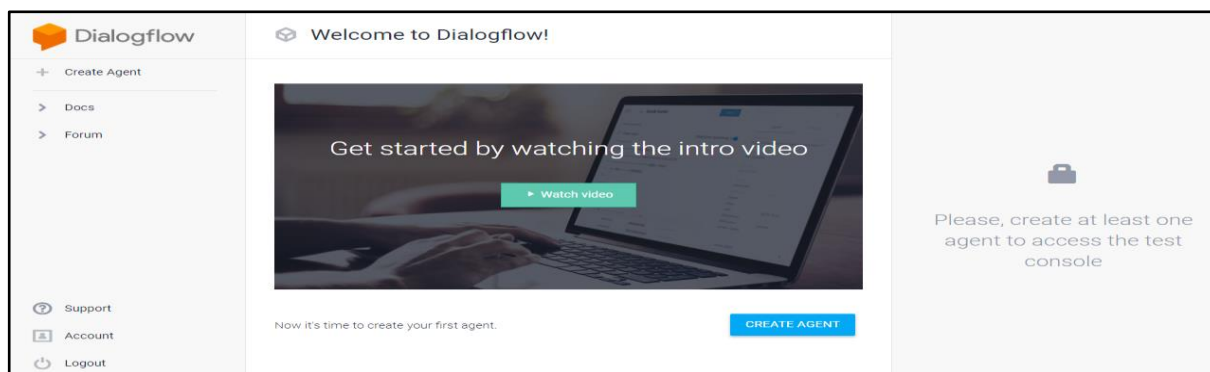
Fonte: Página oficial da *Dialogflow*, 2017.

<sup>19</sup> <https://accounts.google.com/SignUp?hl=pt-BR>

<sup>20</sup> Endereço da API do Google: <https://dialogflow.com/>

Ao acessar a página, deve-se logar com a conta do Google. Após autenticar, o console é disponibilizado para o usuário (Figura 4).

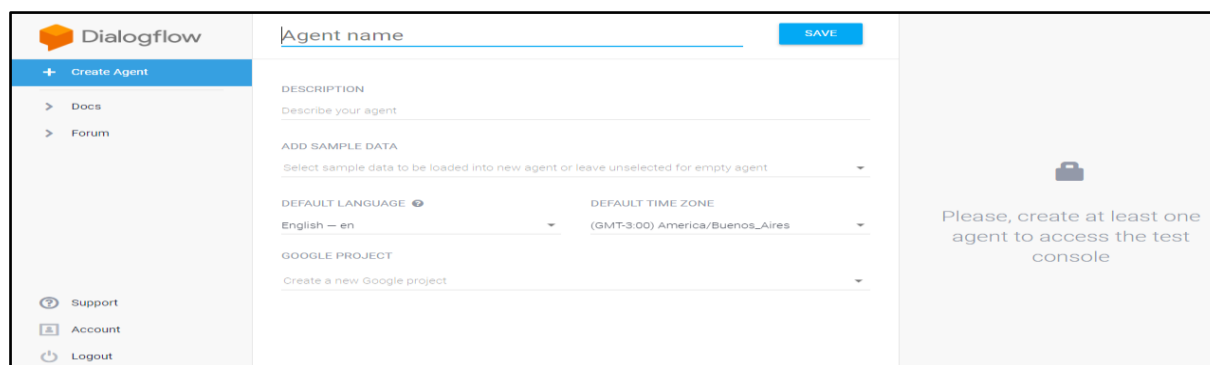
Figura 4 - Console



Fonte: Página oficial da *Dialogflow*, 2017.<sup>21</sup>

Feito isso, deve-se clicar o botão “*create new agent*” para começar o processo de criação do *bot* (abreviatura de *chatbot*); nesta tela, define-se o nome do *bot*, sua descrição e a linguagem padrão (Figura 5).

Figura 5 - Criando Novo Agent



Fonte: Página oficial da *Dialogflow*, 2017.<sup>22</sup>

Nesta etapa, conforme descrito acima, o desenvolvedor do agente irá informar o nome de seu agente, descrição que trata sobre a finalidade do seu agente, idioma em qual vai atuar, fuso horário do país selecionado. Na opção em inglês, a plataforma disponibiliza uma funcionalidade *Add Sample Data* (Adicionar dados de amostra), que são um conjunto com várias Intents alimentadas para o tipo específico do agente que vier a escolher.

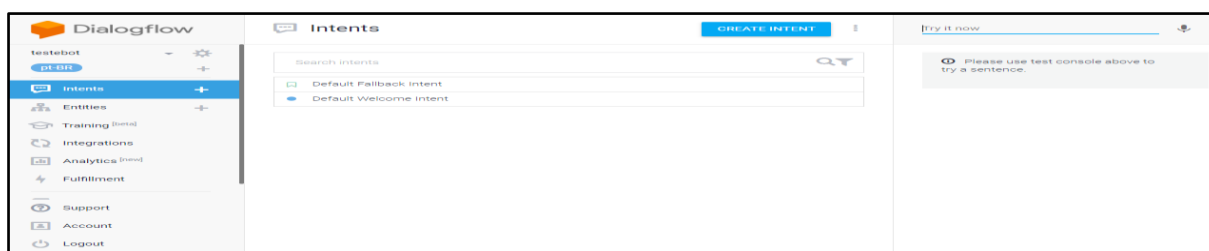
<sup>21</sup> <https://console.dialogflow.com/api-client/#/getStarted>.

<sup>22</sup> <https://console.dialogflow.com/api-client/#/newAgent>.

### 3.2.1 Intents

Para definir o mapeamento do agente é preciso criar Intenções (*Intents*); para isto, deve-se pressionar o botão “*Intents*” no menu à esquerda (vide Figura 6).

Figura 6 - Definição de *Intents*



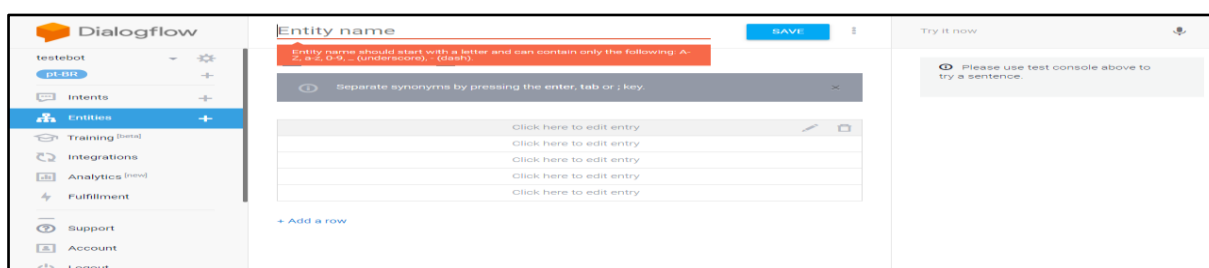
Fonte: Página oficial da *Dialogflow*, 2017.<sup>23</sup>

Neste ponto o desenvolvedor vai definir o nome da *Intent* e posteriormente começar a alimentar seu agente; este processo de aprendizado do agente é através de *Machine Learning* (Aprendizado de Máquina). Esta etapa denomina-se **curadoria** do agente inteligente, na qual alguém fica responsável pelo processo de alimentação do *Bot*. Uma vez que a *Intent* é criada, pode-se alterá-la a qualquer tempo para criar outras expressões.

### 3.2.2 Entities

Para criação de Entidades (*Entities*), o desenvolvedor do agente deve selecionar o botão “*Entities*” no menu à esquerda e, na tela do console, definir o nome da *Entitie*. A opção “*Define synonyms*” deve permanecer marcada como padrão; posteriormente, deve-se adicionar o valor de referência e seus sinônimos (vide Figura 7).

Figura 7 - Definição de *Entities*



Fonte: Página oficial da *Dialogflow*, 2017.<sup>24</sup>

<sup>23</sup> <https://console.dialogflow.com/api-client/#/agent/abd3ab6d-5867-4a34-9d0f-0e7e61549a00/intents>.

Existem três tipos de Entidades: *Developer Entities* (criadas por desenvolvedores), *User Entities* (definida para um utilizador específico) e *System Entities* (pré-criadas pela api). O desenvolvimento da aplicação que é alvo deste estudo será feito utilizando *Developer Entities*.

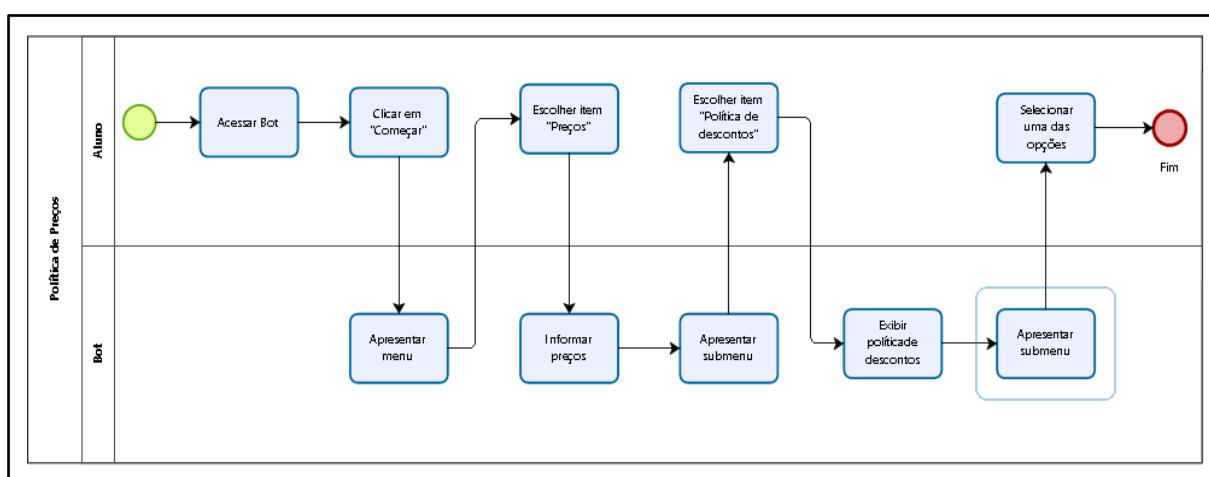
### 3.2.3 Fluxos Guiados (BPMN)

O *Business Process Model and Notation* (BPMN) é uma notação utilizada na metodologia de gerenciamento de processos de negócio; sua grande força deve-se à simplicidade e expressividade, que facilitam o entendimento do usuário. Utilizada para modelar os processos de negócio da organização, ela descreve os fluxos de negócios atuais e propõe alterações para otimizar os procedimentos (CAMPOS, 2014).

Fluxo de Negócios da Pós-Graduação da Instituição de Ensino Estudada: Atualmente, o aluno deve se informar a respeito de detalhes através de atendimento presencial e no site da pós-graduação, que tem muitas informações; não existe, entretanto, um banco de dados com informações consolidadas destes alunos que fazem consultas, sendo basicamente um atendimento reativo (a pós-graduação espera o aluno procurar seus serviços).

Propostas de fluxos de interação com o *Bot*: A Figura 8 apresenta a proposta de fluxo para a consulta de preços pelos usuários, com as entradas e saídas de informações.

Figura 8 - Fluxo de preços

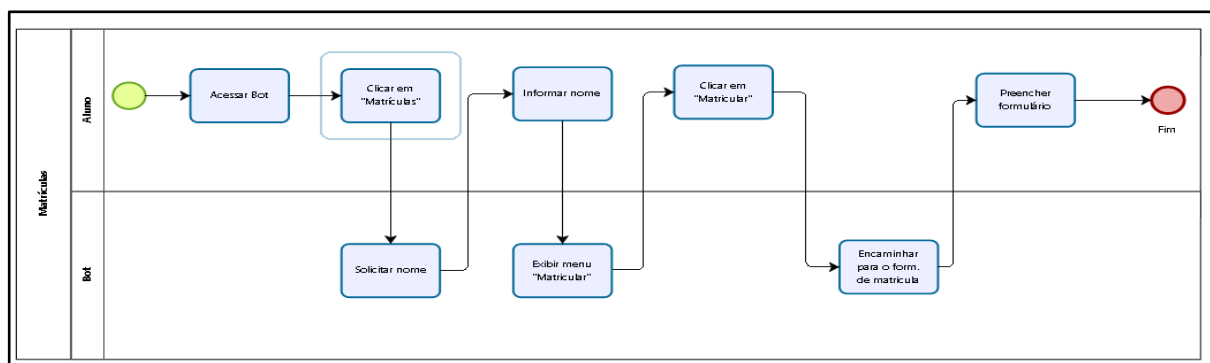


Fonte: Elaborado pelos Autores, 2017.

<sup>24</sup> <https://console.dialogflow.com/api-client/#/agent/abd3ab6d-5867-4a34-9d0f-0e7e61549a00/entities>

A Figura 9 apresenta proposta de fluxo para guiar o usuário para a matrícula no curso escolhido.

Figura 9 - Fluxo de matrícula



Fonte: Elaborado pelos Autores, 2017.

### 3.2.4 Estratégias de Integração com redes sociais

Para o desenvolvimento do agente que é utilizado neste trabalho, optou-se pela integração em redes sociais que têm uma grande demanda de usuários. Desta forma, o agente da Pós-Graduação da Instituição de Ensino está nas seguintes redes sociais: Facebook Messenger, Twitter e Telegram (Figura 10). Foi criado também um endereço eletrônico para a aplicação<sup>25</sup>.

Figura 10 - Integrations



Fonte: Página oficial da Dialogflow, 2017.<sup>26</sup>

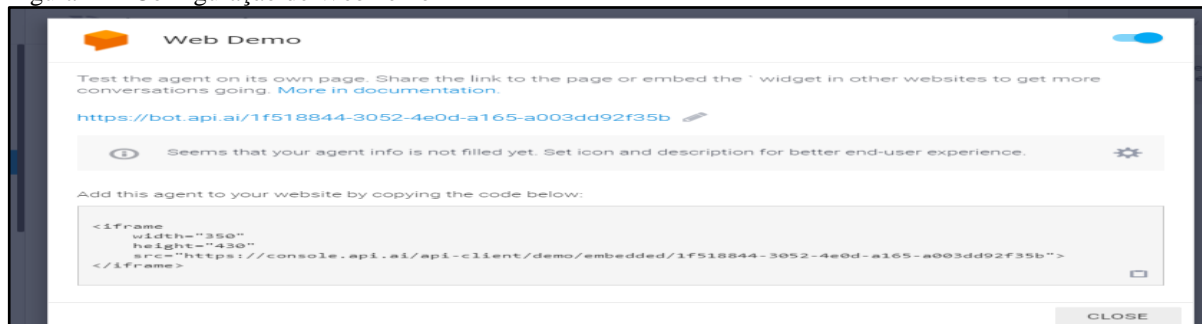
**WEB DEMO** - A opção Web Demo consiste em criar um código html para incluir no site da aplicação; para tanto, o desenvolvedor deve clicar no menu *Integrations*, à esquerda

<sup>25</sup> Endereço eletrônico da aplicação: <https://falcaosport9.wixsite.com/chatbotdireitounipe>.

<sup>26</sup> <https://console.dialogflow.com/api-client/#/agent/abd3ab6d-5867-4a34-9d0f-0e7e61549a00/integrations>.

(na plataforma *DialogFlow*), e selecionar a opção WebDemo. Feito isso, deve copiar o código html e colocar em seu site (Figura 11).

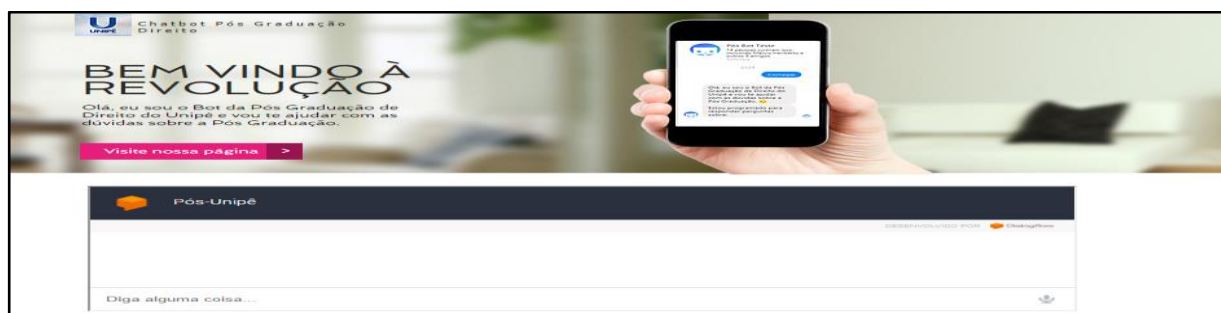
Figura 11 - Configuração de WebDemo



Fonte: Página oficial da *Dialogflow*, 2017.<sup>27</sup>

Isso é realizado através de um Iframe, que permite colocar arquivos em páginas web. Para este trabalho foi utilizada a opção de Iframe na página (Figura 12).

Figura 12 - Endereço eletrônico da aplicação



Fonte: Página do bot no Wix, 2017.<sup>28</sup>

Foi utilizada a plataforma de sites gratuitos **Wix.com** (Wix.com) para o desenvolvimento desta página Web. Esta plataforma também oferece planos pagos; alguns de seus planos são: Vip, eCommerce, Unlimited, Combo, Connect Domain.

**TWITTER** - Outra opção utilizada para integração em redes sociais foi o Twitter, aplicação de mensagens curtas fundada em 21 de Março de 2006, em *San Francisco*, Califórnia, EUA. Conta com aproximadamente 328 milhões de usuário ativos mensais<sup>29</sup>.

<sup>27</sup> <https://console.dialogflow.com/api-client/#/agent/abd3ab6d-5867-4a34-9d0f-0e7e61549a00/integrations>.

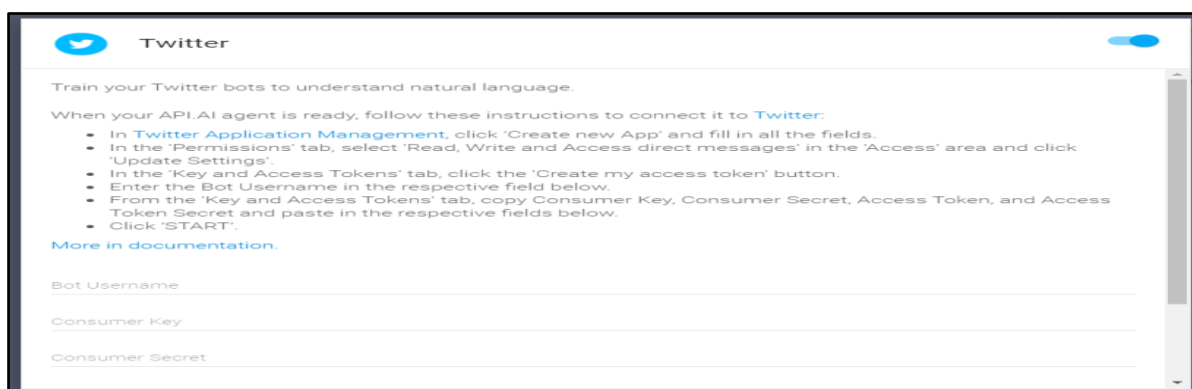
<sup>28</sup> <https://falcaosport9.wixsite.com/chatbotdiretounipe>.

<sup>29</sup> Endereço oficial do Twitter: twitter.com.



Para realizar a integração com a plataforma *Dialogflow*, o desenvolvedor deve, primeiramente, selecionar no menu à esquerda a opção *Integrations* e clicar no ícone do twitter; a seguir, deve seguir os passos descritos no menu (Figura 13).

Figura 13 - Integração com o Twitter



Fonte: Página oficial da *Dialogflow*, 2017.<sup>30</sup>

O desenvolvedor do agente deve, então, acessar o endereço <https://apps.twitter.com/>, e selecionar o botão *create new app*, exibido no console; será exibida, então, uma tela para informar o nome da aplicação, a descrição e o site do projeto; por último, deve-se marcar a opção “*Yes, I have read and agree to the Twitter Developer Agreement*”, e clicar no botão “*Create your Twitter Application*”(Figura 14).

Figura 14 - Criando a aplicação no Twitter

Fonte: Página oficial do Twitter, 2017.<sup>31</sup>

<sup>30</sup> <https://console.dialogflow.com/api-client/#/agent/abd3ab6d-5867-4a34-9d0f-0e7e61549a00/integrations>

<sup>31</sup> <https://apps.twitter.com/app/new>

Continuando o passo a passo, deve-se fazer algumas alterações; a primeira é modificar as permissões do aplicativo e selecionar a opção “*Read, Write and Access direct messages*”; posteriormente, selecionar o botão “*Update Settings*”.

O próximo passo é gerar o token da aplicação; deve-se selecionar a aba “*Keys and Access tokens*”, e clicar no final da tela, no botão “*Create my access token*”. Feito isso, serão geradas as chaves que devem ser colocadas na plataforma *Dialogflow*. Ao todo o desenvolvedor deve copiar quatro chaves e colocar na plataforma *Dialogflow*; são elas: Consumer Key (API KEY), Consumer Secret (API SECRET), Access Token, Access Token Secret (Vide Figura 15).

Figura 15 - Twitter - gerando access token



Fonte: Página oficial do Twitter, 2017.<sup>32</sup>

Copiadas todas as chaves e colocadas na plataforma *Dialogflow*, o desenvolvedor deve clicar no botão “Start” para iniciar o serviço; a configuração estará concluída, e o *bot* já estará apto a ler as mensagens recebidas pelo usuário.

FACEBOOK - Fundado em 4 de fevereiro de 2004, é a maior rede social do mundo com mais de 2 bilhões de usuários<sup>33</sup>. O público-alvo do agente é voltado para essa rede social, por isso este agente possui muitas funcionalidades exclusivas para o Facebook.

Mesmo possuindo sua página no Facebook, o recebimento de mensagens é através da plataforma Messenger, que possui 1,2 bilhões de usuário no mundo<sup>34</sup>.

<sup>32</sup> <https://apps.twitter.com/app/accesstoken>.

<sup>33</sup> Disponível em [https://www.facebook.com/pg/facebook/about/?ref=page\\_internal](https://www.facebook.com/pg/facebook/about/?ref=page_internal). Acesso em 20 set. 2017.

Para configurar a integração com o Facebook Messenger, o desenvolvedor deve selecionar o botão *integrations*, no *Dialogflow*, e habilitar a opção de Facebook Messenger; feito isto, será exibida uma tela semelhante à da Figura 16:

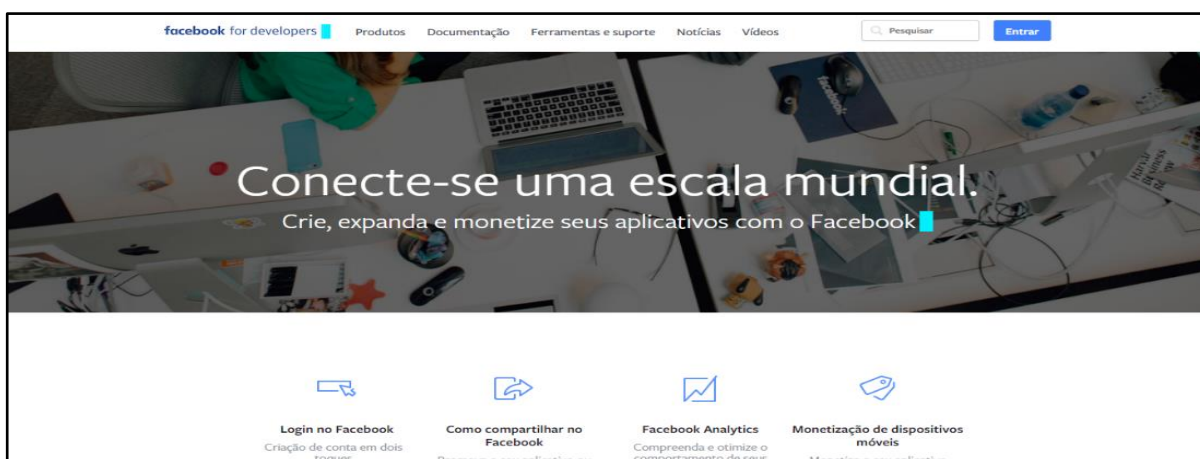
Figura 16 - Integração com o Facebook Messenger

Fonte: Página oficial da *Dialogflow*, 2017.<sup>35</sup>

Para o desenvolvedor obter o *Access Token* (Token de acesso), primeiro precisa-se criar uma *Facebook Page* (Página de Facebook), selecionar o tipo da página como entretenimento e, como personagem fictício, dará um nome para sua página.

Após a criação da página, deve-se acessar a página de desenvolvimento do Facebook, conforme Figura 17.

Figura 17 - Facebook Developers



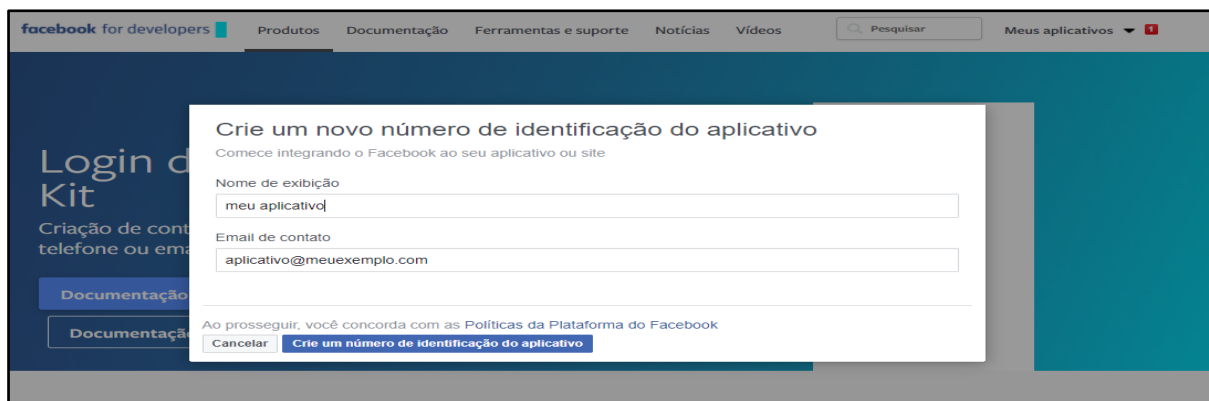
Fonte: Página inicial Facebook Developers<sup>36</sup>

<sup>34</sup> Disponível em <https://olhardigital.com.br/noticia/facebook-messenger-ja-tem-1-2-bilhao-de-usuarios-ativos-por-mes/67480>. Acesso em 20 set. 2017.

<sup>35</sup> <https://console.dialogflow.com/api-client/#/agent/abd3ab6d-5867-4a34-9d0f-0e7e61549a00/integrations>.

Já autenticado em sua conta, o desenvolvedor precisa selecionar o botão “meus aplicativos” e clicar na opção “adicionar um novo aplicativo”. O desenvolvedor deve atribuir um nome de exibição e dizer em qual categoria seu aplicativo se encontra; por fim, deve clicar no botão “criar um número de identificação do aplicativo” e digitar o código de verificação, e clicar em “enviar”(Figura 18).

Figura 18 - Criando novo app no Facebook Messenger



Fonte: Página oficial do Facebook Developers, 2017.<sup>37</sup>

Neste ponto, o desenvolvedor será direcionado para o painel de administração do aplicativo, onde deve selecionar “configurações”, à esquerda (produtos → Messenger → configurações); ele deve configurar o aplicativo para ter permissão de acesso; para tanto, precisa selecionar a página que criou na opção “geração de token”; será aberto um *plug-in* na tela para conceder permissão: o desenvolvedor deve clicar no botão “continuar” e, por fim, no botão “ok”.

Será, então, gerado um *token* (Chave Secreta do Aplicativo – Figura 19), que o desenvolvedor deve colocar na opção “*Page Access Token*” no *Dialogflow*.

Figura 19 - Chave Secreta do Aplicativo



Fonte: Página oficial do Facebook Developers, 2017.<sup>38</sup>

<sup>36</sup> <https://developers.facebook.com/>

<sup>37</sup> <https://developers.facebook.com/products/account-creation>.

Por fim, o desenvolvedor deve atribuir um nome para verificação de token (*Verify Token*) no *Dialogflow* e clicar no botão “start”. Posteriormente, deve acessar, no painel de configuração do aplicativo, a opção “configurações” (produtos → Messenger → configurações), e selecionar a opção “Configurar Webhooks”.

No campo “Verificar token”, o desenvolvedor deve atribuir o mesmo nome de verificação que colocou no *Dialogflow*, e a mesma URL do campo *Callback URL* deve ser inserida no campo “URL de retorno de chamada”, conforme Figura 20:

Figura 20 - Configurando Webhooks

Fonte: Página oficial da *Dialogflow*, 2017.<sup>39</sup>

Na mesma página, deve definir os campos de assinaturas: o desenvolvedor precisa selecionar as opções que seu agente terá permissão para utilizar. O agente que está sendo desenvolvido no presente trabalho terá as seguintes permissões: *messages*, *messaging postbacks*. Por fim, o desenvolvedor deve clicar no botão “verificar e salvar”.

Realizados os procedimentos acima, o desenvolvedor deve testar sua aplicação; o teste inicia-se acessando a página criada no Facebook (a mesma com a qual foi realizada a integração); deve-se clicar no botão “Começar” para testar a aplicação (Figura 21). Ela está disponível no seguinte endereço eletrônico: <https://www.facebook.com/P%C3%B3s-Bot-Teste-430085690673015/>.

Figura 21 - Tela da aplicação no Facebook



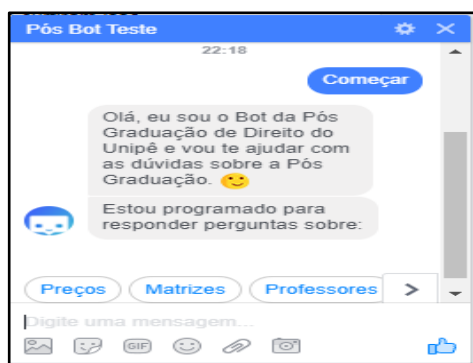
Fonte: Bot no Facebook<sup>40</sup>

<sup>38</sup> <https://developers.facebook.com/apps/1360852697339894/dashboard/>.

<sup>39</sup> <https://console.dialogflow.com/api-client/#/agent/abd3ab6d-5867-4a34-9d0f-0e7e61549a00/webhooks>.

Feito isto, o agente irá retornar uma mensagem de saudação, que foi especificada pelo desenvolvedor. No caso do agente “Pós Bot Teste”, a mensagem inicial dele é a mostrada na Figura 22.

Figura 22 - Mensagem de saudação



Fonte: Bot no Facebook, 2017.<sup>41</sup>

A partir deste ponto, pode-se perceber que o agente está em pleno funcionamento, e que os próximos diálogos dependem da interação Humano & Robô. No entanto, o agente ainda não está disponível para o público; o teste obteve êxito utilizando uma conta de administrador da página, não disponível a terceiros.

Para disponibilizar o agente ao público geral, deve-se acessar o endereço <https://developers.facebook.com/> e clicar em “Meus aplicativos → Painei → Revisão do Aplicativo”; será exibida tela solicitando permissão para tornar o aplicativo público (Figura 23).

Figura 23 - Facebook Developers – Revisão de aplicativo



Fonte: Facebook Developers, 2017.<sup>42</sup>

<sup>40</sup> <https://www.facebook.com/P%C3%B3s-Bot-Teste-430085690673015/>

<sup>41</sup> <https://www.facebook.com/P%C3%B3s-Bot-Teste-430085690673015/>

<sup>42</sup> <https://developers.facebook.com/>

Ao selecionar o botão “Sim”, o aplicativo será preparado para ser enviado para análise dos avaliadores do Facebook, que irão testá-lo; obtendo êxito, eles retornam o feedback informando que o mesmo está apto a ser utilizado.

Para o envio, precisa-se clicar no botão “Adicionar Itens” (Figura 24) e selecionar a opção “*pages\_messaging*”. Posteriormente, deve-se clicar no botão “visualizar notas” e associar a página que está relacionada ao aplicativo.

Figura 24 - Facebook Developers – Adicionar itens



Fonte: Facebook Developers, 2017.<sup>43</sup>

Nesta etapa o desenvolvedor deve marcar a opção “Sua experiência do Messenger inclui respostas automatizadas” e especificar um exemplo de comandos que seu *Bot* está apto a responder, informando a resposta; pode-se colocar quantos comandos quiser (Figura 25).

Figura 25 - Facebook Developers – *pages\_messaging*

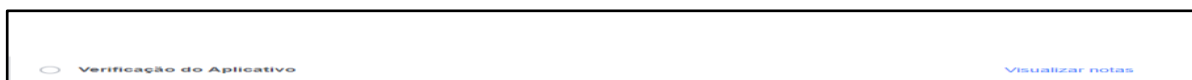
Fonte: Facebook Developers, 2017.<sup>44</sup>

<sup>43</sup> <https://developers.facebook.com/>

<sup>44</sup> <https://developers.facebook.com/>

Realizada esta etapa, deve-se clicar em “visualizar notas”, da opção “verificação de aplicativo”, selecionar o usuário de teste padrão “*Open Graph Test User*” e clicar em salvar, conforme Figura 26.

Figura 26 - Facebook Developers – Verificação do aplicativo



Fonte: Facebook Developers, 2017<sup>45</sup>

No entanto, o aplicativo só irá para revisão se estiver configurado com um ícone, uma plataforma e uma política de url. O desenvolvedor pode realizar esta operação através do caminho “Painel → Configurações → Básico”, conforme figura abaixo:

Figura 27 - Facebook Developers - configurações

Fonte: Facebook Developers, 2017.<sup>46</sup>

Por último, o desenvolvedor precisa acessar o seguinte endereço para enviar seu Bot para avaliação: “Painel → Messenger → Configurações”. No final da página, deve selecionar o botão “Enviar para análise”, aceitar os termos de uso e clicar no botão “Enviar”. Feito isto, o sistema vai exibir um plugin-in na tela informando que, em média, os envios serão analisados dentro de cinco dias.

Realizados corretamente todos os procedimentos acima, no menu “alertas” será gerado um alerta informando que recebeu o seu envio.

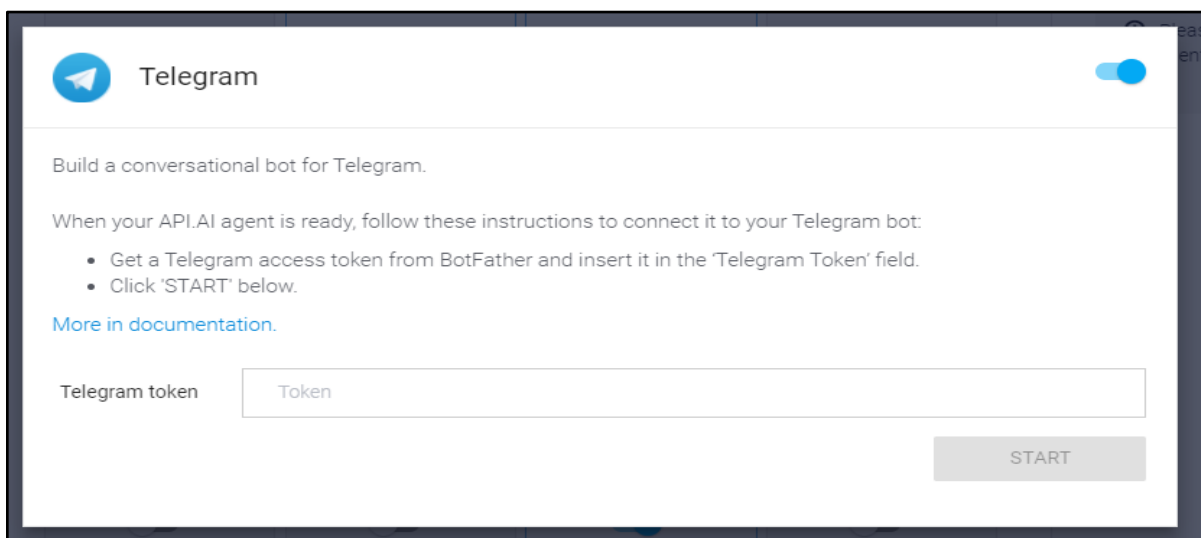
<sup>45</sup> <https://developers.facebook.com/apps/1360852697339894/review-status/>

<sup>46</sup> <https://developers.facebook.com/apps/1360852697339894/settings/>



TELEGRAM - O processo de integração com o Telegram inicia-se com o desenvolvedor selecionando no *Dialogflow* a opção Telegram (Figura 28). Posteriormente, deve-se acessar o endereço <https://web.telegram.org/#/login> e informar um número de celular; é necessário ter o aplicativo Telegram já instalado previamente.

Figura 28 - Telegram - gerando access token



Fonte: Página oficial do *Dialogflow*, 2017.<sup>47</sup>

Quando o desenvolvedor acessar sua conta do Telegram, o mesmo vai pesquisar no console a sua esquerda, pelo nome BotFather e vai selecionar. Na qual o mesmo irá digitar o comando `/start` (Figura 29) para iniciar o processo da geração de seu token para integração entre *Dialogflow* e o Telegram.

Figura 29 - Telegram → BotFather - iniciando o BotFather



Fonte: Telegram, 2017.<sup>48</sup>

<sup>47</sup> <https://console.dialogflow.com/api-client/#/agent/abd3ab6d-5867-4a34-9d0f-0e7e61549a00/integrations>

<sup>48</sup> <https://web.telegram.org/>

Feito isto, o robô virtual do *BotFather* vai retornar uma mensagem com as opções que deseja escolher, como na Figura 30, em que disponibiliza 20 opções de configurações diferentes.

Figura 30 - Telegram → BotFather - criando o bot para o telegram



Fonte: Telegram, 2017.<sup>49</sup>

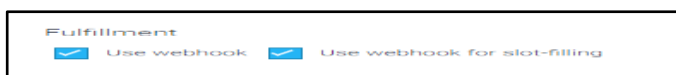
O desenvolvedor do agente vai selecionar o comando */newbot*, que iniciará o processo para criar o agente para o Telegram. Posteriormente, o *BotFather* vai retornar a mensagem perguntando que nome o desenvolvedor deseja colocar em seu agente. Informado o nome, o *BotFather* vai retornar uma mensagem comunicando que seu *bot* foi criado, dando-lhe os parabéns.

Logo abaixo será exibida a seguinte mensagem: *Use this token to access the HTTP API* (Use este token para acessar a API HTTP), com o token que deve ser colocado no *Dialogflow*.

### 3.2.5 Estratégias de Integração – backend

**WEBHOOK** - Serve para acionar APIs externas para buscar valores em bancos de dados; obtidos esses valores, retorna para a intent, até chegar no usuário. Utilizada selecionando as opções “*Use webhook*” e “*Use webhook for slot-filling*” (Figura 31).

Figura 31 - DialogFlow - webhook



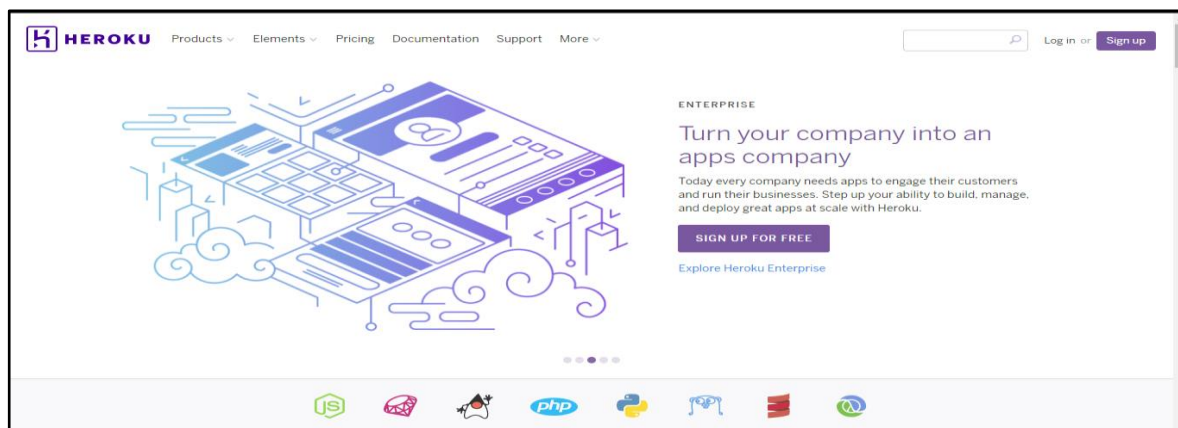
Fonte: Página oficial do *Dialogflow*, 2017.<sup>50</sup>

<sup>49</sup> <https://web.telegram.org/>

<sup>50</sup> <https://console.dialogflow.com/api-client/#/agent/abd3ab6d-5867-4a34-9d0f-0e7e61549a00/intents>.

HEROKU - Para criar uma conta no Heroku, o desenvolvedor deve acessar o seguinte endereço: <https://www.heroku.com/>, clicar na opção *Sign up* (Inscrever-se) e informar seus dados cadastrais (Conforme Figura 32).

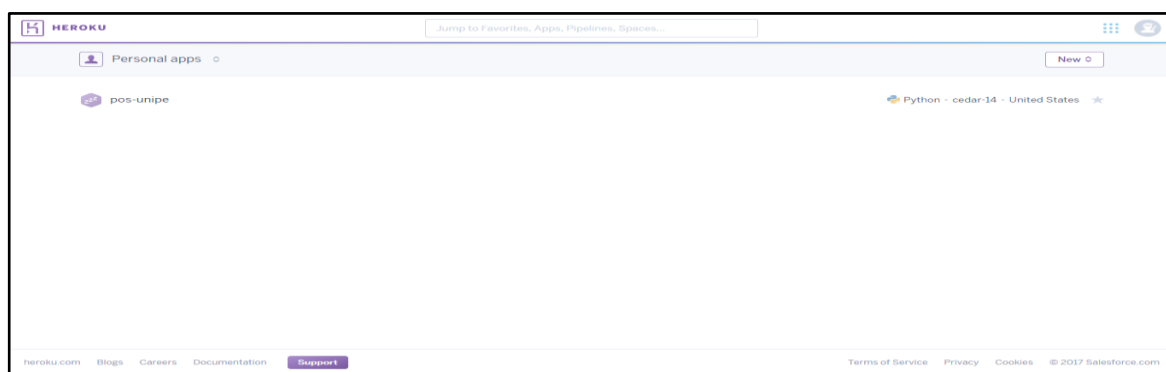
Figura 32 - Heroku - tela inicial



Fonte: Heroku, 2017.<sup>51</sup>

Após ter realizado seu cadastro e se autenticado no Heroku, o desenvolvedor será direcionado para a tela personal apps (Figura 33), onde vai selecionar a linguagem de programação do aplicativo que será criado; posteriormente, deve clicar na opção “*Create New App*” (Criar novo aplicativo).

Figura 33 - Heroku - personal apps



Fonte: Heroku, 2017<sup>52</sup>

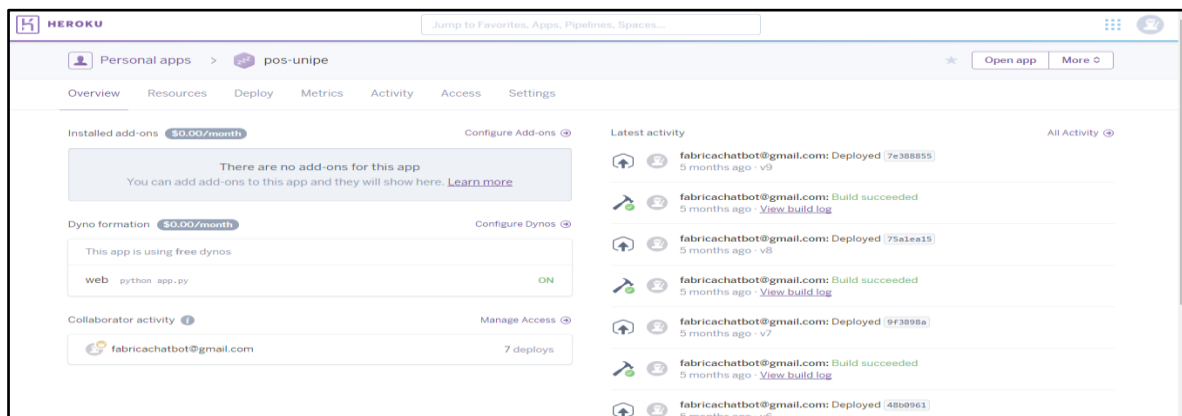
Feito isso, o desenvolvedor deve atribuir um nome para sua aplicação, e marcar a opção em que o back-end será criado; a plataforma disponibiliza duas opções: EUA e Europa; para este trabalho, foi selecionado a opção EUA.

<sup>51</sup> <https://www.heroku.com/>

<sup>52</sup> <https://www.heroku.com/apps>

A Figura 34 demonstra uma visão geral da plataforma Heroku, onde informa todas as funcionalidades da aplicação, tais como: Overview, Resources, Deploy, Metrics, etc.:

Figura 34 - Heroku - overview



Fonte: Heroku, 2017.<sup>53</sup>

Na opção *Deploy*, o desenvolvedor tem todas instruções de como instalar o Heroku, através do link que é disponibilizado: *Download and install the Heroku CLI* (Baixe e instale a CLI Heroku.), em que o desenvolvedor clica na opção, e a plataforma abre uma nova aba com as instruções para instalar no sistema operacional que o desenvolvedor desejar.

O Sistema operacional utilizado para realizar o deploy desta aplicação foi o Sistema Operacional Linux/Ubuntu. A plataforma está disponível para os seguintes sistemas operacionais: Mac OS, Debian/Ubuntu, Windows. Este módulo do agente que utilizou a opção WebHooks, foi desenvolvido na Linguagem de Programação Python. Este código será posteriormente disponibilizado no GitHub (Figura 35).

Figura 35 - Python - webhook



Fonte: Elaborado pelos Autores, 2017.

<sup>53</sup> <https://www.heroku.com/apps>

A funcionalidade deste código é fazer uma comunicação através do Firebase e *Dialogflow*. Para isto, no método *def makeWebhookResult(req)* é definida a *action* que está sendo utilizada no *Dialogflow*.

Também é definido uma variável *result* e uma variável *parameters* para receber os valores da Entidade em que foi especificado os cursos da pós-graduação.

Depois, é importado o firebase e definida a variável *firebase* para receber os dados que foram definidos na opção *database* do firebase.

Posteriormente, é criado um *if* para o caso de o usuário digitar o nome do curso corretamente ( ex: “Carga horária Ciências Criminais”: o sistema irá exibir através da variável *speech*).

Caso o usuário digite uma palavra isolada (ex: Carga horária) e a api não identifique o curso, será retornada uma lista com a carga horária de todos os cursos para o usuário escolher; o usuário escolhe e a api exibe a Carga horária do curso selecionado no console.

Por último, é feito um *if name* para rodar o código no servidor local e assim ter uma url para testar na ferramenta Postman.

### 3.3 SERVIÇOS UTILIZADOS

Para o aplicativo desenvolvido neste Projeto Aplicado, utilizou-se os serviços REST (abstração da arquitetura *www*) e o banco de dados *NOSQL* *Firebase*.

#### 3.3.1 REST

O Postman é um cliente Rest; o aplicativo, que é gratuito, foi criado em outubro de 2012, e logo tornou-se um dos aplicativos mais populares na Chrome Store, com mais de 3 milhões de desenvolvedores inscritos<sup>54</sup>.

O Postman serve para testar requisições HTTP; isto pode ser feito através de vários métodos (verbos); as respostas das solicitações feitas são enviadas no formato *Json*.

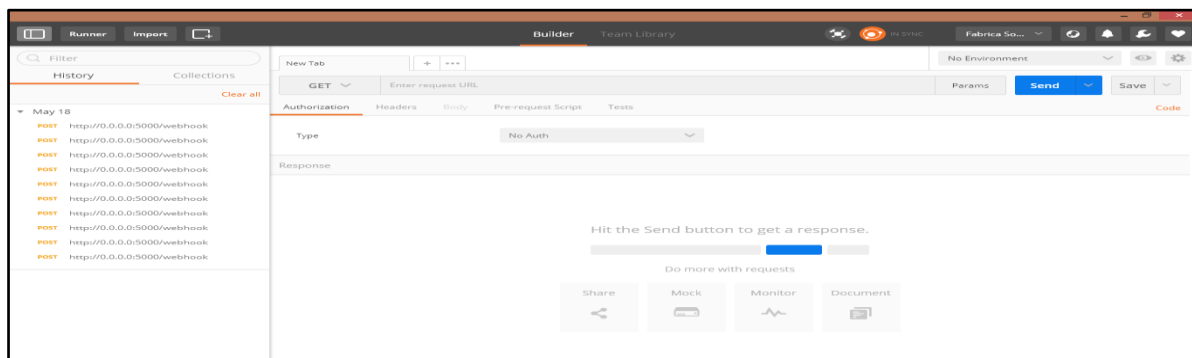
Os testes desta aplicação foram feitos por meio da url do servidor local: *http://0.0.0.0:5000/webhook*; o método utilizado foi o *Post*.

---

<sup>54</sup> Disponível em: <<https://www.getpostman.com/apps>>. Acesso em: 21 out. 2017.

A figura 36 demonstra o layout da tela inicial do Postman e algumas de suas funcionalidades.

Figura 36 - Postman - tela inicial



Fonte: Elaborado pelos Autores, 2017.

### 3.3.2 Utilização do Banco NOSQL (Firestore)

Para criar a árvore de nós para se comunicar com a parte do back-end, foi utilizado o Firestore. Foi criado um projeto no Firestore com o mesmo nome que foi definido no Heroku, neste caso o nome foi pos-unipe. No menu database à esquerda do Firestore foi criado uma árvore de nome Cursos contendo os valores de carga horária (Figura 37) que vão se comunicar com a parte do back-end.

Figura 37 - Firestore - database



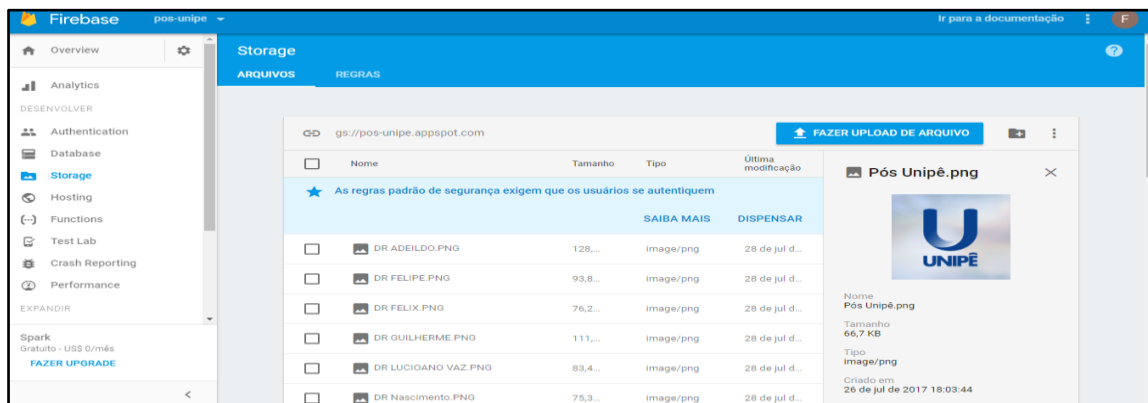
Fonte: Firestore, 2017.<sup>55</sup>

Neste trabalho foi utilizado a opção Storage do Firestore para armazenar as imagens que foram utilizadas na plataforma *Dialogflow*. Ao fazer o upload, é gerado um link que o

<sup>55</sup> <https://console.firebase.google.com/project/pos-unipe/database/pos-unipe/data>

desenvolvedor deve copiar e colocar na opção em que for utilizar na plataforma *Dialogflow*; neste trabalho foi utilizada esta função para a criação do Card (Conforme Figura 38).

Figura 38 - Firebase - storage



Fonte: Firebase, 2017<sup>56</sup>.

<sup>56</sup> <https://console.firebase.google.com/project/pos-unipe/storage/pos-unipe.appspot.com/files?hl=pt-br>

## **4 – RESULTADOS**

Este capítulo demonstra os resultados do processo de desenvolvimento da aplicação, através de avaliação por meio de pesquisa realizada utilizando a plataforma Google Forms.

### **4.1 AVALIAÇÃO DO CHATBOT DESENVOLVIDO**

Neste capítulo será feita uma análise da aplicação desenvolvida; para isso, foram feitos testes com 31 (trinta e uma) pessoas, para termos um feedback das funcionalidades do aplicativo, facilidades de uso, bem como, se ele atende na forma esperada.

Primeiramente os usuários recebiam o link da página no Facebook e pressionava o botão começar para iniciar o diálogo com o chatbot, posteriormente os usuários deveriam acessar o link do formulário e responder o questionário. O questionário foi fragmentado em três etapas: identificação do perfil, experiência do usuário, sugestões e opiniões.

Na primeira etapa do formulário eram respondidas perguntas tais como: sexo, idade, grau de escolaridade, área de atuação, interesse na pós graduação de direito, qual curso escolheria, renda. Na segunda etapa eram feitas perguntas sobre: chatbot atendeu às necessidades, qual a experiência de ser atendido por um robô virtual, quantas respostas inapropriadas o agente gerou durante sua interação, como foi a interação do agente quanto à: facilidade de uso, naturalidade, robustez, coerência. Na terceira etapa eram feitas duas perguntas: você interagiria com este agente novamente? caso não o usuário deveria especificar o motivo de não interagir com esse agente novamente.

#### **4.1.1 Identificação de Perfil**

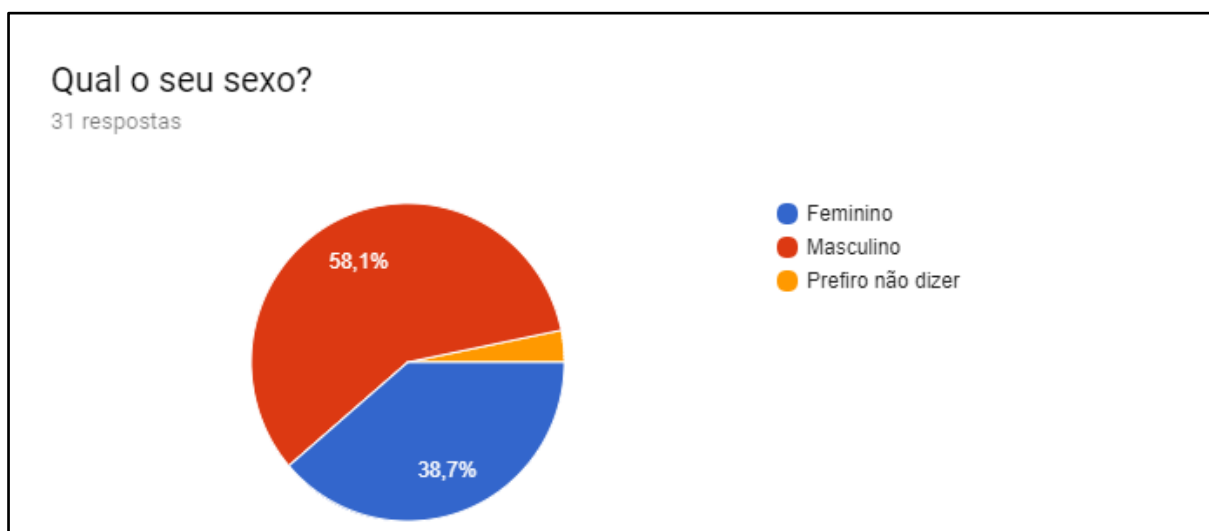
A partir das respostas do questionário, pôde-se identificar que os usuários eram na maioria homens com idade entre 18 e 24 anos com o grau de escolaridade cursando curso superior ou completo, com renda média de até três salários mínimos.

As páginas seguintes apresentam gráficos referentes a sexo, idade, grau de escolaridade, renda média, área de atuação e percentual de interessados nos cursos de pós-graduação da instituição de ensino estudada; por fim, identifica os cursos mais buscados.



Em relação ao sexo dos entrevistados, percebe-se que há uma maioria de homens, mas a participação feminina é considerável, conforme visto na Figura 39.

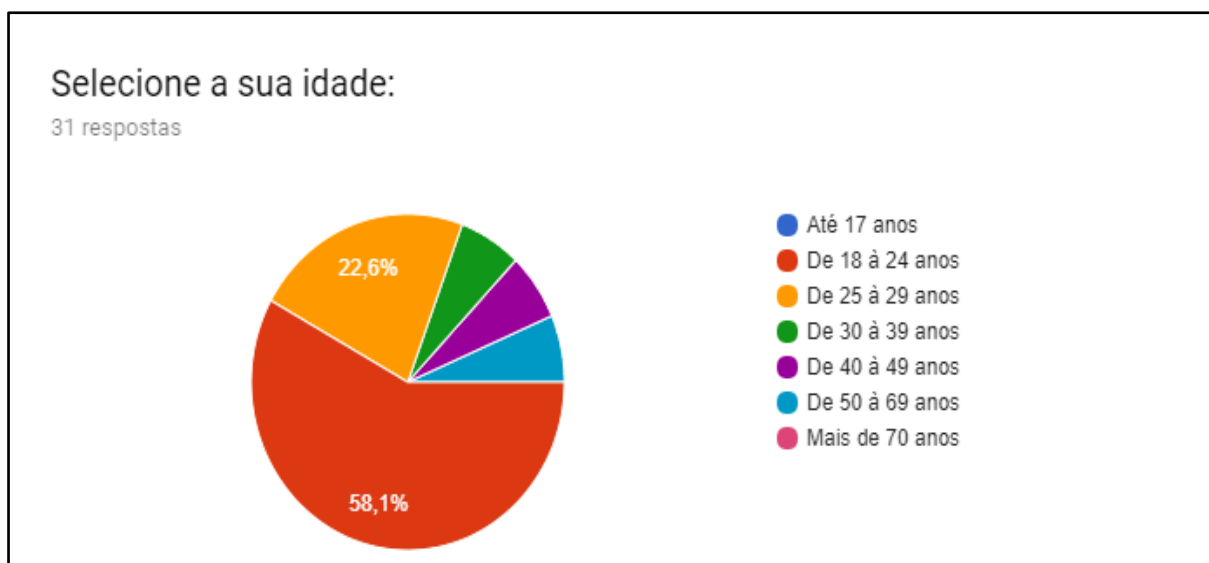
Figura 39 - Dados referentes ao sexo das pessoas que utilizaram o agente



Fonte: Elaborado pelos Autores, 2017.

Já em relação à faixa etária, predominam jovens entre 18 e 29 anos, com mais de 80% do total de entrevistados; este fato reflete a faixa etária de procura por cursos de pós-graduação (Figura 40).

Figura 40 - Dados referentes às idades das pessoas que utilizaram o agente



Fonte: Elaborado pelos Autores, 2017.

Quanto ao grau de escolaridade, a ampla maioria (87%) está cursando o superior ou já é graduado (Figura 41).

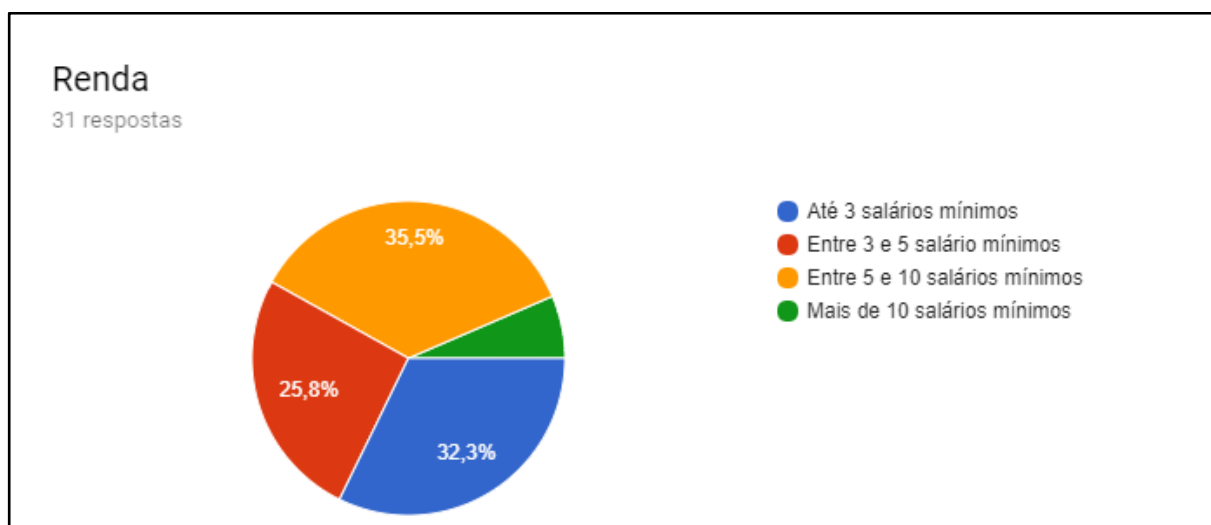
Figura 41 - Grau de escolaridade das pessoas que utilizaram o agente



Fonte: Elaborado pelos Autores, 2017.

Em relação à faixa de renda, existe um equilíbrio nas faixas de até 10 salários mínimos, conforme demonstra a Figura 42.

Figura 42 - Renda média das pessoas que utilizaram o agente

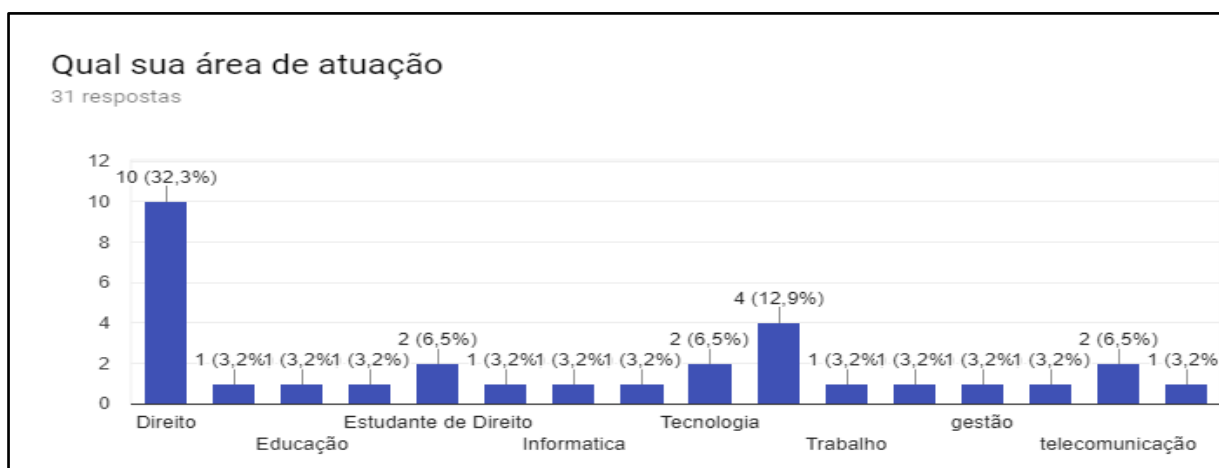


Fonte: Elaborado pelos Autores, 2017.

#### 4.1.2 Interesse na Pós-Graduação

A partir das 31 respostas coletadas no questionário, pôde-se identificar, conforme Figura 43, que 38% dos usuários atuavam na área de direito, sendo 2 estudantes; que 61,3% do total de entrevistados demonstraram interesse em cursar algum curso da pós de direito. Observou-se que 21 dessas pessoas (67%) optaram por informar o seu curso; verificou-se maior ocorrência nos cursos de Criminologia e Psicologia Investigativa Criminal (19% da preferência).

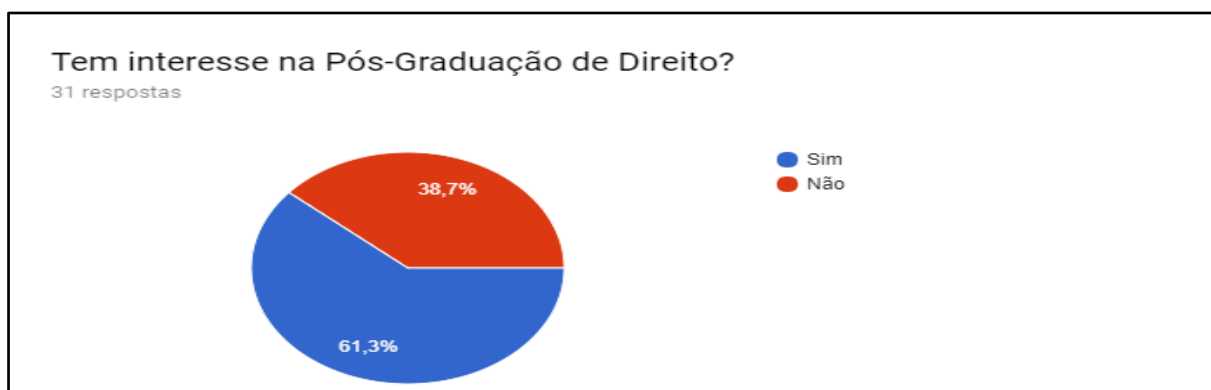
Figura 43 - Área de atuação das pessoas que utilizaram o agente



Fonte: Elaborado pelos Autores, 2017.

Percebemos, também, que o percentual de interessados na pós-graduação de direito, embora em um bom percentual (61,3% dos entrevistados), tem ainda um bom campo para crescimento (Figura 44).

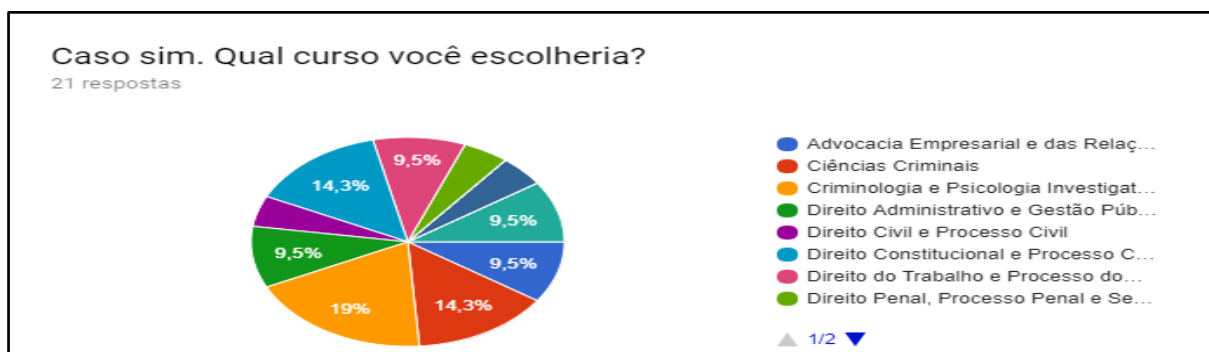
Figura 44 - Percentual das pessoas que demonstraram interesse na pós



Fonte: Elaborado pelos Autores, 2017.

Daqueles que demonstraram interesse na pós, grande parte optaria por Ciências Criminais (14,3%), Direito Constitucional (14,3%) e Criminologia (19%), perfazendo quase metade dos entrevistados (47,6%), conforme Figura 45.

Figura 45 - Interesse nos cursos da pós-graduação através do agente



Fonte: Elaborado pelos Autores, 2017.

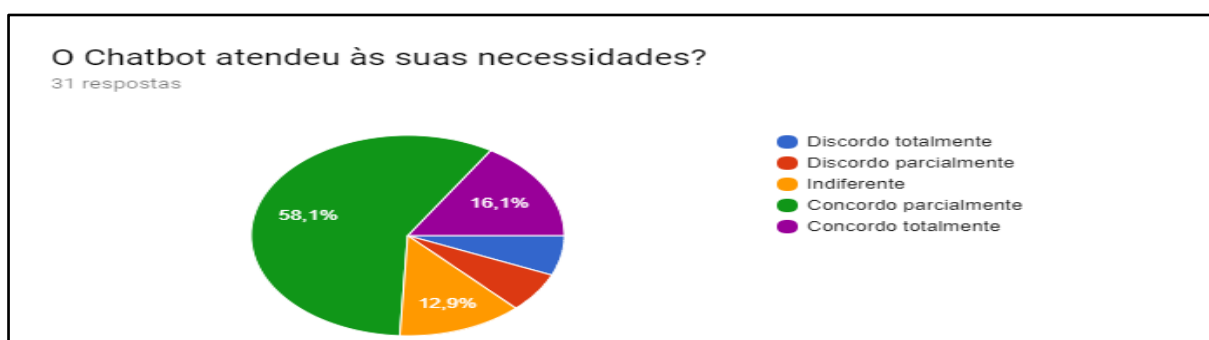
A Figura 45 ficou com respostas inferiores à outras por não ser obrigatória, e os entrevistados só deviam responder caso tivessem interesse no curso.

#### 4.1.3 Experiência do Usuário

Esta terceira parte do questionário visa relatar as experiências dos usuários quanto ao uso do chatbot da Pós-Graduação de Direito; os usuários utilizam a plataforma do Facebook Messenger para relatar sua experiência.

A Figura 46 demonstra que o chatbot atendeu parcialmente as necessidades dos usuários, sendo necessária a retroalimentação e cuidado dos curadores para elevar o grau de satisfação com a ferramenta.

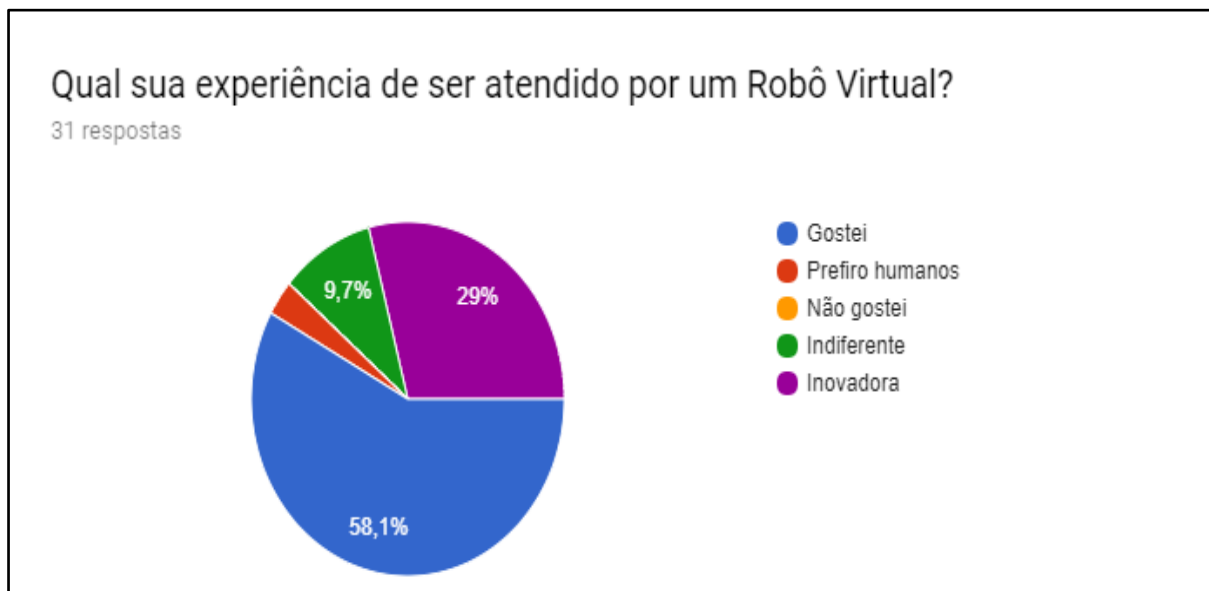
Figura 46 - Dados referentes à qualidade de atendimento do agente



Fonte: Elaborado pelos Autores, 2017.

Quanto à experiência de ser atendido por um robô, a maioria absoluta informou ter gostado (Figura 47).

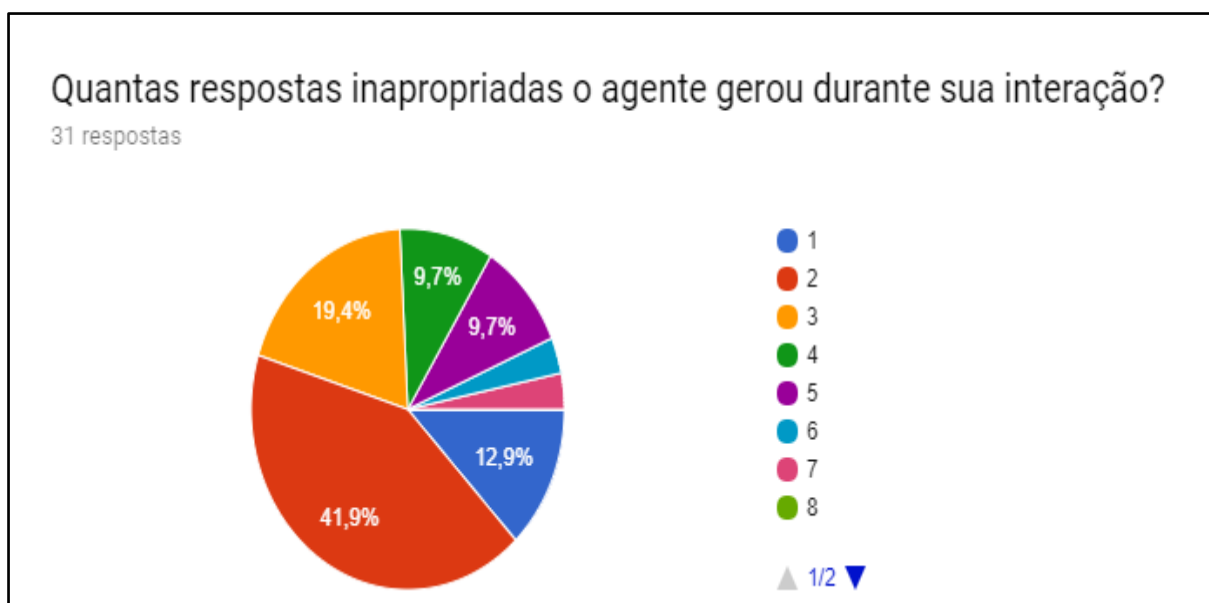
Figura 47 - Dados referentes à experiência das pessoas que utilizaram o Robô Virtual



Fonte: Elaborado pelos Autores, 2017.

Já o percentual de respostas inapropriadas da ferramenta foi baixo, com mais da metade dos usuários informando que houve até 2 (duas) ocorrências (Figura 48).

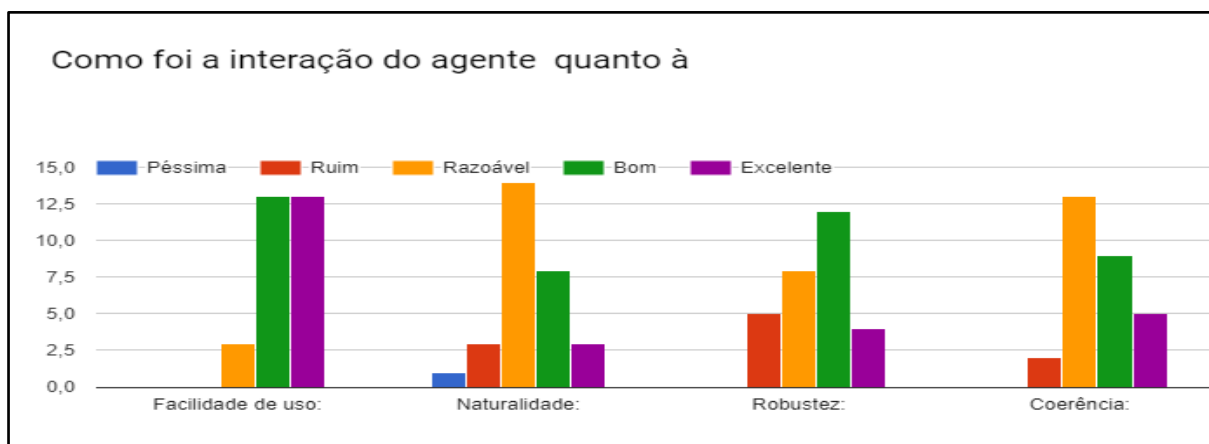
Figura 48 - Percentual da quantidade de respostas inapropriadas durante a interação Humano - Robô



Fonte: Elaborado pelos Autores, 2017.

A Figura 49 informa dados referentes à experiência do usuário quanto à facilidade de uso, naturalidade, robustez e coerência; de um modo geral, a ferramenta teve uma recepção de razoável para boa, e a facilidade de uso foi o destaque da avaliação.

Figura 49 - Dados referentes à experiência das pessoas com o agente



Fonte: Elaborado pelos Autores, 2017.

#### 4.1.4 Sugestões e opiniões do usuário

Esta quarta parte do questionário visa perguntar aos usuários se eles interagiriam com o agente novamente; o resultado da pesquisa foi positivo (vide Figura 50), porém com algumas críticas.

Figura 50 - Dados referentes a opinião das pessoas quanto a uma nova interação com o agente



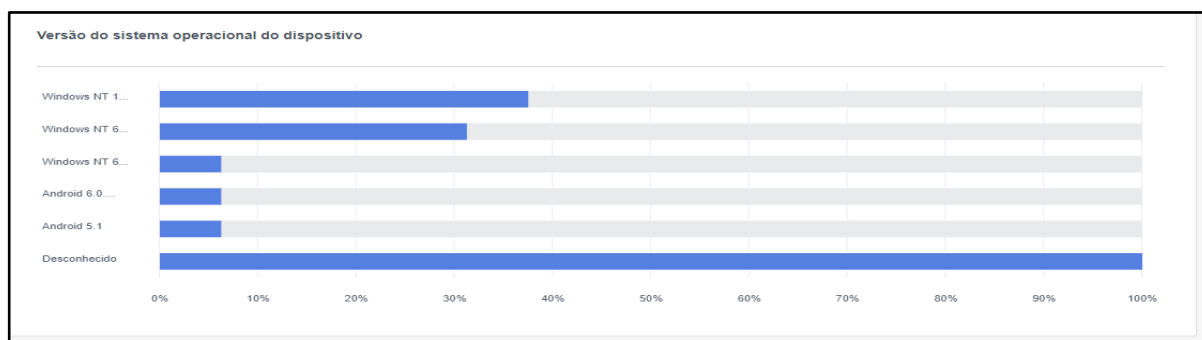
Fonte: Elaborado pelos Autores, 2017.

Em relação às opiniões, percebeu-se que os usuários nunca tinham interagido com um chatbot, ocasionando em opiniões tais como: “Eu penso que a resposta do chatbot, deveria ser respondida em um tempo mais natural, como se fosse um humano, e não instantaneamente.”

## 4.2 ANALYSIS CHATBOT NO MESSENGER

Através da opção visualizar analytics da plataforma Facebook Developers, é disponibilizado um painel para identificar quais meios estão sendo usados para acessar o Bot no Facebook Messenger; neste gráfico (Figura 51) pode-se identificar a versão do sistema operacional que os usuários estão usando para utilizar o Bot no Messenger.

Figura 51 - Facebook Analytics - Versão do sistema operacional



Fonte: Facebook Analytics, 2017.<sup>57</sup>

Outra opção disponibilizada nesta plataforma é de monitorar as atividades dos usuários, na qual permite-se visualizar a quantidade de usuários, total de eventos e média de eventos por usuário (Figura 52).

Figura 52 - Facebook Analytics - atividade dos usuários



Fonte: Facebook Analytics, 2017.<sup>58</sup>

<sup>57</sup> <https://www.facebook.com/analytics/1360852697339894/?section=technology>

<sup>58</sup> [https://www.facebook.com/analytics/1360852697339894/?\\_\\_aref\\_src=devsite&\\_\\_aref\\_id=app\\_dashboard](https://www.facebook.com/analytics/1360852697339894/?__aref_src=devsite&__aref_id=app_dashboard)

Esta análise é bastante útil, pois identifica picos e vales nas interações, ajudando a planejar a disponibilidade da aplicação e o pessoal necessário (curadores) em cada período de utilização.



## 5 CONCLUSÃO

O presente trabalho obteve sucesso na integração do aplicativo com as redes sociais, tendo sido destaque a integração com o Facebook Messenger, por ser a rede social de maior utilização no Brasil, e demonstrou ser intuitivo e de fácil utilização pelos usuários, atendendo a proposta inicial de ser um repositório para sanar as dúvidas dos usuários.

A facilidade de uso da ferramenta demonstra que foi acertada a ideia inicial de se utilizar um robô virtual para solucionar dúvidas gerais, e até algumas específicas, em relação aos cursos de pós-graduação na área de direito da instituição pesquisada. A ferramenta pode, também, ser desenvolvida para outras instituições de ensino, como escolas de ensino fundamental e médio, universidades e cursinhos, com tendência de ser uma das ferramentas mais utilizadas pelos alunos destas instituições.

O levantamento dos requisitos do sistema demonstrou ser conclusivo, e as demonstrações dos benefícios da utilização da aplicação surtiram o efeito desejado, com boa receptividade dos usuários e funcionários da pós-graduação nos cursos de direito. A ausência de solução semelhante no mercado criou expectativas e dúvidas do cliente; o sucesso da aplicação levou o cliente a acreditar no retorno advindo da sua utilização e manutenção.

Como todo *bot* que trabalha com inteligência artificial, o aprendizado de máquina é lento e baseado na utilização da ferramenta, interações com os usuários, retroalimentação das informações pelos curadores da aplicação e anexação de novas *intents* e *entities*, motivo pelo qual sua base de dados é ainda simples. Com a sequência de interações e incrementos na base de dados, entretanto, terá sem dúvida uma base sólida em um médio prazo.

Os curadores terão uma aplicação com fácil interpretação das interações, podendo incrementar a base de dados facilmente. Obviamente, se faz necessário um treinamento da equipe de curadoria, em relação à utilização da ferramenta, sobre como inserir *intents* e alimentar o *bot* a partir das interações dos usuários. Este treinamento pode fazer parte de trabalhos futuros, inclusive, com possibilidade de utilização do conhecimento adquirido pelos alunos que desenvolveram o presente trabalho.

Percebe-se que os objetivos deste trabalho de conclusão de curso foram atingidos, e que a aplicação pode se tornar, em um futuro próximo, o principal meio de interação dos usuários com a pós-graduação de um Centro Universitário localizado no município de João Pessoa.

Em relação aos trabalhos futuros, há consenso de que a ferramenta tende a ser o principal meio de comunicação em poucos anos, principalmente nas interações entre

organizações de todos os setores e seus clientes; na área educacional, o presente trabalho nos deu uma base sólida para desenvolver novas aplicações para universidades, cursinhos, institutos educacionais e afins, apresentando um horizonte de possíveis trabalhos nesta área.

## REFERÊNCIAS

ABREU, Bruno. **Revolução? O crescimento expressivo de Chatbots e por que testá-los.** Disponível em: <<http://blog.onedaytesting.com.br/chatbots-crescimento/>>. Acesso em 22 ago. 2017.

BARBOSA, Marystela. **ChatBots: uma tendência cada vez mais forte na tecnologia.** Disponível em: <<https://startupi.com.br/2016/10/chatbots-uma-tendencia-cada-vez-mais-forte-na-tecnologia/>>. Acesso em 10 abr. 2017.

BASSET, Lindsay. **Introdução ao JSON.** São Paulo: Novatec, 2015.

BENGFORT, Benjamin; KIM, Jenny. **Analítica de Dados com Hadoop – Uma Introdução para Cientistas de Dados.** São Paulo: Novatec, 2016.

BORGES, Luiz Eduardo. **Python para Desenvolvedores.** Rio de Janeiro: Edição do autor, 2010.

CAMPOS, André Luís Nogueira. **Modelagem de Processos com BPMN (2ª edição).** Rio de Janeiro: Brasport, 2014.

CHATBOTS MAGAZINE. **Tutorials on Chatbots.** Disponível em: <<https://chatbotsmagazine.com/tutorials/home>>. Acesso em 21 abr. 2017.

COELHO, Flávio Codeço. **Computação Científica com Python.** Petrópolis (RJ): Edição do autor, 2007.

COPPIN, Ben. **Artificial Intelligence Illuminated.** Sudbury MA (EUA): Jones & Bartlett Publishers, Inc., 2010.

DIALOGFLOW, **Docs – Basics.** Disponível em: <<https://dialogflow.com/docs/getting-started/basics>>. Acesso em: 26 ago. 2017.

DOMINGOS, Pedro. **O Algoritmo Mestre.** São Paulo: Novatec, 2017.

FACEBOOK FOR DEVELOPERS. **Documentos / Plataforma do Messenger.** Disponível em: <<https://developers.facebook.com/docs/messenger-platform>>. Acesso em 23 abr. 2017.

FIREBASE, **Documentos – Guias – Reaktune Database – Web – Estruturar dados.** Disponível em: <<https://firebase.google.com/docs/database/web/structure-data?hl=pt-br>>. Acesso em 23 ago. 2017.

FIREBASE, **Documentos – Guias – Realtime Database - Introdução.** Disponível em: <<https://firebase.google.com/docs/database/?hl=pt-br>>. Acesso em: 26 ago. 2017.

GARIBA, M. Jr., Schneider, M. C. K., Rosa, A. E., Casagrande, J. B., Santos, C. S. **Reconhecimento de Fala e Processamento da Linguagem Natural.** Disponível em: <<http://www.cefetsc.edu.br/~gariba/VOZ.PRN.pdf>>. Acesso em: 05 de Julho 2017.

GERHARDT, Tatiana Engel; Silveira, Denise Tolfo. **Métodos de Pesquisa**. Porto Alegre: SEAD/UFRGS, 2009.

GONÇALVES, André Luz. **Desenvolvimento de um aplicativo android utilizando banco de dados não-relacional para organização e controle de presença de um time de futebol**. Porto Alegre: UFRS, 2016.

HADOT, Pierre. **Eloge de la philosophie antique**. Paris: Allia, 2006.

LAVEN, Simon James. **The Simon Laven Page**. Disponível em: <<http://www.simonlaven.com/>>. Acesso em 23 abr. 2017.

MAULDIN, Michael Loren. **CHATTERBOTS, TINYMUDs, and the Turing Test: Entering the Loebner Prize Competition**. Pittsburgh: Carnegie Mellon University, 1994.

MCCARTHY, John. **A Proposal for the Dartmouth Summer Research Project on Artificial Intelligence**. Hannover (USA): Dartmouth College, 1956.

MOLINARI, William. **Desconstruindo a Web: As Tecnologias por trás de uma Requisição**. Casa do Código, 2016.

PRENSKY, Marc. **Digital Natives, digital immigrants**. NCB University Press, Vol. 9 nº 5, 2001.

PROJECT BUILDER, **Software de Gerenciamento de Projetos**. Disponível em: <[www.projectbuilder.com.br](http://www.projectbuilder.com.br)>. Acesso em 01 jun. 2017.

PROJECT MANAGEMENT INSTITUTE, **Um Guia do Conhecimento em Gerenciamento de Projetos (GUIA PMBOK®)**. Quinta Edição, 2013.

REDAÇÃO *BotsBrasil*. **O que é um ChatBot?** Disponível em: <<https://medium.com/botsbrasil/o-que-%C3%A9-um-chatbot-7fa2897eac5d>>. Acesso em: 30 mar. 2017.

REDAÇÃO *O Futuro das Coisas*. **Gartner: em 2020 não haverá mais aplicativos**. Disponível em: <<http://ofuturodascoisas.com/gartner-em-2020-nao-havera-mais-aplicativos/>>. Acesso em: 30 mar. 2017.

RUSSELL, Stuart; NORVIG, Peter. **Inteligência Artificial**. Rio de Janeiro: CAMPUS, 2004.

TEIXEIRA, Sérgio; MENEZES, Crediné Silva de. **Facilitando o uso de Ambientes Virtuais através de Agentes de Conversação - Artigo**. XIV Simpósio Brasileiro de Informática na Educação – NCE – IM/UFRJ, 2003.

TURING, Alan Mathison. **Computing Machinery and Intelligence**. Oxford: Mind, LIX, 1950.

UNIPÊ CENTRO UNIVERSITÁRIO, **Mais informações**. Disponível em: <<https://ww.facebook.com/unipeoficial/>>. Acesso em: 10 abr. 2017.

UNIPÊ PÓS-GRADUAÇÃO, **Cursos**. Disponível em: <<http://unipe.br/pos-graduacao/>>. Acesso em: 10 abr. 2017.

UNIPÊ VIRTUAL, **GTI – Guia para Relatório Técnico Projeto Aplicado I 2017.1**. Disponível em: <<https://virtual.unipe.br/ead/mod/folder/view.php?id=161411>>. Acesso em: 02 abr. 2017.

WALLACE, Richard S. **The Anatomy of A.L.I.C.E.** Artificial Intelligence Foundation, Inc, 1995.

WEIZENBAUM, Joseph. **ELIZA: A Computer Program for the Study of Natural Language Communication Between Man and Machine**. Cambridge (MIT): Communications of the ACM, 1966.

WIKIPÉDIA, **Chatterbot**. Disponível em: <<https://pt.wikipedia.org/wiki/Chatterbot>>. Acesso em 03 abr. 2017.

## APÊNDICES

### TERMO DE ABERTURA DO PROJETO

Segundo o Guia PMBOK® Quinta Edição, o termo de abertura do projeto apresenta o resumo do cronograma de marcos e os requisitos de aprovação do projeto que influenciarão o gerenciamento do cronograma do projeto.

<b>Nome do Projeto</b>  Desenvolvimento de ChatBot para a Coordenação dos Cursos de Pós-Graduação da UNIPÊ	<b>Data</b>  20/03/2017
<b>Gerente do Projeto</b>  Antônio Leonardo Falcão Filho	<b>Contato</b>  (83) 99927 3978
<b>Professor Orientador</b>  Msc. Ricardo Roberto de Lima	

### JUSTIFICATIVA DO PROJETO

Os alunos dos cursos de pós-graduação da UNIPÊ atualmente carecem de um serviço eficaz de comunicação com as suas coordenações, fazendo-se necessário um projeto de mudança na comunicação destas coordenações com o seu público-fim.

Um serviço de ChatBot bem elaborado pode suprir esta carência, desafogando os profissionais dos cursos de pós-graduação da grande quantidade de informações que devem oferecer periodicamente aos alunos, pois a maior parte destas informações poderá ser respondida de forma automática pelo serviço.

### PRODUTOS E SERVIÇOS OFERECIDOS

O projeto oferecerá um serviço de ChatBot integrado a um aplicativo de mensagens, para utilização pelos alunos de pós-graduação de Universidade Particular de João Pessoa.

## ESCOPO PRETENDIDO

Pretende-se que o serviço desenvolvido se torne a principal fonte de informações e interações dos alunos de pós-graduação de Universidade Particular de João Pessoa com as suas coordenações.

## PREMISSAS DO PROJETO

- Existência de alunos com necessidade de aplicação mais efetiva nas comunicações com a coordenação;
- Inexistência de concorrência de um serviço de ChatBot específico para a Universidade;
- Disponibilidade dos aplicativos a serem utilizados no projeto (Word Online, FastFormat, REST etc.).

## RESTRIÇÕES DO PROJETO

- Utilização do S.O. Android;
- Utilização de API REST para acesso ao serviço de mensagens.
- Utilização da API.IA do Google para desenvolvimento do ChatBot.
- Utilização de Banco de Dados NOSQL.
- Utilização da Linguagem Python.
- Integração nas Redes Sociais tais como Telegram, Slack, Messenger.

## ACOMPANHAMENTO DO PROJETO

O projeto terá acompanhamento semanal pela equipe, professor orientador e professor da cadeira Projeto Aplicado I, e apresentações dos marcos, nas datas definidas na disciplina.

## ESTIMATIVA DE CUSTOS E PRAZOS

O prazo para entrega final do projeto é de 8 (oito) meses, considerando a data para a apresentação do TCC; estima-se um custo total de R\$ 300,00 (trezentos reais), relativo às mensalidades dos aplicativos utilizados, impressão e encadernação e utilização de internet.

## AUTORIZAÇÃO

Local e data:

João Pessoa, 20/03/2017

---

Assinatura do Professor Orientador



## **DECLARAÇÃO DE ESCOPO DO PROJETO**

De acordo com o Guia PMBOK® Quinta Edição, a Declaração de Escopo do Projeto é uma parte importante do seu projeto, pois descreve as entregas do projeto e o trabalho necessário para criá-las.

### **OBJETIVOS DO DOCUMENTO**

O objetivo deste documento é descrever, de forma clara e objetiva, qual trabalho deverá ser realizado e quais entregas serão produzidas.

### **JUSTIFICATIVA DO PROJETO**

No ano de 2015, uma pesquisa apontou que cerca de 84% das pessoas utiliza apenas cinco aplicativos mensalmente, mostrando que os usuários estão cada vez mais exigentes na demanda por serviços e necessidades de comunicação.

O desenvolvimento de um ChatBot para a pós-graduação da UNIPÊ, além de facilitar bastante a comunicação, abre mais um canal de interação da instituição com o seu público (interno e externo), podendo ser uma das principais maneiras da mesma interagir com o seus discentes, respondendo à maioria das demandas em tempo real.

Os diversos aplicativos que são utilizados pelos discentes não tem um serviço integrado para contato com a instituição, o que significa que é interessante e atrativo criar um serviço de mensagens que possa ser integrado com o aplicativo favorito deste usuário.

Percebe-se uma crescente exigência por entrega de informação de forma rápida e simplificada, e a falta de tempo faz com que, muitas vezes, alunos ou potenciais alunos não tenham condições de comparecer presencialmente ou telefonar para tirar dúvidas acerca de assuntos relacionados ao curso.

Verifica-se também, em determinados períodos do dia, um grande fluxo de atendimento presencial, ocasionando irritação pela demora no atendimento, desistência de potenciais alunos e perda de conteúdo dos alunos já matriculados; a criação de um ChatBot faz com que o atendimento se torne quase que totalmente automatizado, fazendo com que apenas assuntos de maior complexidade exijam atendimento humano.

## CRITÉRIOS DE SUCESSO

O projeto será considerado um sucesso se atender a todos os critérios de aceitação das entregas, respeitar as restrições e cumprir o cronograma de execução.

## ESCOPO DO PRODUTO

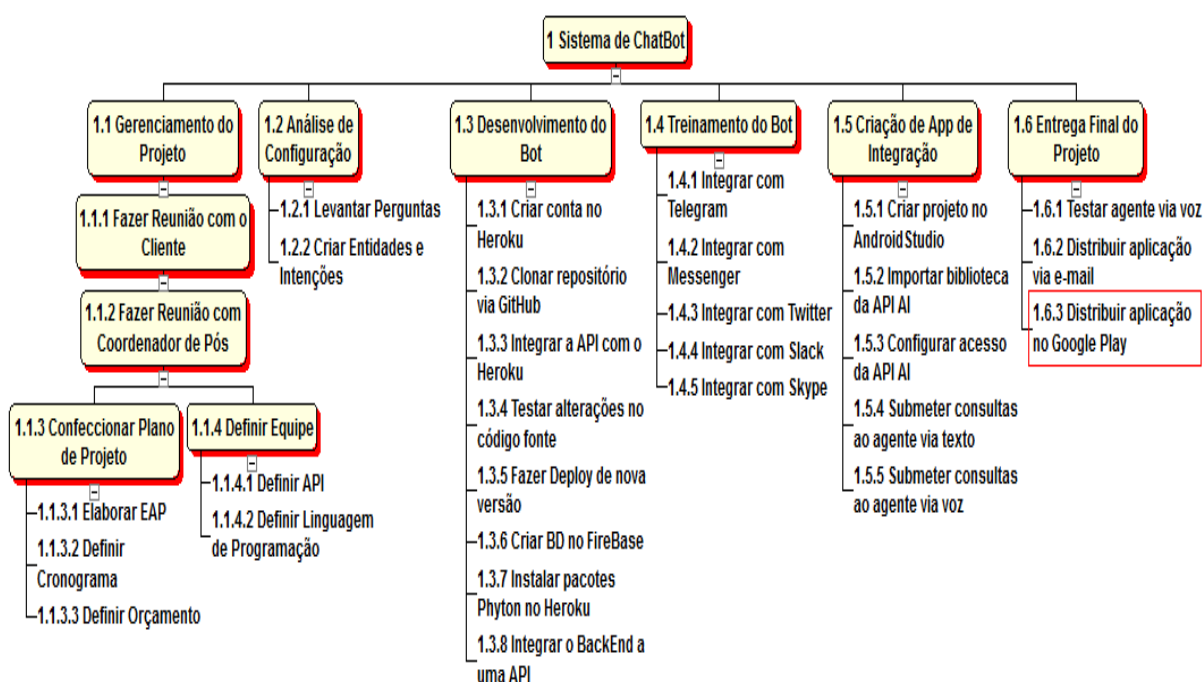
O produto deverá possibilitar ao usuário enviar mensagem para a coordenação de pós-graduação através do aplicativo de mensagens escolhido, e ter a resposta em tempo real.

## ESCOPO DO PROJETO

Work Breakdown Structure – WBS (Estrutura Analítica do Projeto – EAP)

De acordo com o Guia PMBOK® - Quinta Edição, a EAP é uma decomposição hierárquica orientada à entrega do trabalho a ser executado pela equipe para atingir os objetivos do projeto e criar as entregas requisitadas, com cada nível descendente da EAP representando uma definição gradualmente mais detalhada do trabalho do projeto.

Figura 53 – WBS Sistema de ChatBot



Fonte: Elaborado pelos Autores, 2017.

## DICIONÁRIO DA EAP

Segundo o Guia PMBOK® - Quinta Edição, O dicionário da EAP é um documento que fornece informações detalhadas sobre entregas, atividades e agendamento de cada componente da estrutura analítica do projeto (EAP).

Quadro 1 – Dicionário da EAP

(continua)

Dicionário da EAP - ChatBot					
EAP#	Tarefa	Descrição da tarefa	Produto de trabalho	Proprietário	Est. nível de esforço
1.1	Gerenciamento do projeto	Todas de gestão e gerenciamento.			
1.1.1	Fazer reunião com o cliente	Reunir-se com Joanna, para coletar informações técnicas sobre os cursos de pós graduação de direito.	Planejamento	Gerente de projeto	2 hrs
1.1.2	Fazer reunião com coordenador de pós	Reunir-se com Arthur, para coletar informações específicas dos cursos.	Planejamento	Gerente de projeto	2 hrs
1.1.3	Confeccionar Plano de Projeto	Desenvolver EAP, projeção de orçamento e tempo.	EAP, dicionário da EAP e Plano do Projeto	Gerente de projeto	8 hrs
1.1.3.1	Elaborar EAP	Fazer a EAP e o dicionário.	EAP, Dicionário da EAP	Gerente de projeto	4 hrs
1.1.3.2	Definir Cronograma	Elaborar o cronograma	Cronograma	Gerente de projeto	2 hrs
1.1.3.3	Definir Orçamento	Elaborar o orçamento	Orçamento	Gerente de projeto	2 hrs
1.1.4	Definir equipe	Definir pessoas que irão participar da equipe suas respectivas funções e trabalhos.	Planejamento, escolha de API, escolha de Linguagem	Gerente de projeto	2 hrs
1.1.4.1	Definir API	Definir a API que será utilizada no projeto.	Escolha da API	Gerente de projeto	1 hrs
1.1.4.2	Definir Linguagem de Programação	Definir qual será a linguagem utilizada no projeto.	Escolha da linguagem	Gerente de projeto	1 hrs
1.2	Análise de Configuração	Iniciar processo para criação das primeiras perguntas, e criação de Entidades e Intenções.	Análise	Gerente de projeto	40 hrs
1.2.1	Levantar perguntas	Levantar perguntas para alimentar o Bot.	Análise	Gerente de projeto	14 hrs
1.2.2	Criar Entidades e Intenções	Criar entidades e sinônimos, criar intenções e respostas.	Análise	Gerente de projeto	16 hrs

Quadro 1 – Dicionário da EAP

(continua)

Dicionário da EAP - ChatBot					
EAP#	Tarefa	Descrição da tarefa	Produto de trabalho	Proprietário	Est. nível de esforço
1.3	Desenvolvimento do Bot	Iniciar desenvolvimento do ChatBot.	Desenvolvimento	Gerente de projeto	9 hrs
1.3.1	Criar conta no Heroku	Criar uma conta no Heroku para iniciar desenvolvimento.	Desenvolvimento	Gerente de projeto	1 hrs
1.3.2	Clonar repositório via GitHub	Instalar Heroku CLI e clonar repositório via GitHub.	Desenvolvimento	Gerente de projeto	1 hrs
1.3.3	Integrar a API com o Heroku	Integra API escolhida na área de Webhook com o Heroku.	Desenvolvimento	Gerente de projeto	1 hrs
1.3.4	Testar alterações no código fonte	Testar alterações do backend utilizando Postman.	Desenvolvimento	Gerente de projeto	1 hrs
1.3.5	Fazer Deploy de nova versão	Fazer o Deploy da aplicação para testar alterações.	Desenvolvimento	Gerente de projeto	1 hrs
1.3.6	Criar BD no FireBase	Criar Banco de Dados para armazenar todas as informações referentes ao Bot.	Desenvolvimento	Gerente de projeto	2 hrs
1.3.7	Instalar pacotes Phyton no Heroku	Instalar o pacote Phyton FireBase no Heroku.	Desenvolvimento	Gerente de projeto	1 hrs
1.3.8	Integrar o BackEnd a uma API	Integrar backend a uma API via Json.	Desenvolvimento	Gerente de projeto	1 hrs
1.4	Treinamento do Bot	Treinar o Bot para ensiná-lo perguntas rotineiras.	Treinamento	Gerente de projeto	5 hrs
1.4.1	Integrar com Telegram	Integrar a API com o telegram utilizando BotFather.	Integração	Gerente de projeto	1 hrs
1.4.2	Integrar com Messenger	Integrar a API com o messenger utilizando Facebook Developers.	Integração	Gerente de projeto	1 hrs
1.4.3	Integrar com Twitter	Integrar a API com o Twitter utilizando tokens.	Integração	Gerente de projeto	1 hrs
1.4.4	Integrar com Slack	Integrar a API com o Slack utilizando Slack App.	Integração	Gerente de projeto	1 hrs
1.4.5	Integrar com Skype	Integrar a API com o Skype utilizando Microsoft App Developers.	Integração	Gerente de projeto	1 hrs
1.5	Criação de App de Integração	Criar aplicativo Android com interface simples, para se comunicar com a API.	Desenvolvimento	Gerente de projeto	10 hrs

Quadro 1 – Dicionário da EAP

(conclusão)

Dicionário da EAP - ChatBot					
EAP#	Tarefa	Descrição da tarefa	Produto de trabalho	Proprietário	Est. nível de esforço
1.5.1	Criar projeto no AndroidStudio	Criar projeto no AndroidStudio para iniciar o desenvolvimento do aplicativo.	Desenvolvimento	Gerente de projeto	3 hrs
1.5.2	Importar biblioteca da API AI	Importar biblioteca da API para o AndroidStudio.	Desenvolvimento	Gerente de projeto	3 hrs
1.5.3	Configurar acesso da API AI	Configurar acesso da API para se comunicar com o aplicativo Android.	Desenvolvimento	Gerente de projeto	2 hrs
1.5.4	Submeter consultas ao agente via texto	Fazer consultas via texto para testar o aplicativo.	Testes	Gerente de projeto	1 hrs
1.5.5	Submeter consultas ao agente via voz	Fazer consultas via voz para testar o aplicativo.	Testes	Gerente de projeto	1 hrs
1.6	Entrega Final do Projeto	Distribuir o Aplicativo.	Entrega	Gerente de projeto	3 hrs
1.6.1	Testar agente via voz	Fazer testes do Bot via voz.	Testes	Gerente de projeto	1 hrs
1.6.2	Distribuir aplicação via e-mail	Distribuir o Aplicativo aos interessados, via e-mail.	Entrega	Gerente de projeto	1 hrs
1.6.3	Distribuir aplicação no Google Play	Disponibilizar aplicativo no Google Play.	Entrega	Gerente de projeto	1 hrs

Fonte: Elaborado pelos Autores, 2017.

Tecnologias:

- DialogFlow
- Linguagem Python;
- Banco de Dados NoSQL Firebase;
- Json;
- API REST;
- Git Hub;
- Heroku;
- Postman;
- Linguagem Java;
- Sistema Operacional Windows;
- Facebook Developers para integrar com Messenger;

- Bot Father para integrar com Telegram;
- Microsoft APP para integrar com Skype;
- Slack APP para integrar com Slack;
- Actions on Google.

#### Exclusões do Projeto / Fora do Escopo:

Não houve exclusões no projeto. Não estava no escopo o treinamento para utilização e manejo da ferramenta, nem o gerenciamento de sua disponibilidade.

#### ENTREGAS E CRITÉRIOS DE ACEITAÇÃO

As entregas estão definidas no documento *Dicionário da EAP*, constante deste relatório; como critérios de aceitação, espera-se que a funcionalidade desenvolvida seja entregue em estágio de plena utilização e totalmente integrada com as aplicações de mensagens indicadas no projeto.

## ESTRUTURA DE DESENVOLVIMENTO

Este capítulo apresenta a estrutura de organização dos trabalhos a serem efetuados, os responsáveis pelos trabalhos, partes interessadas, além do plano de Gestão de Riscos e o Cronograma, detalhadamente descritos e registrados em itens e subitens.

### NECESSIDADES DO CLIENTE

Nesta seção se faz necessário destacar e reconhecer a motivação e alinhamento do projeto com o negócio do cliente.

- Os diversos aplicativos que são utilizados pelos discentes não tem um serviço integrado para contato com a instituição, o que significa que é interessante e atrativo criar um serviço de mensagens que possa ser integrado com o aplicativo favorito deste usuário.
- A criação de um ChatBot faz com que o atendimento se torne totalmente automatizado, fazendo com que apenas assuntos de maior complexidade precisem de um comparecimento presencial à instituição.

### DOCUMENTAÇÃO DOS REQUISITOS

Não há documentação de requisitos.

### ENTREGAS

As entregas deste projeto estão elencadas no documento *Estrutura Analítica do Projeto – EAP*, constante deste relatório.

### PARTES INTERESSADAS

As partes interessadas no projeto são aqueles que, de alguma forma, fazem parte do planejamento e execução do todo ou alguma parte dele, e aqueles que utilizarão o produto ou serviço desenvolvido, após a entrega final.

Nome	E-mail	Telefone
Antônio Leonardo	<a href="mailto:falcaosport@hotmail.com">falcaosport@hotmail.com</a>	(83) 9-9927-3978

Márcio Heriberto	<a href="mailto:marcioheriberto@gmail.com">marcioheriberto@gmail.com</a>	(83) 9-8704-3733
Gyan Lima	<a href="mailto:gyan.patrick1@gmail.com">gyan.patrick1@gmail.com</a>	(83) 9-8881-7338
Ricardo Roberto	<a href="mailto:ricardo.roberto@unipe.br">ricardo.roberto@unipe.br</a>	(83) 9-9107-8943
Arthur	<a href="mailto:pgpdireito@unipe.br">pgpdireito@unipe.br</a>	(83) 2106-9200

## MATRIZ DE RESPONSABILIDADES


### Papéis e Responsabilidades

Nome	Departamento	Papel	Responsabilidade
Antônio Leonardo	Estudante	Gerente do Projeto	Gerenciar o Projeto, desenvolver o ChatBot.
Márcio Heriberto	Estudante	Scrum	Gerenciar a equipe, desenvolver o ChatBot.
Gyan Lima	Estudante	Líder Time	Ficará responsável pela parte escrita do trabalho.
Ricardo Roberto	Educador	Professor/Orientador	Ficará responsável por auxiliar os alunos no desenvolvimento do trabalho, através de reuniões semanais.

## GESTÃO DE RISCOS (PMO)

De acordo com o ProjectBuilder ([www.projectbuilder.com.br](http://www.projectbuilder.com.br), 22/04/2017), “O Project Management Office, abreviado como PMO, é o departamento responsável por definir e manter os padrões de gerenciamento de projetos na empresa(...).”

O Escritório de Gerenciamento de Projetos, como é conhecido no Brasil, elabora, entre outros artefatos, o Plano de Gerenciamento de Riscos de projetos.

Plano de gerenciamento dos riscos	
Desenvolvimento de ChatBot	

Controle de Versões			
Versão	Data	Autor	Notas da Revisão
1	25/05/2017	Márcio Heriberto de Brito	-



## OBJETIVO DO PLANO DE GERENCIAMENTO DOS RISCOS

Gerenciar os riscos do projeto requer um Plano de Gerenciamento dos Riscos descrevendo como os processos de riscos serão estruturados e executados, iniciando pela identificação dos riscos, suas análises qualitativa e quantitativa, seu plano de respostas e concluindo com a forma que os riscos serão controlados e monitorados.

O Plano de gerenciamento dos riscos é desenvolvido e aprovado durante a fase de planejamento do projeto e é um plano auxiliar do Plano de gerenciamento do projeto. Tem como objetivo aumentar a probabilidade e o impacto dos eventos positivos, reduzir a probabilidade e o impacto dos eventos negativos no projeto e orientar a equipe do projeto sobre como os processos de riscos serão executados.

## GERENCIAMENTO DOS RISCOS

As seções abaixo identificam os componentes do Plano de Gerenciamento dos Riscos, de acordo com o guia PMO.

### *Processos de Riscos*

#### Identificar os riscos

Determinar quais riscos podem afetar o projeto e documentar suas características.

#### Realizar a análise qualitativa dos riscos

Avaliar a exposição ao risco para priorizar os riscos que serão objetos de análise ou ação adicional.

#### Realizar a análise quantitativa dos riscos

Efetuar a análise numérica do efeito dos riscos identificados nos objetivos gerais do projeto.

#### Planejar as respostas aos riscos

Desenvolver opções e ações para aumentar as oportunidades e reduzir as ameaças aos objetivos do projeto.

### Controlar os riscos

Monitorar e controlar os riscos durante o ciclo de vida do projeto.

#### ***Documentos Padronizados de Riscos***

Documento	Descrição	Template
<a href="#">Plano de gerenciamento dos riscos</a>	O Plano de Gerenciamento dos riscos tem como objetivo aumentar a probabilidade e o impacto dos eventos positivos, reduzir a probabilidade e o impacto dos eventos negativos no projeto e orientar a equipe do projeto sobre como os processos de riscos serão executados.	<a href="#">Plano de gerenciamento dos riscos.docx</a>
<a href="#">Registro dos riscos</a>	O registro dos riscos é iniciado no processo Identificar os riscos, e é atualizado conforme os outros processos de gerenciamento dos riscos (análise qualitativa, quantitativa, planejar as respostas aos riscos e monitorar e controlar os riscos) são conduzidos, resultando em um aumento no nível e no tipo de informações contidas no registro dos riscos ao longo do tempo.	<a href="#">Registro dos riscos.xlsx</a>

#### ***Responsabilidades dos Riscos***

Membro da Equipe	Responsabilidades
Gerente do Projeto	Certificar que os riscos foram identificados e tratados de modo a aumentar a probabilidade e o impacto dos eventos positivos, reduzir a probabilidade e o impacto dos eventos negativos no projeto. Monitorar os riscos conforme descrito neste plano. Divulgar informações pertinentes aos riscos do projeto
Scrum	Implantar durante o processo do desenvolvimento de software os controles necessários à Gestão dos Riscos.
Líder Time	Confeccionar os Templates exigidos para documentação do Plano de Gerenciamento dos Riscos.

#### ***Ferramentas Utilizadas***

Ferramenta	Descrição da aplicação	Quando aplicar	Responsável
Brainstorming	Utilizado para identificar os riscos do	No início do projeto e sempre que for	Gerente do Projeto

	projeto.	necessário revisar os riscos identificados.	
Plano de Contingência	Utilizado para resposta a riscos não identificados.	Quando algum risco não identificado ocorrer.	Gerente do Projeto
Avaliação da Probabilidade e Impacto	Utilizado para medir probabilidades e impactos dos riscos.	No início do projeto e sempre que for necessário revisar os riscos identificados.	Gerente do Projeto
Estratégias para riscos negativos ou ameaças	Utilizado para definir o que fazer caso ocorra riscos ou ameaças ao projeto.	No início do projeto e sempre que for necessário revisar os riscos identificados.	Gerente do Projeto

## IDENTIFICAÇÃO DOS RISCOS

Será utilizado o Brainstorming (tempestade de ideias) para identificação dos riscos do projeto. O Gerente do Projeto deverá compor uma equipe multidisciplinar para participar do Brainstorming, de modo que todas as áreas estejam bem representadas, e que os riscos principais do projeto sejam identificados.

## DEFINIÇÕES DE PROBABILIDADE E IMPACTO DOS RISCOS – ANÁLISE QUALITATIVA

### *Matriz de Probabilidade*

Probabilidade	% de certeza
1-Muito baixa	0 a 20%
2-Baixa	20 a 40%
3-Média	40 a 60%
4-Alta	60 a 80%
5-Muito Alta	> 80%

### *Matriz de Impacto*

Impacto
1-Muito baixo
2-Baixo
3-Médio

4-Alto
5-Muito Alto

O impacto irá variar de acordo com a área impactada. Quando um risco impactar mais de uma área, deverá ser usada a área mais impactada.

Abaixo temos a orientação sobre como classificar o impacto:

	<b>Muito baixo (Nota = 1)</b>	<b>Baixo (Nota = 2)</b>	<b>Médio (Nota = 3)</b>	<b>Alto (Nota = 4)</b>	<b>Muito alto (Nota = 5)</b>
<b>Custo</b>	Até 2% no orçamento	De 2 a 5% no orçamento	De 5 a 8% no orçamento	De 8 a 10% no orçamento	Acima de 10% no orçamento
<b>Tempo</b>	Até 2% no prazo total	De 2 a 5% no prazo	De 5 a 8% no prazo	De 8 a 10% no prazo	Acima de 10% no prazo
<b>Escopo</b>	Sem impacto	Mudança impactará no custo	Mudança impactará no custo e no tempo	Mudança impactará no custo, tempo e qualidade	Mudança total de escopo

O grau de risco está definido na matriz de probabilidade x impacto ( $G = I \times P$ ), demonstrado abaixo:

Probabilidade						
5	5	10	15	20	25	
4	4	8	12	16	20	
3	3	6	9	12	15	
2	2	4	6	8	10	
1	1	2	3	4	5	
Impacto	1	2	3	4	5	

Graus de risco elencados: Vermelho = elevado; Amarelo = médio; Verde = baixo.

## REALIZAR A ANÁLISE QUANTITATIVA DOS RISCOS

A análise quantitativa dos riscos a princípio não será utilizada, motivada pela percepção da pequena abrangência do projeto, que será local. Caso se decida utilizá-la no decorrer do projeto, será formado comitê específico para esta finalidade.

## PLANEJAR AS RESPOSTAS AOS RISCOS

### *Reserva de Contingência*

Para os riscos não identificados e os identificados e tratados via reserva de contingência, o orçamento e prazo originais serão acrescidos em 10%. O Comitê do Projeto deverá aprovar a utilização da reserva em reunião extraordinária.

### *Estratégias para Riscos Negativos ou Ameaças*

<b>Estratégia</b>	<b>Descrição</b>	<b>Exemplo</b>
<b><i>Eliminar</i></b>	Remover a probabilidade de ocorrência.	Cancelar o projeto, ou a parte com elevado grau de risco.
<b><i>Transferir</i></b>	Transferir o impacto (total ou parcialmente) para terceiro(s).	Contratar um seguro.
<b><i>Mitigar</i></b>	Reduzir a probabilidade de ocorrência.	Promover redundância de recursos.
<b><i>Aceitar</i></b>	Utilizar plano de contingência, caso o evento ocorra.	-

### *Estratégias para Riscos Positivos ou Oportunidades*

<b>Estratégia</b>	<b>Descrição</b>
<b><i>Explorar</i></b>	Garantir que a oportunidade ocorra para explorar seus benefícios.
<b><i>Compartilhar</i></b>	Transferir total ou parcial a propriedade da oportunidade para um terceiro que tem maior capacidade de explorá-la.
<b><i>Melhorar</i></b>	Aumentar probabilidade e/ou impacto de uma oportunidade.
<b><i>Aceitar</i></b>	Tirar proveito caso a oportunidade ocorra.

## CONTROLAR OS RISCOS

O Gerente do Projeto e os responsáveis, conforme elencado na matriz de responsabilidade, devem acompanhar os riscos identificados, monitorar os riscos residuais, identificar novos riscos, executar os planos de respostas a riscos e avaliar sua eficácia durante todo o ciclo de vida do projeto.

O gerente de projeto executa o que foi planejado na análise de riscos e controla os riscos novos identificados durante a execução do projeto.

Este processo consiste em:

- Identificar, analisar, e planejar para riscos novos;
- Monitorar os riscos identificados;
- Analisar novamente os riscos existentes de acordo com as mudanças de contexto;
- Monitorar condições para ativar planos de contingência;
- Monitorar riscos residuais;
- Rever a execução do plano de respostas aos riscos para avaliar sua eficácia;
- Determina se as premissas do projeto ainda são válidas;
- Determinar se as políticas e os procedimentos de gestão de risco estão sendo seguidas;
- Determinar se as reservas de contingência de custo e prazo devem ser modificadas com os riscos do projeto.

<b>Aprovações</b>		
<b>Participante</b>	<b>Assinatura</b>	<b>Data</b>
Patrocinador do Projeto		
Gerente do Projeto		