

**UNIVERSIDADE TECNOLÓGICA FEDERAL DO PARANÁ
CÂMPUS CORNÉLIO PROCÓPIO
DIRETORIA DE GRADUAÇÃO E EDUCAÇÃO PROFISSIONAL
DEPARTAMENTO DE COMPUTAÇÃO
ENGENHARIA DE COMPUTAÇÃO**

RAFAEL SIAN DE FREITAS

**UTILIZAÇÃO DE BOTS PARA RECUPERAÇÃO DE DADOS EM ÓRGÃOS
PÚBLICOS**

TRABALHO DE CONCLUSÃO DE CURSO

**CORNÉLIO PROCÓPIO
2017**

RAFAEL SIAN DE FREITAS

**UTILIZAÇÃO DE BOTS PARA RECUPERAÇÃO DE DADOS EM ÓRGÃOS
PÚBLICOS**

Trabalho de Conclusão de Curso apresentada ao Departamento de Computação da Universidade Tecnológica Federal do Paraná como requisito parcial para obtenção do título de “Bacharel em Engenharia de Computação”.

Orientador: Prof. Dr. Diego Roberto Antunes

**CORNÉLIO PROCÓPIO
2017**



TERMO DE APROVAÇÃO

Utilização de bots para recuperação de dados em órgãos públicos

por

Rafael Sian de Freitas

Este Trabalho de Conclusão de Curso foi julgado adequado para obtenção do Título de “Bacharel em Engenharia de Computação” e aprovado em sua forma final pelo Departamento de Computação da Universidade Tecnológica Federal do Paraná.
Cornélio Procópio, 14/06/2017.

Antonio Carlos Fernandes Da Silva, Prof. Dr.
Coordenador do Curso

Banca Examinadora:

Diego Roberto Antunes, Prof. Dr.
Orientador

Willian Massami Watanabe, Prof. Dr.
Universidade Tecnológica Federal do Paraná

Andre Takeshi Endo, Prof. Dr
Universidade Tecnológica Federal do Paraná

AGRADECIMENTOS

Agradeço de forma especial a todos os meus familiares, especialmente aos meus pais, avós, irmã e sobrinha que sempre me apoiaram desde o meu primeiro dia como estudante de graduação. Durante essa jornada, o apoio dessas pessoas queridas foi essencial em diversos momentos.

A minha namorada, Gabriela, uma pessoa extremamente paciente com todos as dificuldades encontradas, presente em todos os momentos e que sempre esteve me aconselhando e trazendo conforto para fazer eu chegar até aqui.

Aos meus amigos e colegas que passaram pela minha vida durante esses cinco anos, enriquecendo a minha experiência e caráter, principalmente como ser humano.

A todos os meus professores, tanto da vida acadêmica quanto do ensino básico, que foram capazes de transmitir os seus conhecimentos, experiências, afetividade e inúmeras qualidades que ajudaram na formação do que eu sou hoje.

Ao meu orientador, Prof. Dr. Diego R. Antunes pela enorme dedicação, disponibilidade e ajuda durante os últimos semestres da minha graduação, pelas inúmeras correções de texto e auxílio com os problemas enfrentados na minha vida profissional.

RESUMO

FREITAS, Rafael S.. **Utilização de bots para recuperação de dados em órgãos públicos.** 2017. 53 f. Trabalho de Conclusão de Curso – Engenharia de Computação, Universidade Tecnológica Federal do Paraná. Cornélio Procópio, 2017.

As informações e documentos em sites de órgãos públicos devem possibilitar um fácil acesso para todos os usuários. Porém, a grande quantidade de informações e documentos que possuem um nome semelhante ou até mesmo um contexto parecido acaba atrapalhando o usuário a encontrar o que realmente deseja. Além disso, sites institucionais e governamentais possuem uma alta densidade de informação, criando diversos setores e menus que podem dificultar ao usuário encontrar a informação. Com base nisso, foram realizadas duas pesquisas através de um formulário eletrônico com o intuito de verificar se o site da UTFPR apresenta problemas para encontrar informações através do sistema de busca disponibilizado no site. Os resultados das pesquisas mostraram que mais da metade dos usuários estão insatisfeitos com o sistema de busca e vários deles já enfrentaram problemas para encontrar informações e documentos no site. O principal intuito deste trabalho foi investigar a utilização de *chatterbots* como uma ferramenta de busca alternativa para auxiliar os usuários do site da UTFPR a encontrar informações e documentos que estão contidos no repositório da instituição.

Palavras-chave: *Chatterbot, bot, aplicação web, assistente inteligente*

ABSTRACT

FREITAS, Rafael S.. **Bots application for data retrieval in public agencies.** 2017. 53 p. Completion of Coursework – Computer Engineering, Federal University of Technology - Paraná. Cornélio Procópio, 2017.

The information and files in public entities websites should, easily, be available for all its users. However, the large amount of information and files with similar name or even content, muddle the users trying to find what them really want. Furthermore, institutions and government websites possess a high information density, creating several sections and menus which can make it difficult for the user to find information. With that in mind, two surveys were done using an electronic form with the intention to verify when the UTFPR website showed problems to find information using its own search engine. The surveys results showed that more than a half of the users are not satisfied with the website's search engine and several of them faced some problems to find information and files in the website. The main goal of this work is to investigate the usage of chatterbots as a search tool alternative to help UTFPR website users to find informations and files contained in the institution's repositories.

Keywords: Chatterbot, bot, web application, virtual assistant

LISTA DE FIGURAS

FIGURA 1 – Marcando um compromisso.	17
FIGURA 2 – Solicitando compromissos.	17
FIGURA 3 – Escolhendo uma categoria.	18
FIGURA 4 – Escolhendo uma subcategoria.	18
FIGURA 5 – Solicitando algo para comer.	19
FIGURA 6 – Conversando com o Assistant.	19
FIGURA 7 – Funcionamento das plataformas de linguagem natural.	20
FIGURA 8 – Criação do agente no Api.ai.	22
FIGURA 9 – Entidades de um agente no Api.ai.	22
FIGURA 10 – Intent de um agente no Api.ai.	22
FIGURA 11 – Pesquisa 1 - Questão 1 (Vínculo com a UTFPR).	26
FIGURA 12 – Pesquisa 1 - Questão 2 (Onde a informação necessária é procurada).	27
FIGURA 13 – Pesquisa 1 - Questão 3 (Grau de satisfação com o sistema de busca).	27
FIGURA 14 – Pesquisa 1 - Infográfico da análise das questões 2 e 3.	28
FIGURA 15 – Pesquisa 1 - Questão 4 (Problemas encontrados no sistema de busca).	29
FIGURA 16 – Pesquisa 2 - Questão 1 (Grau de satisfação com o sistema de busca).	30
FIGURA 17 – Pesquisa 2 - Questão 2 (Dúvidas solucionadas através do site).	30
FIGURA 18 – Pesquisa 2 - Questão 4 (O que poderia melhorar).	31
FIGURA 19 – Cadastrando o Abono de Faltas.	33
FIGURA 20 – Cadastrando a Entidade Professores.	34
FIGURA 21 – Cadastrando a <i>Intent</i> de Email.	35
FIGURA 22 – Painel Administrativo para cadastrar Professor.	36
FIGURA 23 – Aplicação Web Responsiva.	37
FIGURA 24 – Trecho do código do Webhook.	38
FIGURA 25 – Trecho do código do <i>Crawler</i> .	38
FIGURA 26 – Arquitetura Geral.	39
FIGURA 27 – Sistema de Gestão de Certificados Eletrônicos UTFPR	41
FIGURA 28 – Requisitando um certificado.	41
FIGURA 29 – Conseguindo o certificado.	41
FIGURA 30 – Cardápio no Sistemas UTFPR-CP	42
FIGURA 31 – Requisitando o cardápio.	42
FIGURA 32 – Programas disponíveis na UTFPR.	43
FIGURA 33 – Programa de monitoria.	43
FIGURA 34 – Exame de Suficiência.	45
FIGURA 35 – Abono de Faltas.	45
FIGURA 36 – Convalidação.	46
FIGURA 37 – Trancamento de Curso.	46
FIGURA 38 – Cancelamento de Matrícula.	47
FIGURA 39 – Emissão de Certificados.	48
FIGURA 40 – Informações sobre Monitoria.	49
FIGURA 41 – Confusões visuais da página da Monitoria.	49

SUMÁRIO

1	INTRODUÇÃO	9
1.1	MOTIVAÇÃO	10
1.2	OBJETIVO	10
1.3	ORGANIZAÇÃO DO TEXTO	11
2	FUNDAMENTAÇÃO TEÓRICA	12
2.1	INTERAÇÃO HUMANO-COMPUTADOR	12
2.2	INTERAÇÃO NATURAL	12
2.3	Bot	13
2.3.1	Chatterbot	13
2.4	TRABALHOS RELACIONADOS	15
2.4.1	Doroty	15
2.4.2	Persona-AIML	16
2.4.3	Maria	16
2.4.4	Jarvis	17
2.4.5	Babun	18
2.4.6	Google Assistant	19
3	TECNOLOGIAS	20
3.1	PLATAFORMAS DE LINGUAGEM NATURAL	20
3.1.1	Wit.ai	20
3.1.2	Api.ai	21
3.2	FUNCIONAMENTO DAS PLATAFORMAS DE LINGUAGEM NATURAL	21
3.3	APLICAÇÃO WEB	23
3.3.1	Webhook	24
3.3.2	RESTful	24
3.3.3	Firebase	24
4	DESENVOLVIMENTO	25
4.1	ELICITAÇÃO DE REQUISITOS	25
4.1.1	Primeira pesquisa	26
4.1.2	Segunda pesquisa	29
4.2	MATERIAIS E MÉTODOS	31
4.3	CHATTERBOT	32
4.3.1	Configuração do Chatterbot	32
4.3.2	Aplicação Web	36
4.3.3	Arquitetura Geral	37
4.3.4	Funcionalidades	39
4.3.4.1	Exame de Suficiência e Convalidação de Matérias	40
4.3.4.2	Trancamento de Curso, Cancelamento de Matrícula, Segunda Chamada, Turma sem Presença, Jubilamento e Abono de Faltas	40
4.3.4.3	Emissão de Certificados	40
4.3.4.4	Consulta ao Cardápio do Dia	40
4.3.4.5	Informações sobre Professores	42
4.3.4.6	Programas	42
5	RESULTADOS E DISCUSSÕES	44
5.1	EXAME DE SUFICIÊNCIA, ABONO DE FALTAS, CONVALIDAÇÃO, TRANCAMENTO DE CURSO E CANCELAMENTO DE MATRÍCULA	44

5.2	EMISSÃO DE CERTIFICADOS	47
5.3	INFORMAÇÕES SOBRE MONITORIA	48
6	CONSIDERAÇÕES FINAIS	50
6.1	LIMITAÇÕES	50
6.2	TRABALHOS FUTUROS	51
	REFERÊNCIAS	52

1 INTRODUÇÃO

Devido à melhoria exponencial do hardware de computadores e de dispositivos móveis tornou-se mais simples e barato desenvolver softwares relacionados à inteligência artificial e ao aprendizado de máquina, tais como os chatterbots, que estão cada vez mais presentes em nosso cotidiano. Esses agentes inteligentes que eram geralmente restritos às aplicações e pesquisas acadêmicas, agora estão sendo utilizados para diversas atividades que comumente eram exercidas de forma manual pelo homem (LEONHARDT, 2005).

Por meio da linguagem natural, os *chatterbots* tornam a interação com os usuários mais fácil, tendo a sua utilização tanto para o meio acadêmico quanto para aplicações comerciais, tornando esses robôs capazes de explorar uma relação social com os humanos (PREECE; ROGERS; SHARP, 2005; PILASTRI; BREGA, 2009; GALVÃO, 2003).

Atualmente o uso comercial dos *chatterbots* está cada vez maior, por exemplo, eles podem ser vistos em *Frequently Asked Questions* (FAQ), comércio eletrônico, ensino a distância, entretenimento, entre vários outros setores onde é necessária a comunicação de dois ou mais interlocutores através de texto ou fala (GALVÃO, 2003; PILASTRI; BREGA, 2009; LEONHARDT, 2005).

Devido à inclusão digital, a quantidade de dados produzidos na internet aumenta a cada dia e com isso vários órgãos públicos apresentam uma quantidade muito grande de documentos e informações com nomes semelhantes, porém, com datas ou contextos diferentes. Com isso, vários usuários não conseguem encontrar e acessar essas informações com facilidade.

Por exemplo, a Universidade Tecnológica Federal do Paraná (UTFPR) (estudo de caso apresentado neste trabalho) possui esse problema que também afeta outros órgãos públicos no Brasil, tais como: documentos relacionados aos cursos de graduação que têm informações semelhantes entre si.

Atualmente a UTFPR possui 13 Câmpus espalhados no estado do Paraná, aumentando consideravelmente o número de informações no site da instituição. Além disso, uma característica comum de sites de instituições públicas consiste na categorização de atividades por setores, o que gera menus de navegação muito extensos e com inúmeros níveis de navegação. Este é mais um exemplo de problema que pode dificultar um acesso rápido à informação pelo usuário.

A fim de averiguar o impacto destes problemas com os usuários, foram realizadas duas pesquisas por meio do questionário eletrônico do Google chamado Google Forms¹, com o objetivo de identificar os problemas mais comuns com o acesso a documentos e informações no site da instituição.

Esta primeira pesquisa foi respondida por 58 pessoas (docentes, discentes e técnicos),

¹www.forms.google.com

dentre as quais 50% responderam que possuem um baixo grau de satisfação com sistema de busca do site da UTFPR. Além disso, mais da metade desse grupo respondeu que não utiliza o sistema de busca disponível no site e vários usuários responderam que não conseguem encontrar as informações e documentos que necessitam, sendo muitas vezes necessário utilizar outros meios para encontrá-los.

Com base nos dados coletados e nas informações apresentadas no Capítulo 4 sobre a pesquisa, definiu-se como hipótese a utilização de *chatterbots* como um mecanismo de busca alternativo que pode minimizar os problemas de busca relatados e fornecer uma interação mais natural entre os usuários e o site da instituição.

1.1 MOTIVAÇÃO

Os *chatterbots* utilizam recursos de Inteligência Artificial (IA) para fornecer um mecanismo de interação em linguagem natural aos usuários, tornando a interação e a comunicação com o sistema mais simples e comum. Além disso, os usuários de sistemas *online* comumente utilizam *chats* e sistemas de mensagens instantâneas para comunicação em diversos contextos, tornando a implantação deste tipo de tecnologia mais simples aos usuários (PILASTRI; BREGA, 2009; DAHLBERG, 1978).

Conforme relatado brevemente na introdução deste trabalho, vários usuários do site da UTFPR possuem uma frustração quando utilizam o sistema de busca disponível para encontrar informações e arquivos, sendo necessário recorrer a outros meios de busca para conseguir encontrar a informação ou documento que deseja.

O mecanismo de busca existente no site da UTFPR, bem como a quantidade e a organização disponível no acervo dificultam a busca pelos usuários. Sendo assim, o intuito deste trabalho consistiu em investigar o problema e desenvolver uma solução alternativa para resolvê-lo. Desta forma, pode-se analisar a nova proposta de interação e, a partir dos resultados deste estudo de caso, testar a aplicação do recurso desenvolvido em outros setores e contextos da instituição, bem como outras instituições e setores governamentais.

1.2 OBJETIVO

Este trabalho teve como objetivo desenvolver um *chatterbot* para a comunidade da UTFPR do Câmpus Cornélio Procópio e explorar sua utilização como um sistema de busca alternativo visando facilitar a pesquisa de dados e informações pelos usuários no site na instituição.

1.3 ORGANIZAÇÃO DO TEXTO

Este texto está organizado em 6 capítulos, os quais estão distribuídos da seguinte maneira: o Capítulo 1 apresenta uma introdução sobre o cenário atual dos problemas que vários órgãos públicos estão enfrentando e também uma breve introdução sobre os *chatterbots*, seguido da motivação e objetivos do trabalho; o Capítulo 2 contém uma revisão sobre os conceitos necessários para o entendimento do trabalho; o Capítulo 3 apresenta as tecnologias utilizadas em todo o trabalho; o Capítulo 4 possui toda a descrição do desenvolvimento, juntamente com os materiais e métodos utilizados; o Capítulo 5 contém toda a análise dos resultados obtidos ao longo do trabalho; e no Capítulo 6 é apresentado as considerações finais, limitações e trabalhos futuros de todo o trabalho desenvolvido.

2 FUNDAMENTAÇÃO TEÓRICA

Neste capítulo é apresentada a fundamentação teórica utilizada como base para o desenvolvimento deste trabalho. O capítulo foi dividido nas seguintes seções: Interação Humano-Computador, Interação Natural, Bot e Trabalhos Relacionados.

2.1 INTERAÇÃO HUMANO-COMPUTADOR

Para Hewett et al. Interação Humano-Computador (IHC), é uma área que estuda o projeto, implementação e avaliação de um sistema computacional interativo para o uso do homem, junto com os fenômenos relacionados ao seu uso.

Uma aplicação da IHC em um sistema computacional interativo pode oferecer benefícios para os usuários do sistema, tais como: contribuir com a produtividade, satisfação e bem-estar do usuário. Essa área pode aproveitar características humanas e o poder da computação para projetar sistemas com uma boa interação, podendo assim utilizar dos seus benefícios (BARBOSA; SILVA, 2010).

Os objetos da interação atuam como uma ponte para ligar o homem ao computador, sendo por meio deles que o usuário consegue interagir com o computador. Muitos destes artefatos de interação são modelados computacionalmente como metáforas de objetos existentes no mundo real, por exemplo, o uso de gestos e linguagem natural para realizar ações. Os principais estilos de interação são: menus; janelas; formulários; linha de comando; interação natural (CYBIS; BETIOL; FAUST, 2010; PREECE; ROGERS; SHARP, 2005).

2.2 INTERAÇÃO NATURAL

Para Preece, Rogers e Sharp (2005) o desenvolvimento de sistemas que possuem diferentes tipos de socialização entre homem-máquina deve ser natural, sendo que esses tipos de sistemas devem existir pois os humanos são extremamente sociáveis. Um sistema que utiliza uma interação natural mais próxima do homem geralmente fornece uma maior usabilidade.

Uma vantagem da utilização do estilo de interação baseado em linguagem natural consiste na proximidade ao usuário, pois não há a necessidade do aprendizado do fluxo e da forma de interação uma vez que a interação é mediada pela escrita ou fala. Já os outros estilos citados (e.g. menus), necessitam que o usuário aprenda como utilizar e qual o fluxo de navegação, aumentando a curva de aprendizado da interação (PREECE; ROGERS; SHARP, 2005; GNECCO et al., 2012).

2.3 BOT

Bot é uma simplificação da palavra inglesa *Robot*, que na tradução livre significa Robô. Essa palavra possui uma origem tcheca devida a uma peça de Karel Capek. Nessa peça existia um robô humanoide que possuía a habilidade de realizar tarefas feitas pelo homem (PRIMO et al., 1998; CAPEK, 1921).

Primo et al. (1998) definem *bot* como sendo um agente que seja capaz de simular alguma atividade humana para fins de uma máquina ou usuário.

Segundo Wooldridge e Jennings (1995) o termo agente é utilizado amplamente na área da IA e isso pode causar alguns enganos, já que cada autor define esse termo conforme o escopo do seu trabalho. Para Wooldridge e Jennings (1995), o termo agente pode ser definido como um sistema baseado em software ou hardware o qual deve considerar as seguintes características: autonomia, capacidade social, reatividade e pró-atividade.

Um agente também precisa ser capaz de perceber o ambiente em que se encontra por meio de algumas entradas e agir através delas sobre esse mesmo ambiente, para apresentar saídas coerentes (LEONHARDT, 2005; RUSSELL; NORVIG, 1995).

Coppin (2010) também define um agente como sendo uma entidade que realiza tarefas para o auxílio do homem. Esses agentes inteligentes podem ser de vários tipos, como biológicos, robóticos ou de software. Além disso, Coppin (2010) chama de agentes reativos aqueles que reagem aos eventos do ambiente em que se encontram, definindo ações baseado naquele ambiente por meio do processamento das entradas para a geração de uma saída.

Russell e Norvig (1995) atentam ao fato de que a primeira etapa do projeto de um agente é especificar o ambiente em que ele atuará. Essa especificação deve ser a mais completa possível, especificando um conjunto de percepções, ações, objetivos e ambiente. Esse conjunto de especificações é conhecido como Perceptions, Actions, Goals, Environment (PAGE).

2.3.1 Chatterbot

Chatterbot, ou *chatbot*, é um tipo de *bot* cuja principal finalidade é responder aos usuários de uma forma natural, tornando a interação mais próxima à comunicação com um ser humano. A conversação entre usuário e *chatterbot* ocorre por meio de um recurso de interação natural, sendo assim, o *chatterbot* tem que ser capaz de receber uma entrada em linguagem natural e processar uma base de conhecimento para poder retornar algum conteúdo ou valor para o usuário (PRIMO et al., 1998; TEIXEIRA et al., 2005).

Para Leonhardt (2005), o *chatterbot* pode ser considerado, na maioria das vezes, como um agente conversacional, uma vez que ele pode apresentar um comportamento além do verbal, simulando gestos, expressões e processamento de linguagem natural, podendo assim aumentar

o entendimento da mensagem para o usuário.

Galvão (2003) utiliza o conjunto PAGE, para classificar alguns tipos de *chatterbots*. Segundo o autor, essa prática de classificação não é comum para quem trabalha na área, pois as percepções e ações podem se tornar obsoletas, já que na maioria dos casos elas se resumem respectivamente à interpretação de texto escrito e à produção de texto. Na tabela 1 pode ser visto a classificação de *chatterbots* utilizando o conjunto PAGE.

Tabela 1 – Classificação de *chatterbots* utilizando o conjunto PAGE

Aplicação	Percepções	Ações	Objetivos	Ambiente
Entretenimento	Texto escrito	Conversar sobre um tema qualquer, guiar o usuário em um jogo, etc.	Divertir o usuário geralmente simulando "vida artificial"	Usuários
Ensino a distância	Texto escrito	Sugerir assuntos de discussão, repreender alunos, tirar dúvidas, efetuar demonstrações, etc.	Melhorar a aprendizagem do estudante	Estudantes
Atendimento ao consumidor	Texto escrito	Responder dúvidas, anotar reclamações, prestar informações sobre processos em andamento, etc.	Minimizar custos de atendimento	Consumidores
Comércio eletrônico	Texto escrito	Sugerir a compra de produtos, guiar o comprador, etc.	Maximizar as vendas e satisfazer o cliente	Compradores

Fonte: Galvão (2003)

Para o correto funcionamento dos *chatterbots* e sua contínua melhoria são necessários o processamento de linguagem natural e o aprendizado de máquina. As linguagens naturais em sua maioria apresentam várias ambiguidades, sendo muito diferente das linguagens utilizadas pelos computadores (COPPIN, 2010; RUSSELL; NORVIG, 1995).

A fim de possibilitar uma interação natural entre usuário e sistemas computacionais, surgiu a área de Processamento de Linguagem Natural (PLN). O PLN estuda as técnicas para que agentes computacionais possam entender a linguagem natural criada pelos homens. Além dessas técnicas de compreensão, é necessário um conhecimento do domínio específico onde o agente estará situado (COPPIN, 2010).

Neste sentido, muitas plataformas de linguagem natural, utilizadas por *chatterbots*, tais como as apresentadas na Seção 3.1, utilizam mecanismos de PLN para reduzir ambiguidades

e buscar a correção da entrada fornecida pelo usuário a fim de aumentar a precisão no seu funcionamento.

O Aprendizado de Máquina, ou mais conhecido pelo seu nome em inglês *Machine Learning*, é uma área da IA na qual um agente adquire informações automaticamente sem a necessidade de programá-lo para esta finalidade (COPPIN, 2010; RUSSELL; NORVIG, 1995).

Russell e Norvig (1995) citam três principais motivos para que um agente aprenda para melhorar o seu desempenho. O primeiro motivo, o qual seria mais complicado resolver escrevendo código puro é que em algumas aplicações os projetistas não conseguem prever todas as possíveis situações que agente enfrentará. Em segundo, também não é possível saber se haverá mudanças no seu ambiente. O último motivo é que certas aplicações possuem uma abstração tão grande que os programadores não conseguem resolver sem os algoritmos de aprendizagem de máquina.

2.4 TRABALHOS RELACIONADOS

Nesta seção são apresentados os trabalhos relacionados com o tema proposto deste texto. Os trabalhos relacionados foram contextualizados a partir de aplicações comerciais e trabalhos ou pesquisas acadêmicas que utilizam a tecnologia dos *chatterbots*.

2.4.1 Dorothy

O trabalho de Leonhardt (2005) consistiu em adaptar o *chatterbot Artificial Linguistic Internet Computer Entity* (ALICE)¹ com o intuito de auxiliar o treinamento e gerência de redes para profissionais com pouco conhecimento na área, utilizando o protocolo Simple Network Management Protocol (SNMP)².

O *chatterbot* ALICE, o qual Leonhardt (2005) utilizou como base para o Dorothy, foi construído através de uma variação da linguagem de marcação *eXtensible Markup Language* (XML)³, o *Artificial Intelligence Markup Language* (AIML)⁴. ALICE possui uma ampla base de conhecimento constituída por mais de 5000 palavras além de alguns módulos que podem ser acrescentados para auxiliar a classificação de algumas expressões.

O *chatterbot* Dorothy, construído no trabalho de Leonhardt (2005) possui um alto grau de semelhança com a proposta deste texto, com o objetivo de auxiliar usuários através de um serviço que possui uma fácil interação utilizando uma linguagem natural.

¹<http://www.alicebot.org/about.html>

²<http://www snmp.com/protocol>

³<https://www.w3.org/XML>

⁴<http://www.alicebot.org/aiml.html>

ALICE possui algumas semelhanças com a plataforma de linguagem natural que será utilizada neste trabalho, o Api.ai:

- Ambas plataformas são abertas, qualquer usuário pode criar o seu próprio *chatterbot*;
- Não é necessário um conhecimento muito profundo em programação para utilizar nenhum dos dois;
- As duas plataformas permitem que sejam adicionados conhecimentos para respostas em ocasiões específicas.

2.4.2 Persona-AIML

Nesse trabalho, Galvão (2003) propõe uma extensão do AIML com intuito de adicionar personalidade à linguagem. Galvão (2003) utilizou uma arquitetura para colocar personalidade a partir dos seguintes elementos: traços, atitudes, humor, emoções e estados físicos. O maior problema enfrentado para colocar a personalidade no AIML foi que os *chatterbots* construídos nessa linguagem não conseguem armazenar dados ou informações de uma mensagem anterior. Ou seja, se o interlocutor ofender o robô, ele não conseguiria lembrar-se disso na próxima interação.

A motivação para a criação do Persona-AIML foi baseada em estudos que demonstram a importância de explorar o comportamento do usuário, sendo necessário adicionar um recurso de personalidade no *chatterbot*.

2.4.3 Maria

O *chatterbot* Maria, desenvolvido por Rothermel e Domingues (2007), também foi construído para auxiliar usuários em algumas tarefas, mais especificamente auxiliar alunos de um curso de Administração em uma disciplina. A motivação para a criação desse *chatterbot* foi o pouco recurso disponível para o ensino de administração no ensino superior.

Esse *chatterbot* utiliza a linguagem AIML. Segundo Rothermel e Domingues (2007) esse *chatterbot* possui aproximadamente seiscentas perguntas que são acompanhadas com as respectivas respostas. O ponto fraco de Maria é que foi necessário adicionar várias perguntas com diferentes maneiras de serem feitas, por esse motivo a quantidade de perguntas é muito grande.

Além das aplicações acadêmicas, este trabalho revisou algumas aplicações comerciais. A pesquisa e a utilização dessas aplicações mostram-se importantes para poder avaliar os conceitos explicados anteriormente. O principal conceito abordado nas aplicações comerciais analisadas, consiste no ambiente, ou seja, sobre o contexto no qual o agente atua. As três

primeiras aplicações atuam somente em um ambiente específico para o qual foi desenvolvido e a última possui um ambiente muito amplo já que se trata de um assistente virtual.

2.4.4 Jarvis

O Jarvis foi construído para lembrar de compromissos. Ele está disponível para conversas na sua página do Facebook⁵ através da plataforma do Messenger⁶ e atualmente entende somente o idioma Inglês.

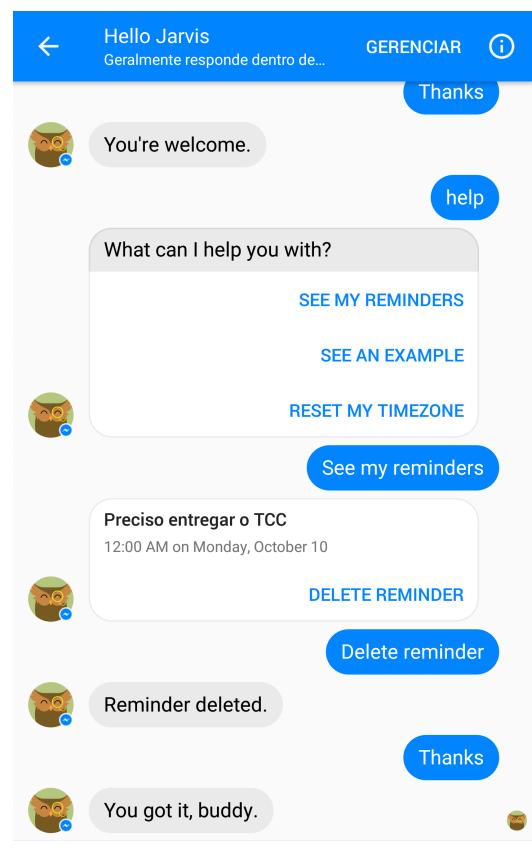
Esse *chatterbot* possui uma fácil interação com o usuário, funcionando através de comandos simples. Os testes realizados no Jarvis mostraram-se satisfatórios para o principal propósito: marcar compromissos para serem notificados. Nas Figuras 1 e 2 pode ser visto uma conversa com o Jarvis.



Escreva uma mensagem...

Figura 1 – Marcando um compromisso.

Fonte: Autoria própria.



Escreva uma mensagem...

Figura 2 – Solicitando compromissos.

Fonte: Autoria própria.

⁵<https://www.facebook.com/helloimjarvis>

⁶<https://www.messenger.com>

2.4.5 Babun

Babun é um *chatterbot* construído para auxiliar *startups* para ajudar na área da produtividade, gerência e marketing. O Babun está disponível na sua própria página do Facebook⁷ funcionando através da plataforma Messenger. Atualmente esse *chatterbot* só funciona no idioma inglês.

Durante a conversação com o Babun, um usuário consegue selecionar a categoria que necessita de ajuda e com base nisso o *chatterbot* envia alguns serviços (produtos externos) que se encaixam na categoria escolhida. Além disso, é possível enviar um produto para o *chatterbot* com a finalidade de cadastrá-lo em alguma subcategoria de um produto. As Figuras 3 e 4 ilustram a conversação entre um usuário e o Babun utilizando o recurso de categorias e serviços externos.

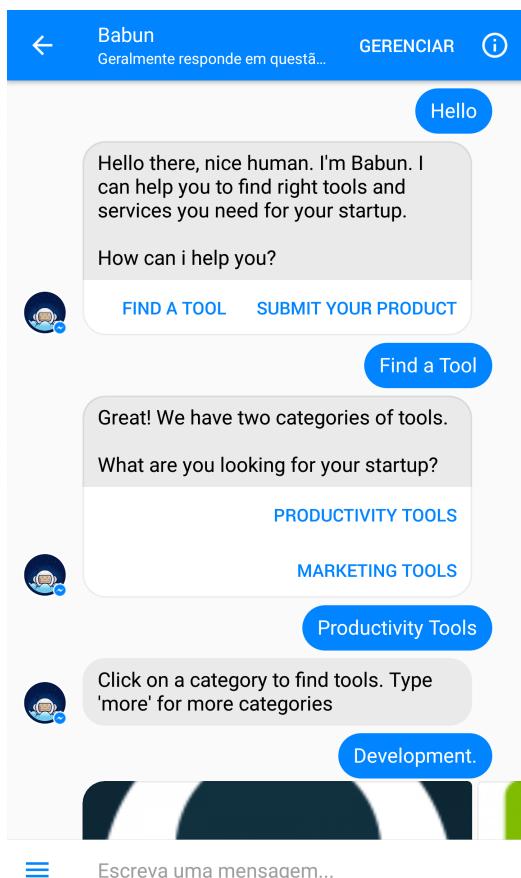


Figura 3 – Escolhendo uma categoria.

Fonte: Autoria própria.

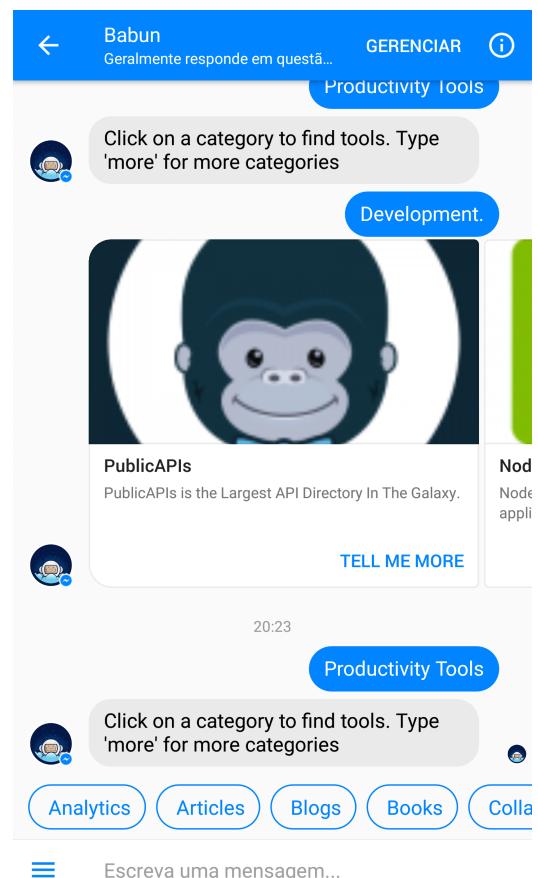


Figura 4 – Escolhendo uma subcategoria.

Fonte: Autoria própria.

⁷<https://www.facebook.com/bigbabun>

2.4.6 Google Assistant

O *chatterbot* Google Assistant está funcionando em uma edição beta através do aplicativo de mensagens instantâneas Google Allo⁸, disponível para dispositivos móveis.

O Google Assistant foi desenvolvido com a proposta de ser um assistente virtual para os usuários do Allo. A interação pode ser feita através de um chat exclusivo, o qual funciona como um assistente virtual, ou ele também pode ser adicionado em uma conversa com outra pessoa, com o intuito de buscar informações na Internet. As Figuras 5 e 6 apresentam um exemplo de funcionamento desse assistente virtual do Google.

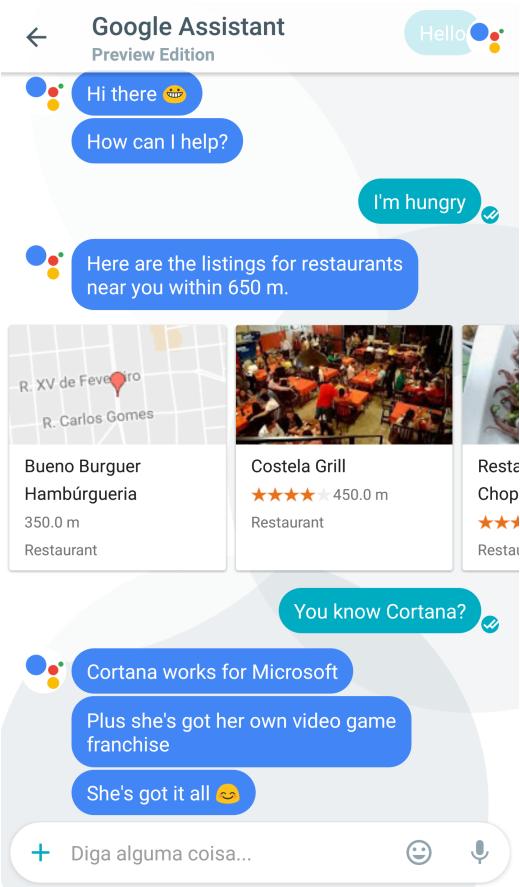


Figura 5 – Solicitando algo para comer.

Fonte: Autoria própria.

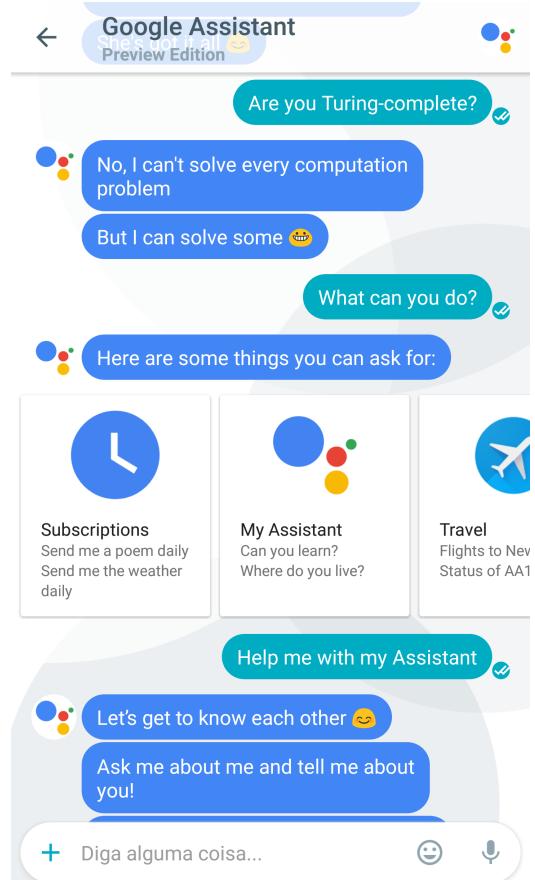


Figura 6 – Conversando com o Assistant.

Fonte: Autoria própria.

⁸<https://allo.google.com>

3 TECNOLOGIAS

Neste capítulo são apresentadas as tecnologias utilizadas neste trabalho.

3.1 PLATAFORMAS DE LINGUAGEM NATURAL

Nesta seção são apresentadas duas plataformas de linguagem natural: Wit.ai e Api.ai. Essas duas plataformas possuem a finalidade de oferecer um serviço capaz de transformar sentenças de uma frase, através de texto ou voz, em dados estruturados. Esse funcionamento pode ser visto na Figura 7.

Essas plataformas utilizam técnicas de Aprendizado de Máquina, pelas quais cada interação realizada pelo usuário é utilizada pelo *chatterbot* para aprender mais sobre como agir em diversas situações (WIT.AI, 2016; API.AI, 2016).

As duas plataformas foram desenvolvidas para serem utilizadas para os serviços de: *chatterbots*, dispositivos vestíveis, aplicativos móveis, serviços para automóveis, automatização residencial, robôs e aplicativos para televisões inteligentes (WIT.AI, 2016; API.AI, 2016).

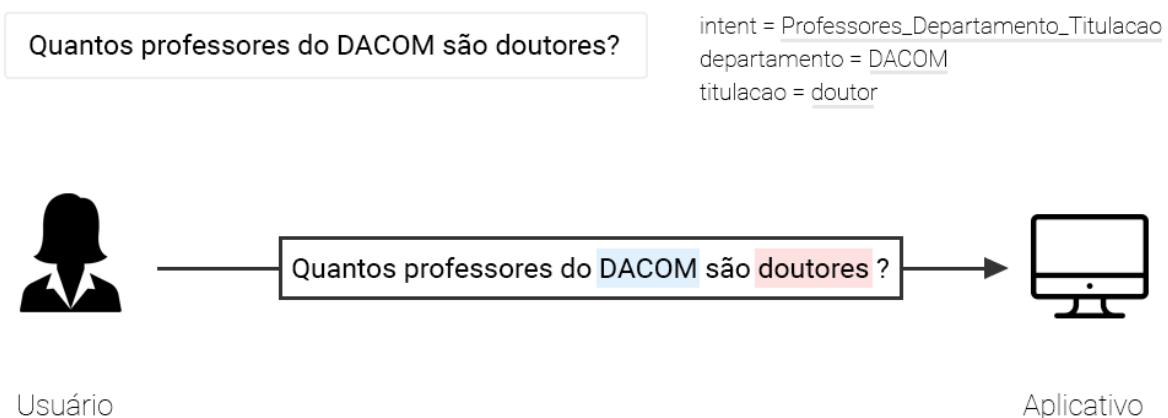


Figura 7 – Funcionamento das plataformas de linguagem natural.

Fonte: Autoria própria.

3.1.1 Wit.ai

O Wit.ai é uma plataforma de linguagem natural pertencente ao Facebook. Essa plataforma é totalmente gratuita e não possui nenhum plano comercial. Atualmente o Wit.ai possui suporte para 50 idiomas, dentre eles o Português do Brasil. Além disso, não existe limite de requisições na utilização dos serviços disponíveis. O Wit.ai possui três SDKs¹ para

¹Um SDK, também conhecido como Kit de Desenvolvimento de Software, é um conjunto de ferramentas que permite os desenvolvedores a compilar um código fonte para uma plataforma específica.

os desenvolvedores utilizarem na integração da sua aplicação com a plataforma. Também está disponível a utilização dos serviços da plataforma através do estilo arquitetural RESTful (WIT.AI, 2016).

3.1.2 Api.ai

A plataforma de linguagem natural Api.ai foi adquirida recentemente pelo Google². Diferentemente do Wit.ai, essa plataforma possui planos comerciais para a utilização dos serviços. A principal diferença nos planos pagos são os limites de requisições que podem ser feitas aos seus serviços. O Português do Brasil também está entre os idiomas suportados pela plataforma (API.AI, 2016).

O Api.ai oferece uma integração de um clique para as principais aplicações de mensagens instantâneas, assistentes virtuais e plataformas relacionadas à Internet das Coisas (IoT). Além disso, também são oferecidos SDKs para mais de quinze plataformas de programação. Assim como o Wit.ai, também é permitido fazer requisições para a utilização dos serviços da plataforma através do RESTful (API.AI, 2016)

3.2 FUNCIONAMENTO DAS PLATAFORMAS DE LINGUAGEM NATURAL

Ambas as plataformas de linguagem natural já apresentadas tem o seu funcionamento parecido. Os principais passos para a utilização dessas plataformas são: criação do agente, criação de entidades, criação dos *intents*, testar e treinar o agente e por último, integrar o agente em alguma aplicação (API.AI, 2016; WIT.AI, 2016).

A criação do agente consiste no primeiro passo para a utilização dessas plataformas em uma aplicação. Nesse passo será possível escolher algumas informações, tais como: nome do agente, descrição, fuso horário padrão, dados de um *bot* que você já criou anteriormente e o idioma padrão (API.AI, 2016; WIT.AI, 2016). A criação do agente pode ser vista na Figura 8.

As entidades nessas plataformas são objetos específicos que são utilizados no mapeamento das frases pelo agente. Na criação de entidade é necessário cadastrar um nome junto com vários mapeamentos para esse objeto, juntamente com os sinônimos que a palavra pode ter, conforme pode ser visto na Figura 9 (API.AI, 2016; WIT.AI, 2016).

Na criação dos *intents* é necessário cadastrar entradas e saídas para uma determinada requisição. As entradas são cadastradas com base no que o usuário pode escrever ou falar e as saídas serão as respostas que o agente responderá. Na Figura 10 pode ser observado o cadastro de um *Intent* novo na plataforma do Api.ai (API.AI, 2016; WIT.AI, 2016).

²<https://api.ai/blog/2016/09/19/api-ai-joining-google/>

Agent name

DESCRIPTION
Describe your agent

ADD SAMPLE DATA
Select sample data to be loaded into new agent or leave unselected for empty agent

LANGUAGE ⓘ
English

DEFAULT TIME ZONE
(GMT-3:00) America/Buenos_Aires

Figura 8 – Criação do agente no Api.ai.**Fonte:** Autoria própria.

DACOM	DACOM, Departamento Acadêmico de Computação, Departamento de Computação, DACOMP, Computação
DAELE	DAELE, Departamento Acadêmico de Elétrica, Departamento de Elétrica, Elétrica
DAMEC	DAMEC, Departamento Acadêmico de Mecânica, Departamento de Mecânica, Mecânica
DAMAT	DAMAT, Departamento Acadêmico de Matemática, Departamento de Matemática, Matemática
DACIN	DACIN, Departamento Acadêmico De Ciências Da Natureza, Departamento de Ciências da Natureza
DACHS	DACHS, Departamento Acadêmico De Ciências Humanas E Sociais, Departamento De Ciências Humanas e Sociais

Figura 9 – Entidades de um agente no Api.ai.**Fonte:** Autoria própria.

Contexts

User says

Add user expression

Search in user says

- Qual o cardápio do dia
- cardápio do dia
- consulta ao cardápio do dia
- cardápio
- Cardápio de hoje
- Gostaria do cardápio de hoje
- Cardápio Restaurante Universitário
- Cardápio RU

Figura 10 – Intent de um agente no Api.ai.**Fonte:** Autoria própria.

Para o treinamento do agente é necessário informar as entradas (texto ou voz) para poder verificar se a saída (resposta) do agente está correta. A integração do agente em alguma aplicação dependerá sobre qual a plataforma o aplicativo funcionará (API.AI, 2016; WIT.AI, 2016).

Uma das possíveis aplicações onde um agente do Api.ai pode ser criado, é em uma aplicação web, onde as requisições para o Api.ai podem ser realizadas através do RESTful. Essa aplicação web pode ser desenvolvida utilizando várias tecnologias e linguagens de programação suportadas pelos servidores e navegadores.

3.3 APLICAÇÃO WEB

Aplicações Web são softwares executados diretamente no navegador do cliente, sem a necessidade da instalação de outra ferramenta ou software. Um site pode ser composto por várias linguagens, como o HTML, CSS, JavaScript, entre outros (FERREIRA et al., 2015).

O avanço da tecnologia também trouxe alguns tipos de adaptações dos sites para diversas plataformas, como pode ser visto com as Aplicações Web. Uma Aplicação Web é um site que permite a utilização do mesmo através do navegador em um dispositivo móvel (tablets e smartphones), podendo assim oferecer um conteúdo com uma melhor interação para esse dispositivo. Um dos principais pontos importantes de uma Aplicação Web é possuir uma facilidade maior de desenvolvimento comparado com aplicações nativas e híbridas (PRESSMAN; MAXIM, 2016; MAHMUD; ABDULLAH, 2014; KARADIMCE; BOGATINOSKA, 2014).

Uma das linguagens utilizadas para a construção de Aplicações Web consiste no JavaScript, uma linguagem de programação interpretada, com tipagem dinâmica e orientação a objetos que possui uma execução diretamente no lado do cliente. As páginas web costumam ser estáticas se elas forem construídas somente com HTML. Quando uma página web utiliza o JavaScript ela é capaz de adicionar eventos específicos dessa linguagem, tornando essa página estática em dinâmica (FLANAGAN, 2006).

Atualmente existem vários *frameworks* de desenvolvimento para JavaScript, que possuem o intuito de facilitar, otimizar e melhorar o desenvolvimento de *scripts* para essa linguagem. Um desses *frameworks* é o AngularJS, que implementa várias funcionalidades a sua Aplicação Web (ANGULARJS, 2017).

Outra linguagem que também é muito utilizada em Aplicações Web é o PHP. Essa é executada totalmente do lado do servidor. Sendo assim, quando um cliente faz uma requisição para uma página em PHP, o servidor executa todos os códigos necessários e retorna um texto que pode estar em diferentes formatos, como o HTML, JSON ou XML (NIEDERAUER, 2004).

3.3.1 Webhook

Um Webhook é uma forma de prover informações para clientes web quando um evento ocorre a partir de dados recebidos de uma outra aplicação. Com isso, esses dados são processados e validados para que o Webhook possa retornar uma resposta de forma formatada. (RUPPEN; PASQUIER; HÜRLIMANN, 2011; SENDGRID, 2017).

Um Webhook é necessário quando algum evento em uma aplicação é disparado e é necessário o processamento de alguns dados. Neste trabalho, o Webhook foi utilizado para processar algumas informações que o Api.ai necessita. Além disso, um Webhook pode ser requisitado utilizando o RESTful.

3.3.2 RESTful

O Transferência de Estado Representacional (REST) é um estilo arquitetural híbrido para sistemas hipermídias derivado de outras arquiteturas baseadas em rede. O REST foi desenvolvido para apresentar um modelo de como a Web deveria funcionar na atualidade (FIELDING, 2000).

O estilo arquitetural REST apresenta algumas restrições, nas quais se forem utilizadas corretamente podem oferecer várias vantagens, tais como: escalabilidade, generalização de interfaces e desenvolvimento de componentes independentes. Além disso, o REST permite que os desenvolvedores não fiquem focados somente na implementação do sistema e sim nos recursos (FIELDING, 2000; RODRIGUEZ, 2008).

RESTful consiste em um formato para *web services* que utiliza os métodos definidos pelo protocolo Hypertext Transfer Protocol (HTTP) para seguir as orientações do estilo arquitetural REST. Com isso, é permitido que sistemas acessem dados na Web sem a necessidade criar novos protocolos específicos da aplicação (RODRIGUEZ, 2008).

3.3.3 Firebase

O Firebase é uma plataforma que possui vários serviços para aplicações web e aplicativos. Um desses serviços é o *Realtime Database*, um Banco de Dados NoSQL que possui o funcionamento em tempo real. Os dados armazenados nessa ferramenta ficam disponíveis na nuvem, possuindo um rápida sincronização de todos os aplicativos conectados (FIREBASE, 2017).

Outra vantagem da utilização do Firebase é a qual não é necessária instalação de nenhum software, tanto por parte do cliente quanto para o desenvolvedor. O serviço de armazenamento de dados funciona diretamente nos servidores do Firebase (FIREBASE, 2017).

4 DESENVOLVIMENTO

Este trabalho utilizou a UTFPR como estudo de caso durante o desenvolvimento. A motivação da escolha dessa instituição foi principalmente pelos constantes problemas relatados por vários usuários do site, conforme apresentado nas próximas seções a partir da análise das pesquisas realizadas com docentes, discentes e técnicos-administrativos.

Neste capítulo são apresentadas as descrições das atividades que foram desenvolvidas para alcançar os objetivos deste trabalho.

4.1 ELICITAÇÃO DE REQUISITOS

Para Preece, Rogers e Sharp (2005), os desenvolvedores devem assegurar que o produto desenvolvido atende às necessidades dos usuários. Quando os usuários são devidamente escutados, o produto final tende a atender às expectativas e necessidades de todos. Portanto, a elicitação de requisitos mostrou-se necessária para o desenvolvimento deste trabalho, pois foi possível entender vários aspectos sobre os problemas enfrentados pelos usuários, além de auxiliar na definição do escopo e na viabilidade de implementação.

Visto a necessidade da elicitação de requisitos através das opiniões dos usuários foi necessário estabelecer como esses dados seriam coletados. Preece, Rogers e Sharp (2005) explicam que um conjunto de requisitos estáveis deve possuir informações relevantes, apropriadas e suficientes. Com base nisso, os dados foram coletados a partir da utilização de questionários eletrônicos enviados aos usuários e entrevistas com grupos de usuários.

A elicitação de requisitos para este trabalho consistiu em duas partes. A primeira consistiu na realização de uma pesquisa por meio do Google Forms¹ para consultar a satisfação, a opinião e a sugestão dos usuários em relação ao sistema de busca e navegação de informações e arquivos no site da UTFPR. Esse formulário de pesquisa foi divulgado através de docentes, grupos da UTFPR em redes sociais e por meio de listas de e-mails. O público alvo dessa pesquisa foi o grupo de usuários que utilizam ou já utilizaram o site da UTFPR para encontrar alguma informação.

A segunda parte da elicitação de requisitos consistiu na realização de uma entrevista, utilizando o Google Forms, com algumas pessoas de setores da universidade que possuem um contato direto com os usuários do site, onde a grande maioria desses usuários são alunos da instituição. A segunda pesquisa foi criada para complementar a primeira, a fim de levantar as principais questões que os setores da UTFPR respondem aos usuários quando a informação não é encontrada no site. As principais pessoas que responderam essa segunda parte foram

¹www.forms.google.com

coordenadores de curso e de trabalho de conclusão, chefe de departamento do curso de Engenharia de Computação, coordenador de curso de Analise e Desenvolvimento de Sistemas e técnicos administrativos da secretaria e órgãos de apoio ao aluno, como o Departamento de Registros Acadêmicos (DERAC).

No final das duas partes foram obtidas 64 respostas, sendo que 58 foram da primeira pesquisa e 6 da segunda. Ambas as partes da elição de requisitos através das pesquisas são analisadas e explicadas nas subseções a seguir.

4.1.1 Primeira pesquisa

Conforme citado anteriormente, o público alvo desta pesquisa consistiu no perfil das pessoas que utilizam ou já utilizaram o site da UTFPR para encontrar informação ou documento. Esse questionário foi composto por cinco questões, sendo elas: três com respostas de múltiplas escolhas e duas de respostas abertas. Além disso, as quatro primeiras questões eram obrigatórias e a última opcional.

A primeira questão no questionário foi para saber qual o vínculo do respondente com a instituição. Na Figura 11 pode ser visto que 77,6% dos respondentes são alunos ou ex-alunos, 20,7% são professores e apenas 1,7% são técnicos administrativos.

Qual o seu vínculo com a UTFPR? (58 respostas)

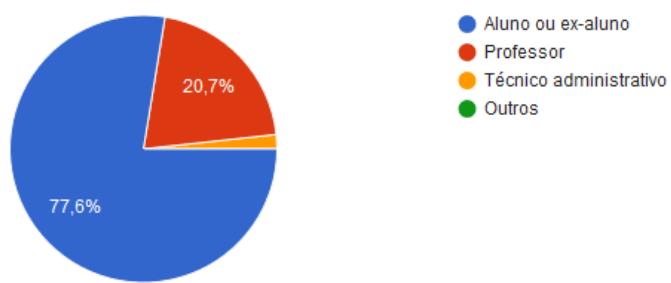


Figura 11 – Pesquisa 1 - Questão 1 (Vínculo com a UTFPR).

Fonte: Autoria própria.

Na segunda questão foi perguntado onde o respondente procura as informações que necessita sobre a UTFPR. Aproximadamente 40% procuram no site da instituição, um número relativamente baixo já que todas as informações estão no repositório do site. Os outros 60% dos respondentes procuram em meios alternativos, sendo eles 31% no Google, 17,2% nas secretarias ou órgãos específicos dentro da instituição, 6,9% em grupos relacionados a UTFPR

no Facebook e 5,2% em outros lugares. Na Figura 12 pode ser visto um gráfico das respostas da segunda questão.

Quando você necessita de algum documento/informação sobre algo da UTFPR, onde você procura?
 (58 respostas)

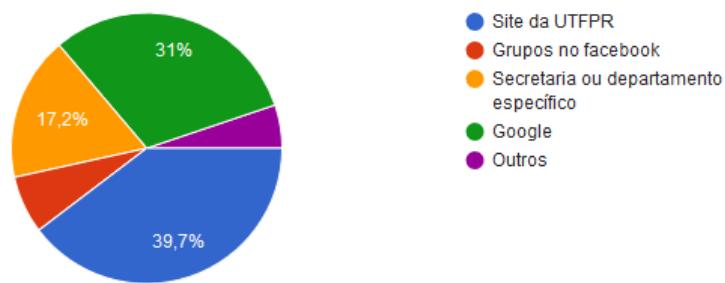


Figura 12 – Pesquisa 1 - Questão 2 (Onde a informação necessária é procurada).

Fonte: Autoria própria.

A terceira questão foi elaborada para saber o grau de satisfação em relação ao sistema de busca disponível no site da UTFPR. Aproximadamente 63% dos respondentes estão insatisfeitos com o sistema de busca, sendo que menos de 10% estão satisfeitos com esse sistema. A Figura 13 apresenta um gráfico do grau de satisfação com o sistema de busca.

Qual o seu grau de satisfação com o sistema de Busca no site da UTFPR?
 (58 respostas)

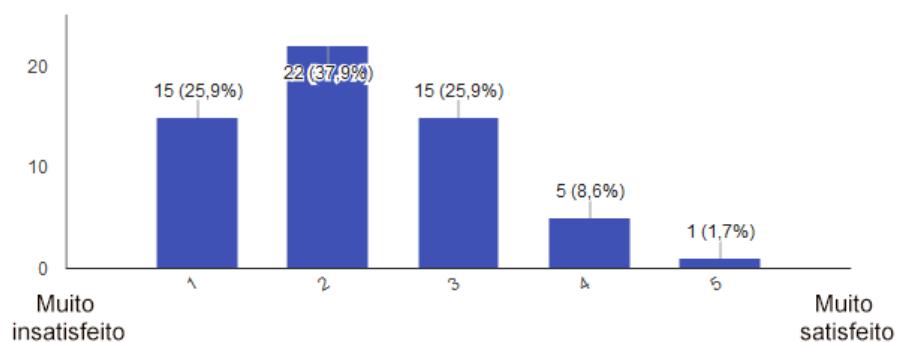


Figura 13 – Pesquisa 1 - Questão 3 (Grau de satisfação com o sistema de busca).

Fonte: Autoria própria.

Após a análise da segunda e terceira questão fez-se necessária uma análise mais profunda para entender o motivo de vários respondentes estarem procurando as informações em meios alternativos ao sistema de busca disponível, sendo que todas as informações estão

contidas no repositório da universidade. Com isso, foram analisadas todas as respostas das pessoas que declararam um baixo grau de satisfação com o sistema de busca, com o intuito de saber quem são os usuários e onde eles buscam as informações.

Analizando o Infográfico da Figura 14 pode ser visto que os usuários insatisfeitos já encontraram algum problema para encontrar alguma informação no sistema de busca. Talvez esses problemas enfrentados levaram os 67% dos usuários a utilizarem sistemas de buscas alternativos para encontrar o que precisavam.

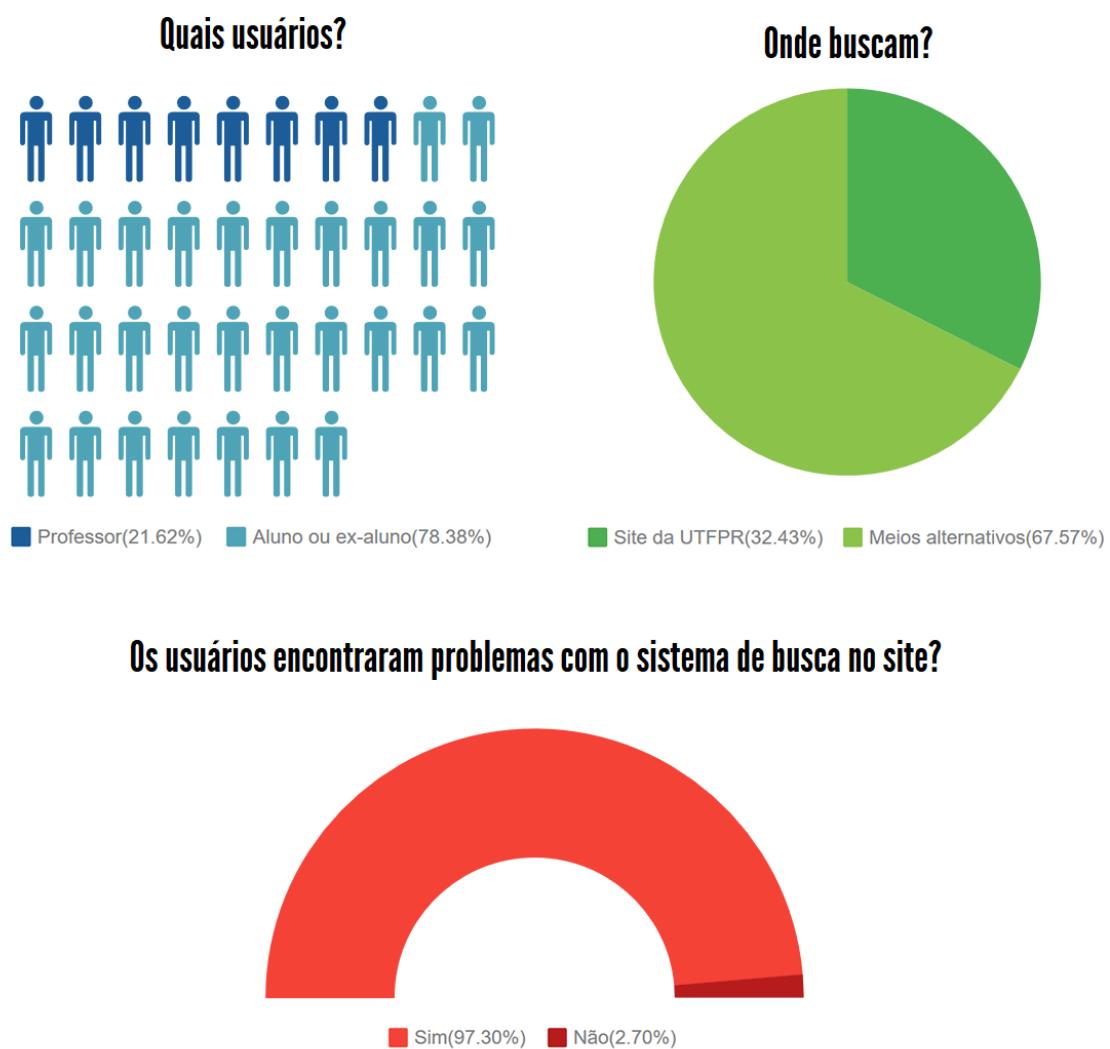


Figura 14 – Pesquisa 1 - Infográfico da análise das questões 2 e 3.

Fonte: Autoria própria.

Na quarta questão o respondente era questionado se já enfrentou algum problema para encontrar alguma informação ou documento no site da UTFPR e perguntava qual o tipo de problema. Por essa questão ser aberta, os usuários relataram diversos problemas, criando uma base de conhecimento inicial a ser utilizada para o treinamento do sistema de linguagem natural.

Com base neste conjunto de respostas, a Figura 15 apresenta um percentual de pessoas que reportaram problemas ao utilizar o recurso de busca disponível no site.

Você já encontrou problemas para achar algum documento/informação no site da UTFPR? Por favor, descreva qual documento/informação e o problema.

(58 respostas)

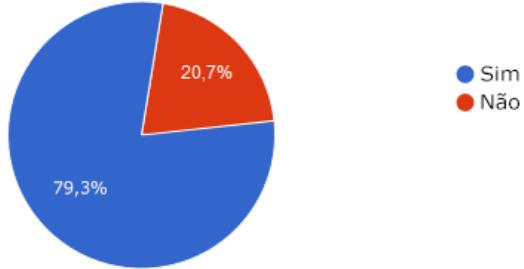


Figura 15 – Pesquisa 1 - Questão 4 (Problemas encontrados no sistema de busca).

Fonte: Autoria própria.

A última pergunta foi elaborada para complementar a base de conhecimento necessária para o treinamento do *chatterbot*. O respondente foi questionado sobre quais informações e/ou documentos gostaria de acessar rapidamente através de um sistema. Nesta questão, opcional e de resposta aberta, foram obtidas 48 respostas sendo que, a maioria das respostas mostraram-se favoráveis para a questão de possuir um acesso rápido as informações e documentos.

4.1.2 Segunda pesquisa

Conforme explicado, a segunda pesquisa teve como objetivo complementar a primeira pesquisa para obter informações adicionais a partir de setores que interagem com diversos usuários no cotidiano. As pessoas que realizaram essa pesquisa foram escolhidas baseadas no setor no qual trabalham e também por sempre serem questionadas sobre dúvidas relacionadas às informações e documentos da instituição.

A primeira questão desse questionário tinha o intuito de saber o grau de satisfação com o sistema de busca disponível no site da UTFPR. Todas as respostas mostraram que os respondentes estão insatisfeitos com esse sistema de busca. Na Figura 16 pode ser visto o gráfico das respostas. Um dos fatores que levam a esta insatisfação é a falta de um mecanismo que retorna as informações requeridas pelos usuários com maior velocidade. Este é um requisito essencial que impacta no desempenho de chefias e coordenações nas tarefas diárias.

Na segunda questão foi perguntado se as dúvidas dos alunos que procuram os respondentes podem ser solucionadas através do site da universidade. Apenas 20% dos respondentes

Como você classificaria o sistema de busca disponível no site da UTFPR?
 (6 respostas)

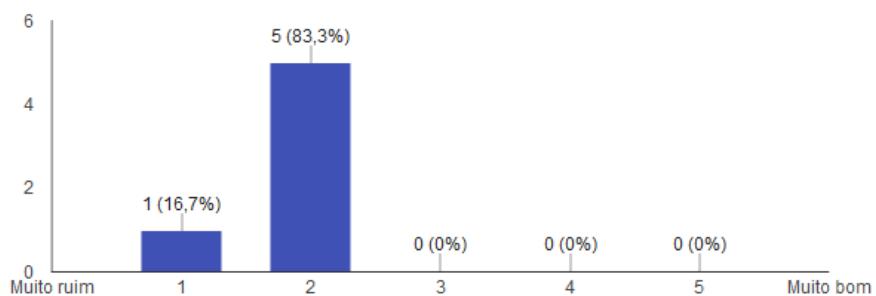


Figura 16 – Pesquisa 2 - Questão 1 (Grau de satisfação com o sistema de busca).

Fonte: Autoria própria.

afirmaram que as dúvidas podem ser solucionadas através do site da UTFPR. Além disso, é comum que muitos usuários procurem diretamente os setores e as coordenações, após não encontrar informações requeridas no site. A Figura 17 mostra os dados dessa pergunta.

A maioria das dúvidas dos alunos que procuram você podem ser solucionadas procurando no site da instituição?

(6 respostas)

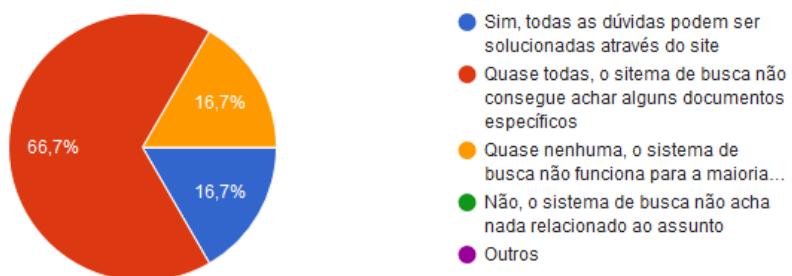


Figura 17 – Pesquisa 2 - Questão 2 (Dúvidas solucionadas através do site).

Fonte: Autoria própria.

Neste sentido, a terceira questão teve o intuito de levantar as dúvidas mais comuns dos alunos. Essas respostas contribuíram para confirmar o que os alunos tinham escrito na primeira pesquisa, sendo assim tornando os dados mais válidos para serem utilizados. A questão possuía o seguinte título: *Quais as principais informações/documentos que os alunos mais procuram quando perguntam para você?* e as respostas foram: legislação em geral, convalidação e transferência de curso, procedimentos das atividades acadêmicas, requerimentos, informações sobre o retorno do intercâmbio, regulamentos e editais, modelos, fichas e normas.

Com base no que foi visto na primeira pesquisa, na qual os respondentes relatam que o sistema de busca não funciona de uma forma eficiente, a questão quatro foi elaborada para saber o que deveria ser alterado no site e no sistema de busca para facilitar o acesso a essas informações e documentos. A Figura 18 mostra as respostas dessa pergunta.

Você acha necessário alguma alteração no site ou no sistema de busca da UTFPR? Por favor, descreva o que facilitaria na hora de encontrar alguma informação/documento.

(6 respostas)

Separar por Campi, melhorar a ordenação das informações e também a apresentação dos resultados.

O problema é a duplicidade de informações em campus diferentes. Sendo que as vezes o aluno acha um documento que é uma versão antiga, e não teria mais valor.

Sim, indexar as buscas como o Google.

Clareza nas localizações das informações

A possibilidade de detalhamento da busca e a definição de datas de inicio e fim para o resultados. Por exemplo, localizar apenas documentos de 2014 até 2015.

É difícil saber, inicialmente, onde estão as informações, sem se conhecer bem a hierarquia destas informações. Por exemplo, o que preciso está no site de CP ou o geral? Está em pró-reitoria de planejamento ou graduação? Está em qual assessoria?

Figura 18 – Pesquisa 2 - Questão 4 (O que poderia melhorar).

Fonte: Autoria própria.

A questão cinco, a qual possuía o título: *O que você acha da implantação de um chatbot no site da UTFPR?*, foi elaborada para saber a opinião desse grupo de pessoas sobre a implantação de um *chatterbot* no site da UTFPR. Todas as respostas obtidas são favoráveis a implementação desse sistema no site da universidade.

Por fim, nas duas últimas questões foram perguntadas informações do departamento no qual o respondente trabalha e o cargo exercido no mesmo. Quatro dos respondentes trabalham no Departamento de Computação (DACOM), um no DERAC e o último no Núcleo de Acompanhamento Psicopedagógico e Assistência Estudantil (NUAPE). Os cargos exercidos por eles são: Coordenadores de Curso, Chefe de Departamento, Coordenador de TCC, Assistente Administrativo e Assistente Social.

4.2 MATERIAIS E MÉTODOS

Para o desenvolvimento deste trabalho foi utilizado um método de desenvolvimento incremental. Esse método foi escolhido por ser adequado aos times pequenos de desenvolvedores e também por possuir requisitos que se modificam conforme o desenvolvimento. Além disso

com um método incremental é possível gerar um rápido panorama do projeto (BECK, 2009). As ferramentas utilizadas neste trabalho estão detalhadas logo abaixo.

- Plataforma de Linguagem Natural: Api.ai;
- JavaScript e AngularJS, para a programação da Aplicação Web e Painel;
- Firebase, como banco de dados para armazenar informações que necessitam ser dinâmicas;
- HTML e CSS, para a definição do corpo e estilo da Aplicação Web e Painel Administrativo;
- PHP, para a programação do Webhook.

Na próxima seção, o desenvolvimento do *chatterbot* é apresentado utilizando as ferramentas e métodos selecionados.

4.3 CHATTERBOT

Conforme visto na Seção 4.1, a elicitação de requisitos teve os seguintes propósitos: conhecer os problemas dos usuários e analisar a viabilidade da implementação de um *chatterbot*. Com base nos resultados discutidos na elicitação de requisitos, é possível considerar a implementação de um *chatterbot*, cujo principal objetivo consiste em fornecer um mecanismo alternativo de interação aos usuários que possibilite uma pesquisa mais eficiente e com uma melhor interação, por meio de uma plataforma de linguagem natural.

4.3.1 Configuração do Chatterbot

Este tópico busca mostrar como as novas informações podem ser cadastradas na base de conhecimento do *chatterbot*. Após o cadastro do *chatterbot* no Api.ai com as devidas informações, é possível adicionar os *Intents* e *Entities*(Entidades). O processamento do Api.ai funciona baseado em *Intents*, onde a entrada do usuário é processada e identificada para ser redirecionado para um *Intent* próprio. Cada um desses *Intents* possui um domínio específico, como o de Abono de Faltas que foi cadastrado para auxiliar o usuário com as informações sobre o abono de faltas na UTFPR.

Após informar um nome para o *Intent*, é necessário informar alguns exemplos de frases que o usuário pode digitar para poder possibilitar o agente a responder corretamente. Para isso foram formuladas algumas frases, conforme pode ser visto na Figura 19.

Com as expressões dos usuários cadastradas, é necessário cadastrar a resposta do *chatterbot*. A resposta cadastrada foi baseada no Regulamento da Organização Didático-Pedagógica dos Cursos de Graduação da UTFPR, a Figura 19 ilustra como ficou a resposta cadastrada no *Intent*.

The screenshot shows the Api.ai platform interface. On the left, there's a sidebar with various sections like 'UTFBot', 'Intents' (which is selected and highlighted in blue), 'Entities', 'Training [beta]', 'Integrations', 'Analytics [new]', 'Fulfillment', 'Prebuilt Agents', 'Docs', and 'Forum'. The main area is titled 'Abono_Faltas'. It has a 'User says' section containing a list of user expressions such as 'como funciona o abono de faltas', 'requerimento de faltas', etc. Below this is a 'Text response' section with a single item numbered 1, which contains text about military students' rights regarding leave cancellation. At the top right of the main area, there are 'SAVE' and '...' buttons.

Figura 19 – Cadastrando o Abono de Faltas.

Fonte: Autoria própria.

Baseado nessas informações cadastradas no *Intent*, o *chatterbot* está apto para responder aos usuários que informarem alguma das expressões cadastradas sobre abono de faltas. Como o Api.ai ainda não oferece suporte ao aprendizado de máquina para o Português Brasileiro, é necessário que o usuário digite frases semelhantes com as cadastradas no *Intent*.

Conforme pode ser observado, o *chatterbot* responderá todos os usuários que pergun-

tarem sobre abono de faltas com a mesma resposta. Isso é compreensível, pois, o abono de faltas funciona do mesmo jeito para todos os usuários. Porém, em alguns *Intents* é esperado uma resposta customizada baseada em alguns parâmetros que o usuário informa. Por exemplo, quando um usuário deseja saber o email de um professor. É inviável cadastrar estaticamente essas informações dentro do Api.ai, sendo recomendável a execução de um processamento desses parâmetros reconhecidos para poder retornar uma resposta baseada neles.

Para exemplificar um cenário onde as informações que são retornadas para o usuário devem ser dinâmicas e retornar mensagens baseadas em alguns parâmetros, foi necessário o cadastro de um *Intent* que retorna o email de um professor requisitado pelo usuário.

Antes de cadastrar esse novo *Intent*, foi necessário cadastrar um *Entity*(Entidade) para os professores. As Entidades servem para armazenar diversos valores que uma variável do Api.ai pode assumir. Isso é necessário para facilitar o cadastro das entradas de um *Intent*. Por exemplo, supondo que existem dois professores, José e Joaquim, seria necessário cadastrar duas frases iguais para um *Intent*: "Qual o email do José?" e "Qual o email do Joaquim?". Utilizando as Entidades do Api.ai, seria necessário cadastrar apenas uma expressão para que o Api.ai entendesse que essa frase pode conter qualquer nome cadastrado naquela Entidade.

Name	Synonyms
Willian Massami Watanabe	watanabe, willian, willian watanabe, watinha
André Takeshi Endo	andre, endo, andre endo
Diego Roberto Antunes	diego, drantunes
Priscila Tiemi Maeda Saito	priscila

Figura 20 – Cadastrando a Entidade Professores.

Fonte: Autoria própria.

Após a configuração da Entidade de professores, que pode ser vista na Figura 20, foi cadastrado o *Intent* com o nome "Email". As expressões cadastradas foram as seguintes: "Qual o email da X", "Qual o email da professora X", "Email da X", "Qual o email do X", "Email do professor X", "Email do X" e "Qual o email do professor X".

A Figura 21 mostra que os nomes dos professores foram grifados com uma cor, isso ocorre quando a Plataforma de Linguagem Natural identifica uma Entidade na expressão informada, conforme explicado anteriormente.

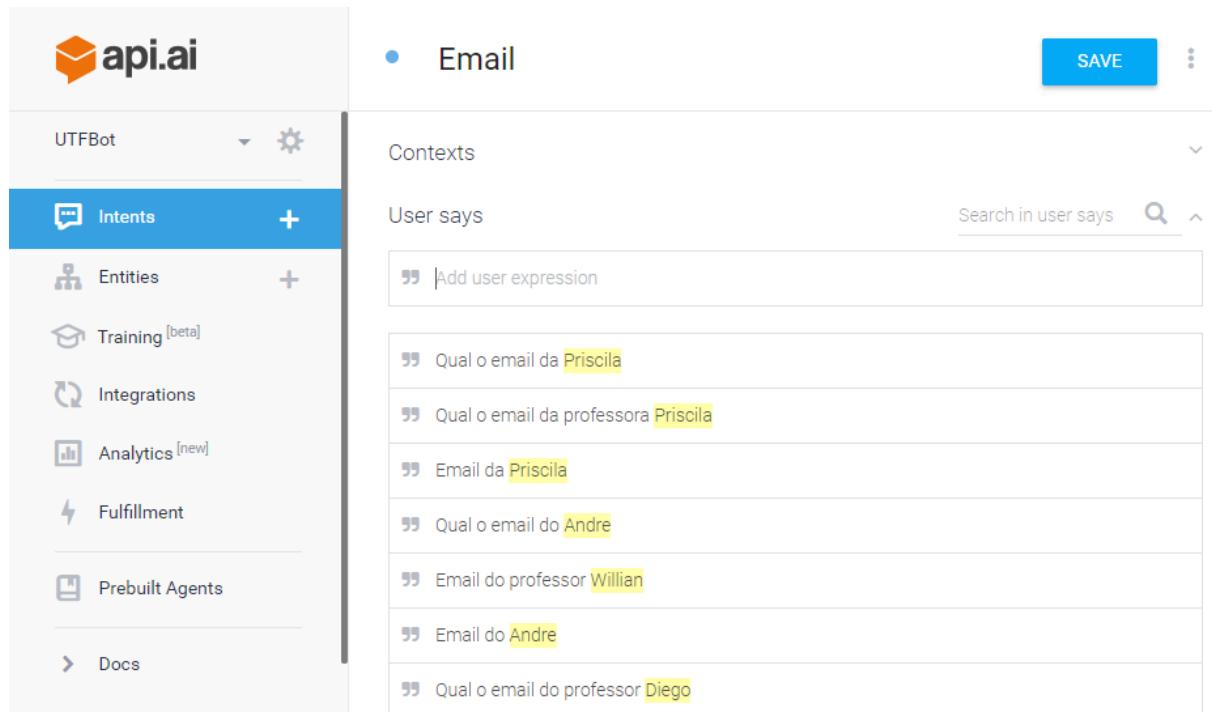


Figura 21 – Cadastrando a *Intent* de Email.

Fonte: Autoria própria.

De acordo com o que foi explicado anteriormente, esse *Intent* deve retornar informações dinâmicas, baseadas nas informações que o usuário irá fornecer. Nesse caso, o usuário poderá perguntar o email de um professor específico. Para isso, será necessário utilizar um Webhook para processar as informações e retornar o resultado dinâmico. Para que o Api.ai envie essas informações reconhecidas para um Webhook, a opção "Use webhook" na seção *Fulfillment* deverá estar marcada no cadastro do *Intent*. As informações são enviadas para o Webhook utilizando o RESTful, o retorno também deve utilizar o RESTful para enviar a resposta do Webhook para o Api.ai.

Pensando em simplificar o cadastro de algumas Entidades, como os professores, foi criado um Painel Administrativo de Cadastro de Entidades. Nesse painel é possível cadastrar os valores que determinadas Entidades podem assumir. Dentro desse painel é possível cadastrar os professores e departamentos. Quando um valor de uma Entidade é cadastrado nesse painel, ele envia o nome e os sinônimos dessa Entidade para o Api.ai por meio do RESTFul, para que possa ser possível cadastrar ele na Entidade, e para o banco de dados do Firebase são enviadas todas as outras informações.

Esse Painel Administrativo, que pode ser visto na Figura 22, foi desenvolvido utilizando HTML, CSS, JavaScript e AngularJS.

Cadastrar Professor

Nome do(a) Professor(a):
Nome

Sinônimos (separar por vírgulas):
Apelidos, sobrenomes, etc

Titulação:
[campo com scroll]

Número do Escaninho:
Escaninho

Email:
Email

Departamento:
[campo com scroll]

Ramal:
Ramat

Cadastrar

Figura 22 – Painel Administrativo para cadastrar Professor.

Fonte: Autoria própria.

4.3.2 Aplicação Web

A Aplicação Web foi desenvolvida para permitir a utilização do *chatterbot* pelos usuários. Essa aplicação foi construída utilizando HTML, CSS e JavaScript. A interação do usuário com o *chatterbot* é simples e de fácil entendimento, o usuário precisa apenas digitar a sua mensagem e enviar.

As mensagens enviadas pelo usuário para o *chatterbot* são transformadas em um JSON através da Aplicação Web por um método no código em JavaScript para ser enviada no corpo da requisição, juntamente com um código de autenticação do agente. Essa requisição é realizada utilizando o AJAX². Após o envio dessa requisição, a Aplicação Web aguarda uma resposta do Api.ai, e quando a resposta é recebida, o conteúdo dela é mostrado para o usuário em forma de um diálogo no chat.

²Realiza requisições assíncronas para algum servidor na internet.

Conforme pode ser visto na Figura 23, a aplicação do *chatterbot* também pode ser utilizada em um dispositivo móvel, como um *smartphone*. A aplicação é capaz de se adaptar conforme o tamanho da tela do dispositivo, tornando assim uma aplicação de fácil acesso a vários dispositivos.

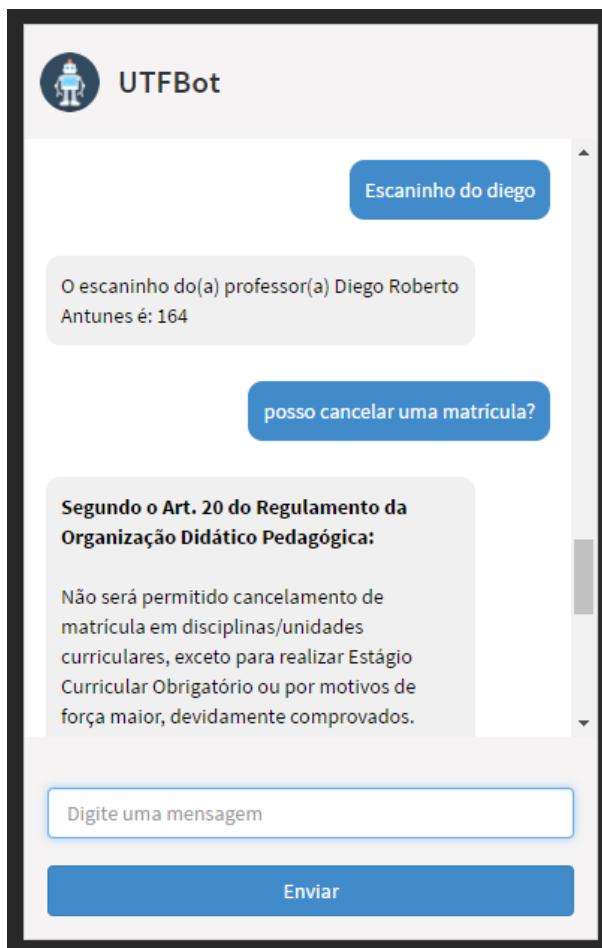


Figura 23 – Aplicação Web Responsiva.

Fonte: Autoria própria.

4.3.3 Arquitetura Geral

De acordo com o que foi visto anteriormente, dependendo do tipo de *Intent* é necessário realizar um processamento dos dados através de um Webhook. O Webhook deste trabalho foi desenvolvido utilizando a linguagem de programação PHP. Todos os dados que necessitam de processamento são enviados para que o Webhook possa retornar um resultado, como algumas informações dos professores (e.g. escaninho, email). O trecho desse código de processamento de informações, retornando os dados do banco de dados, sobre os professores pode ser visto na Figura 24.

```

function getDadosProfessor($professor, $tipo) {

    $firebase = connectFirebase();
    $professores = $firebase->getReference('Professores')->getData();

    foreach($professores as $professor_key => $professorAux){
        if ($professorAux["nome"] == $professor){
            if($tipo == "escaninho"){
                return $professorAux["escaninho"];
            }
            else if($tipo == "email"){
                return $professorAux["email"];
            }
            else if($tipo == "ramal"){
                return $professorAux["ramal"];
            }
            else if($tipo == "departamento"){
                return $professorAux["departamento"];
            }
        }
    }
}

```

Figura 24 – Trecho do código do Webhook.

Fonte: Autoria própria.

Para alguns *Intents* o Webhook necessita utilizar um *crawler*³ para recuperar algumas informações, como acontece na funcionalidade de retornar o cardápio do dia, a qual será explicada posteriormente. A utilização desse *crawler* é necessária pois a UTFPR não possui algum tipo de *web service* para que algumas informações sejam recuperadas através de uma requisição via RESTful. Um trecho do código onde o Webhook que inicializa um *crawler* pode ser visto na Figura 25.

```

function getCardapio() {
    $html = file_get_html('https://sistemas.cp.utfpr.edu.br/slogin/');

    $cardapio = $cardapio . "Data: " . $html->find('th', 0)->plaintext . '<br>';

    foreach($html->find('td') as $e) {
        $labels = $e->find('label');
        for ($i=0; $i < count($labels); $i++) {
            if ($i == 0) {
                $cardapio = $cardapio . $labels[$i]->plaintext . ': ';
            }
            else {
                $cardapio = $cardapio . $labels[$i]->plaintext . ($i == count($labels) - 1 ? '<br>' : ', ');
            }
        }
    }

    return $cardapio;
}

```

Figura 25 – Trecho do código do Crawler.

Fonte: Autoria própria.

Assim como a Aplicação Web, os componentes do Painel Administrativo de Cadastro

³Robôs que navegam em páginas web extraindo conteúdos.

de Entidades também podem ser substituídos facilmente por outras plataformas e tecnologias. O Banco de Dados pode ser substituído por qualquer tipo de SGBD, a Plataforma de Linguagem Natural pode ser substituída por outra e o Painel Administrativo pode ser construído utilizando outras tecnologias e linguagens de programação. A Arquitetura Geral da Aplicação Web e Painel Administrativo pode ser visualizada na Figura 26.

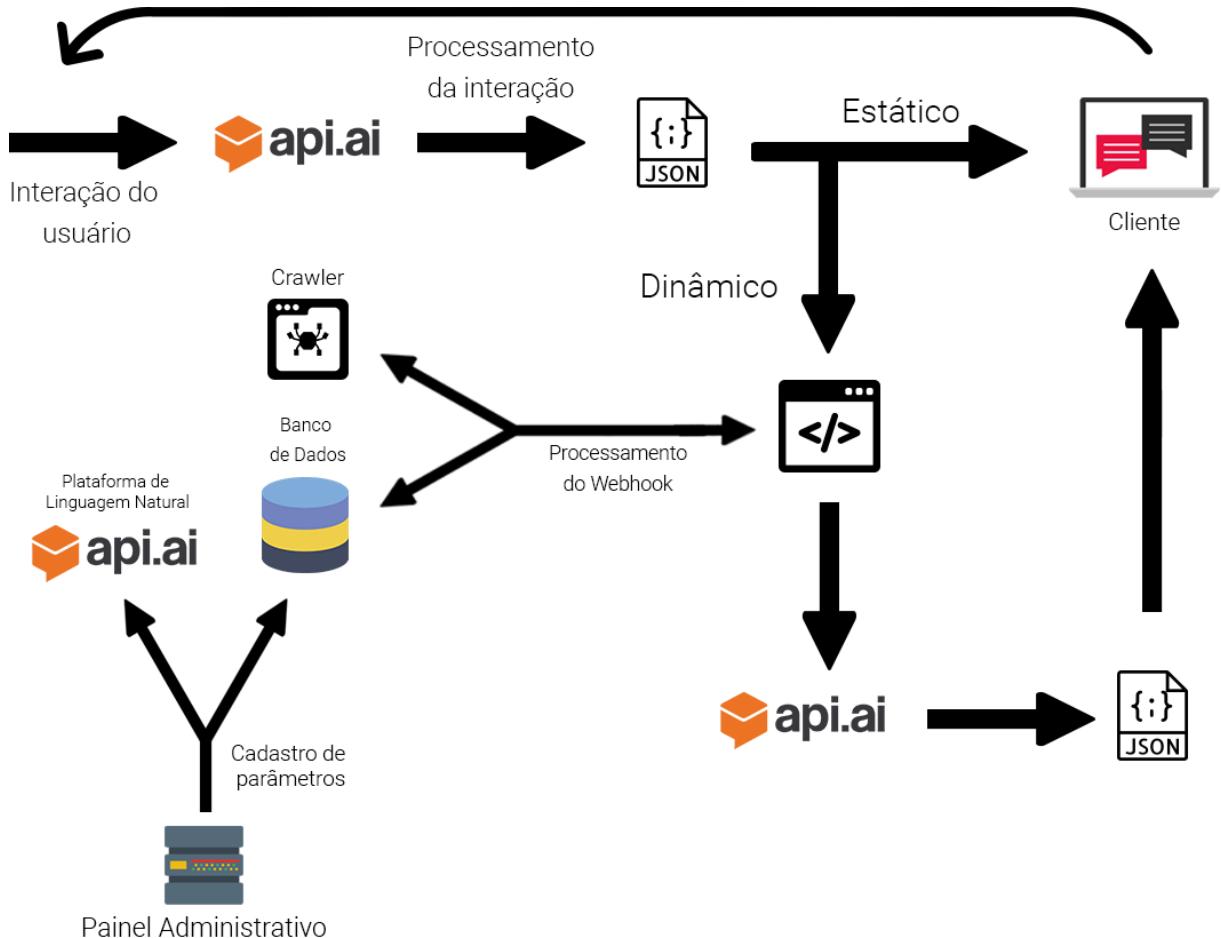


Figura 26 – Arquitetura Geral.

Fonte: Autoria própria.

4.3.4 Funcionalidades

No desenvolvimento do *chatterbot* foram cadastradas as informações citadas anteriormente, Abono de Faltas e Email dos Professores, e algumas outras baseadas nas respostas da Pequisa 1 feita com os usuários do site da UTFPR. A seguir são apresentadas as demais informações cadastradas e utilizadas neste estudo de caso.

4.3.4.1 Exame de Suficiência e Convalidação de Matérias

Esta funcionalidade foi adicionada com o intuito de auxiliar os alunos que desejam realizar um exame de suficiência ou uma convalidação de matéria. Assim, quando o aluno pede ajuda para o *chatterbot* sobre esses tópicos, são apresentadas algumas informações que estão no Regulamento da instituição e como ele pode realizar um requerimento de um exame de suficiência ou convalidação de matérias. Para ambos os casos, o aluno é redirecionado para um site onde é possível fazer o requerimento online.

4.3.4.2 Trancamento de Curso, Cancelamento de Matrícula, Segunda Chamada, Turma sem Presença, Jubilamento e Abono de Faltas

Estas funcionalidades também foram criadas para auxiliar o aluno com várias dúvidas: Como trancar o curso; Como cancelar uma disciplina/matéria; Como realizar uma segunda chamada e quem tem o direito de realizá-la; Como funciona e como abrir uma turma sem presença e Como funciona o jubilamento. Todas essas questões foram respondidas baseadas nos Artigos do Regulamento da Organização Didático-Pedagógica. Além disso, o aluno é instruído como resolver tal problema se o mesmo possuir uma solução.

4.3.4.3 Emissão de Certificados

Atualmente é possível emitir um certificado de palestras, eventos, dentre outros em um sistema online da instituição. Para isso, é necessário acessar o site do sistema, selecionar o câmpus, ano e evento para abrir uma lista com todos os participantes para poder selecionar o seu certificado, conforme pode ser visto na Figura 27. Buscando facilitar essa busca, foi desenvolvido no *chatterbot* a opção de emitir um certificado que esteja dentro desse sistema na UTFPR. Para isso foi necessário a utilização do Webhook para poder ler a página através de um *crawler*⁴ e extrair as informações necessárias. As Figuras 28 e 29 ilustram o funcionamento do *chatterbot* para a emissão de certificados.

4.3.4.4 Consulta ao Cardápio do Dia

Nessa funcionalidade, o usuário do *chatterbot* pode pedir o cardápio do dia do Restaurante Universitário de Cornélio Procópio. O cardápio do dia também está disponível em um site da instituição, onde é possível visualizar o prato principal, acompanhamento, saladas e sobremesa daquele dia. Para essa funcionalidade, também foi utilizado o Webhook para ler a página onde o cardápio se encontra e extrair as informações para retornar para o usuário. Nas

⁴Robôs que navegam em páginas web extraíndo conteúdos.

Ao selecionar o evento, você será direcionado para uma nova página, com a listagem de certificados deste evento.

Câmpus*:
Todos

Ano*:
Todos

Evento*:
Selecione...

Figura 27 – Sistema de Gestão de Certificados Eletrônicos UTFPR

Fonte: Autoria própria.



Figura 28 – Requisitando um certificado.

Fonte: Autoria própria.

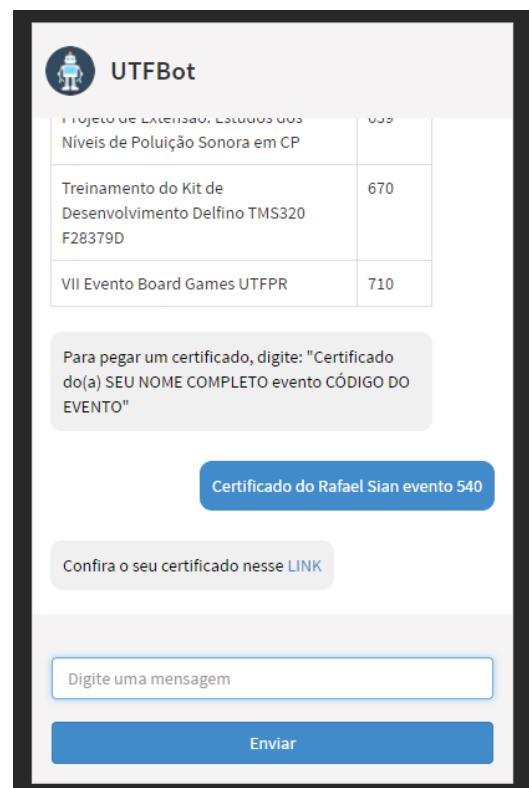


Figura 29 – Conseguindo o certificado.

Fonte: Autoria própria.

Figuras 30 e 31 pode ser visto, respectivamente, o cardápio no site da instituição e o *chatterbot* retornando o cardápio para o usuário.

Cardápio

25/05/2017

Principal
Carne Assada ao molho madeira

Acompanhamento
Repolho Refogado com tomate

Saladas
Almeirão
Tomate com milho

Sobremesa
Fruta

Dúvidas, críticas e sugestões

Dúvidas críticas ou sugestões

Enviar

Figura 30 – Cardápio no Sistemas UTFPR-CP.

Fonte: Autoria própria.

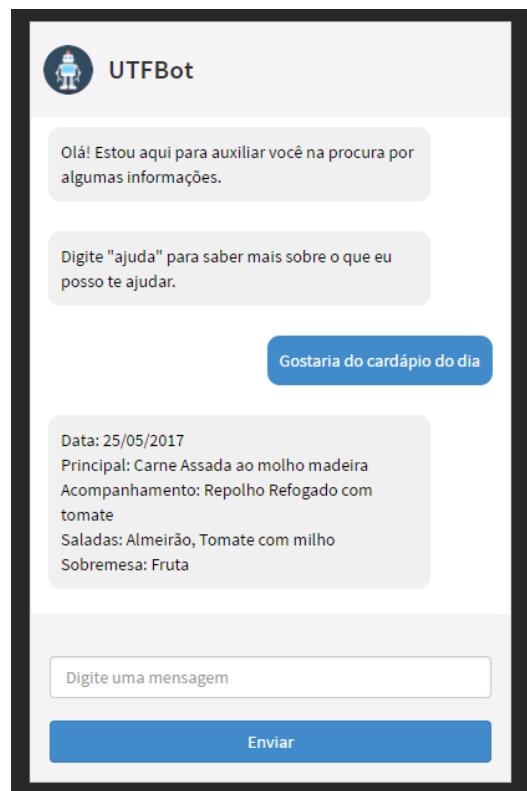


Figura 31 – Requisitando o cardápio.

Fonte: Autoria própria.

4.3.4.5 Informações sobre Professores

Com essa funcionalidade no *chatterbot* é possível retornar para o usuário várias informações sobre professores, tais como: Email; Ramal; Titulação e Escaninho. Essa funcionalidade retorna as informações que estão contidas em um banco de dados da própria aplicação web, que foram cadastradas através do Painel Administrativo. A escolha de guardar as informações em um banco de dados próprio foi tomada pois não foi encontrado no site da instituição uma tabela ou algo do parecido que contenha todas essas informações juntas. A vantagem desse cadastro no banco de dados é que torna possível o cadastro do professor dentro de uma Entidade no Api.ai, podendo assim guardar sinônimos ou apelidos dos professores e tornando assim mais fácil do *chatterbot* entender de qual professor que o usuário está solicitando informações.

4.3.4.6 Programas

A UTFPR possui vários programas para alunos da instituição. Esses programas variam de direção, edital, entre outros detalhes. Quando um usuário pergunta ao *chatterbot* sobre

algum programa disponível, ele retorna alguns detalhes e um link do site da UTFPR onde pode ser encontrado mais informações. Caso o usuário pergunte somente sobre os programas que existem na Universidade, o *chatterbot* retorna uma lista de todos os programas que estão cadastrados.

Essa funcionalidade também utiliza um banco de dados para salvar as informações sobre os programas. A escolha de deixar essas informações em um banco de dados próprio foi pelo motivo de não existir uma página que liste todos os programas existentes na instituição, sendo assim necessário manter essas informações dentro da aplicação. Nas Figuras 32 e 33 pode ser visto o funcionamento do *chatterbot* para os programas da UTFPR.

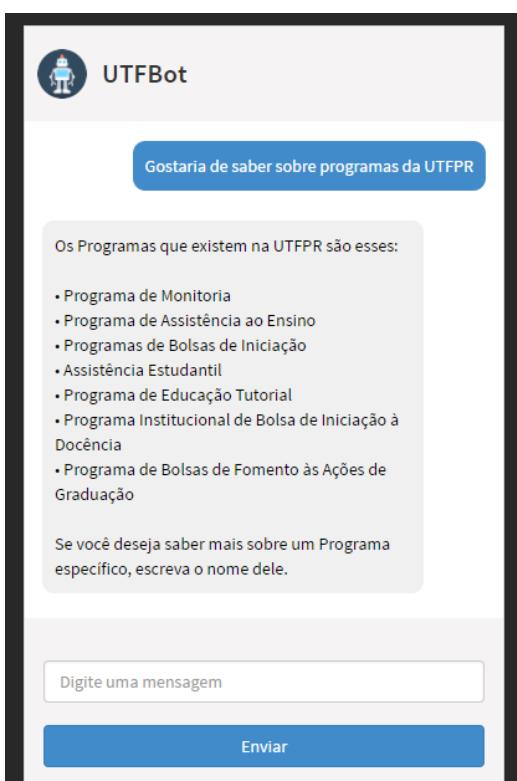


Figura 32 – Programas disponíveis na UTFPR.

Fonte: Autoria própria.

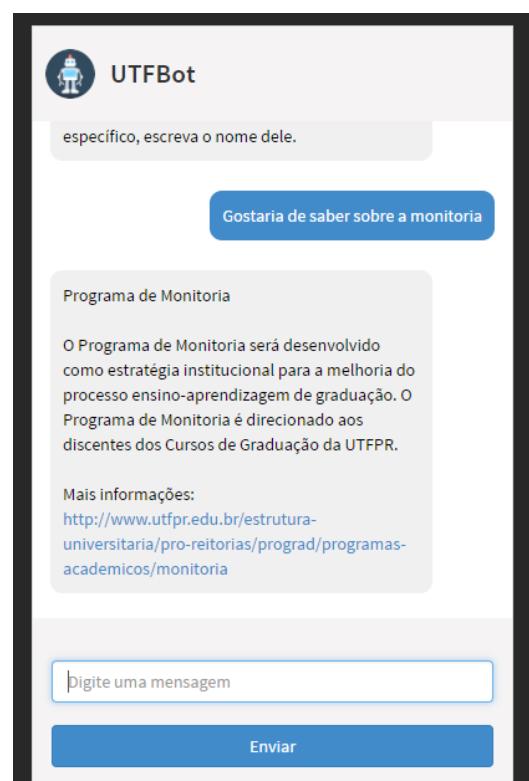


Figura 33 – Programa de monitoria.

Fonte: Autoria própria.

5 RESULTADOS E DISCUSSÕES

Para obter alguns resultados do *chatterbot*, foi utilizado o Google Forms¹ para realizar uma pesquisa com os usuários. O objetivo dessa pesquisa foi verificar se a experiência com a recuperação de dados e informações usando o *chatterbot* desenvolvido poderia indicar uma alternativa viável para o atual sistema de busca do site da UTFPR. A pesquisa foi divulgada entre os usuários do site da instituição, obtendo um total de 15 respondentes.

Esse questionário foi dividido em duas seções. A primeira seção teve o intuito de verificar a dificuldade em encontrar informações e documentos no site da UTFPR. Já a segunda seção foi feita para testar a acurácia do *chatterbot* em compreender e responder os usuários de forma correta. Todas as perguntas foram baseadas em funcionalidades que o *chatterbot* possui, conforme explicado no capítulo anterior.

Em todas as perguntas era requisitado que o respondente visitasse o site da instituição e tentasse achar algum documento ou informação, logo após era requisitado que o respondente tentasse achar o mesmo documento ou informação utilizando o *chatterbot*. Nas seções a seguir são detalhados os resultados e análises do site da UTFPR com o *chatterbot*.

5.1 EXAME DE SUFICIÊNCIA, ABONO DE FALTAS, CONVALIDAÇÃO, TRANCAMENTO DE CURSO E CANCELAMENTO DE MATRÍCULA

Nessas 5 perguntas os usuários foram requisitados a achar informações e documentos sobre cada uma no site da instituição, utilizando a barra de pesquisa, menus e índices.

As imagens 34, 35, 36, 37 e 38 mostram os resultados obtidos com essa pesquisa, comparando o que os usuários responderam quando foi requisitado a utilização do site da UTFPR e utilizando o *chatterbot*.

Conforme pode ser visto, os resultados do *chatterbot* são melhores e mais rápidos do que os do site da UTFPR. Analisando esses resultados com as respostas obtidas na Pesquisa 1 do Capítulo 4, pode ser visto o motivo que leva mais de 60% dos usuários a utilizarem meios alternativos ao sistema de busca do site da universidade.

Como o *chatterbot* é um agente de conversação, ele pode solicitar alguns detalhes para os usuários que podem facilitar a busca pelos documentos, além de reduzir o risco dele retornar algum documento ou informação incorreta, como acontece com a busca no site da universidade, muitas vezes a informação encontrada é referente a outro câmpus. Com o *chatterbot* isso pode ser minimizado, pois bastaria perguntar para o usuário em qual câmpus ele deseja a pesquisa.

Outro ponto que pode ser analisado com a pesquisa realizada é sobre o funcionamento

¹www.forms.google.com

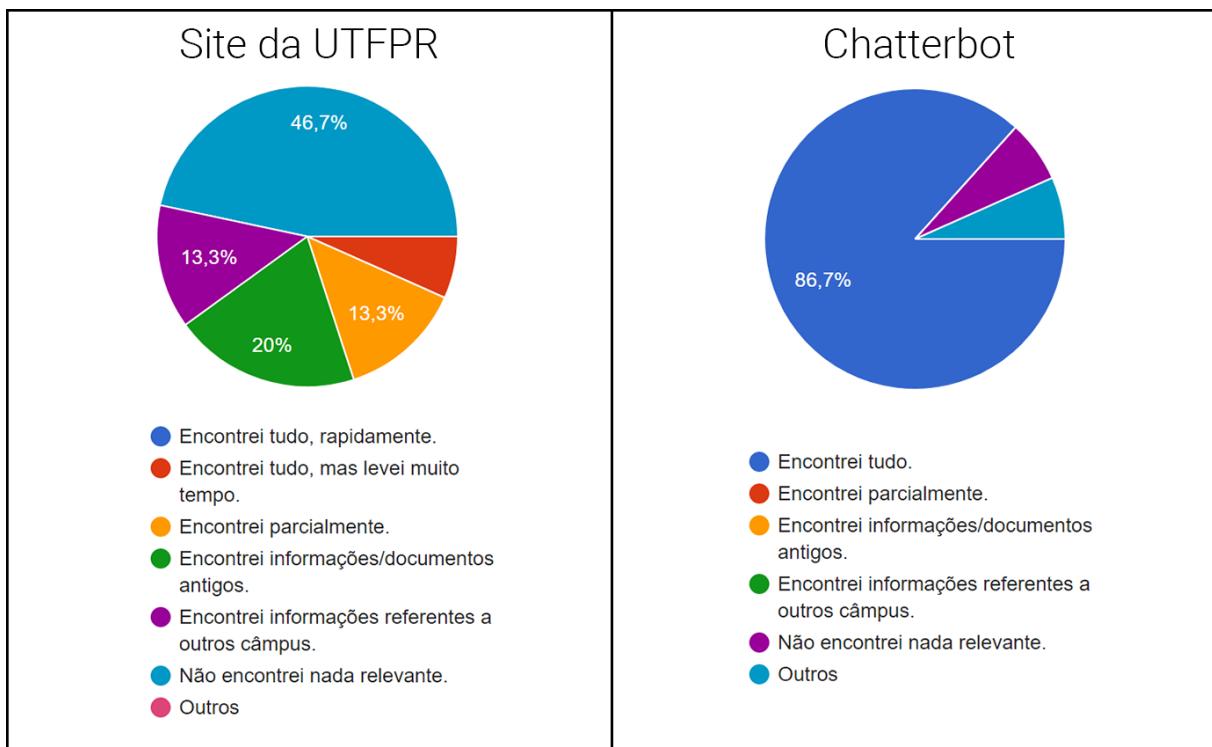


Figura 34 – Exame de Suficiência.

Fonte: Autoria própria.

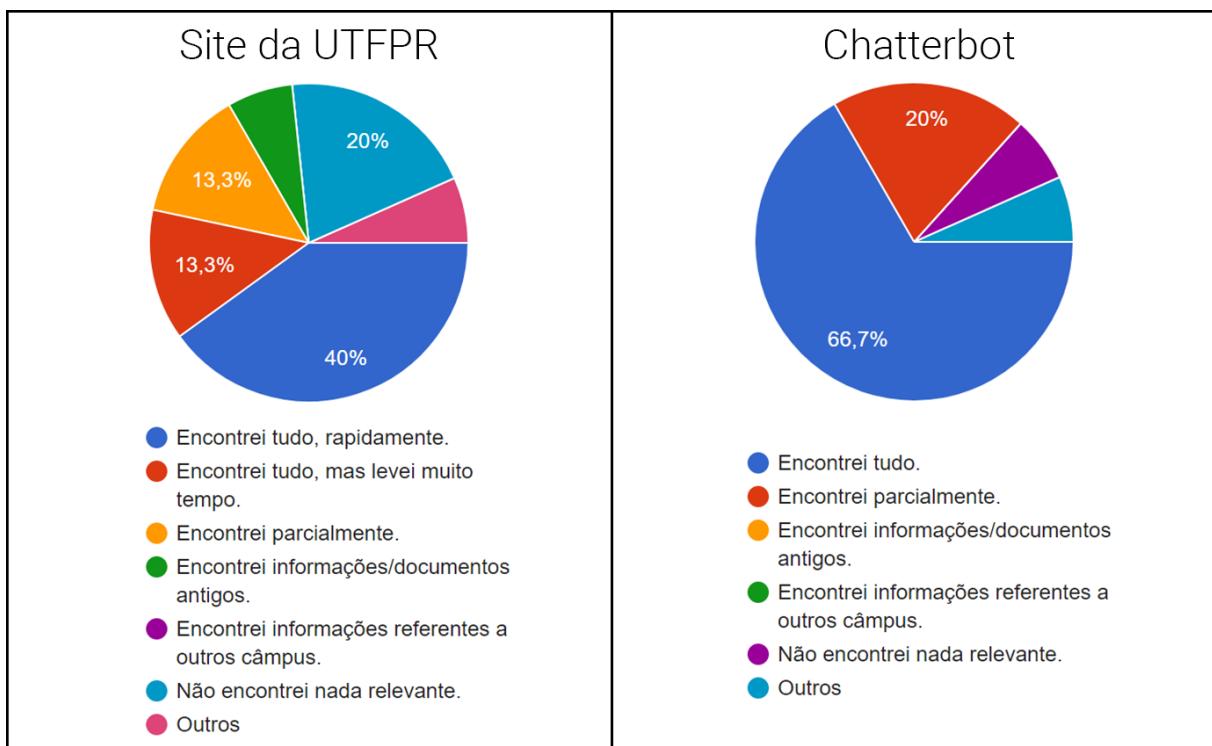


Figura 35 – Abono de Faltas.

Fonte: Autoria própria.

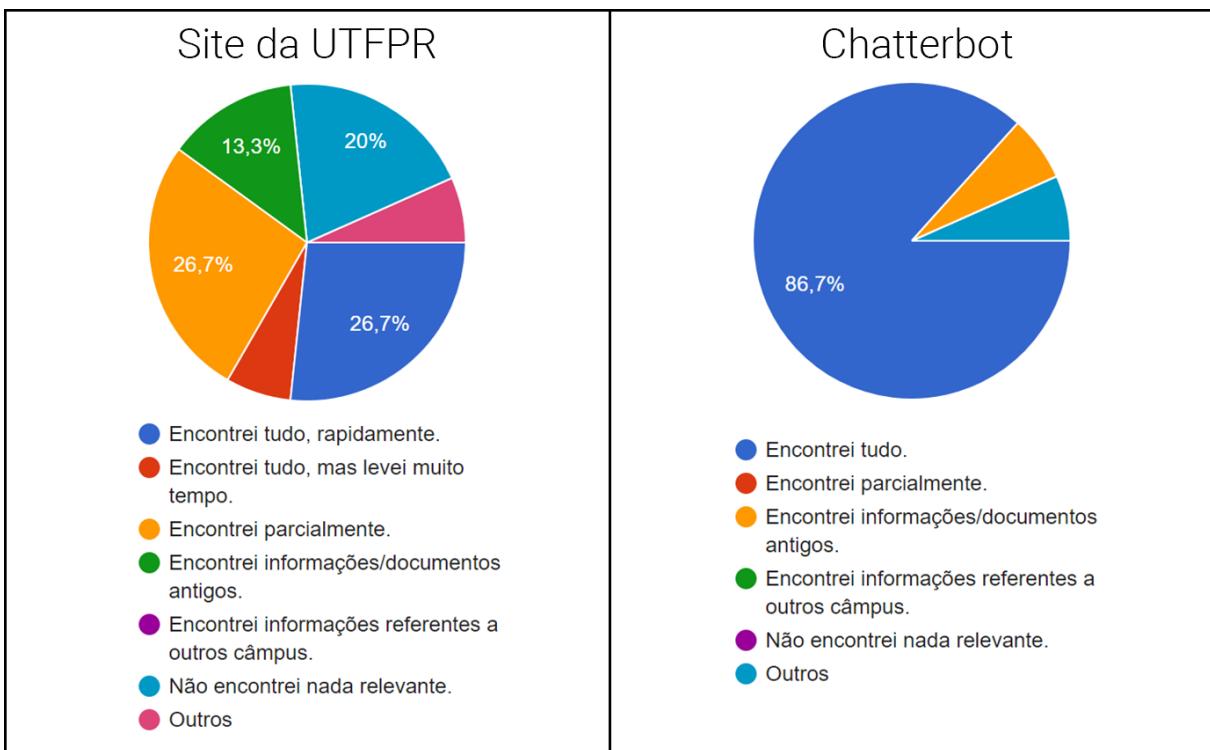


Figura 36 – Convalidação.

Fonte: Autoria própria.

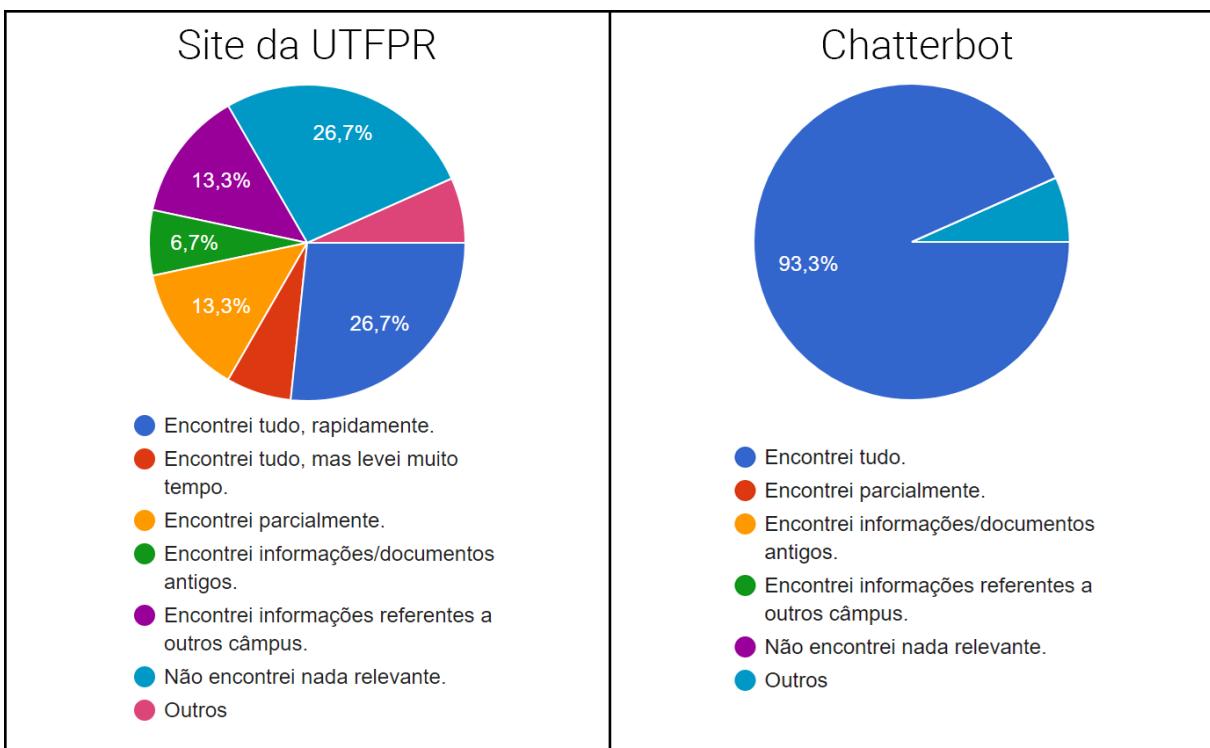


Figura 37 – Trancamento de Curso.

Fonte: Autoria própria.

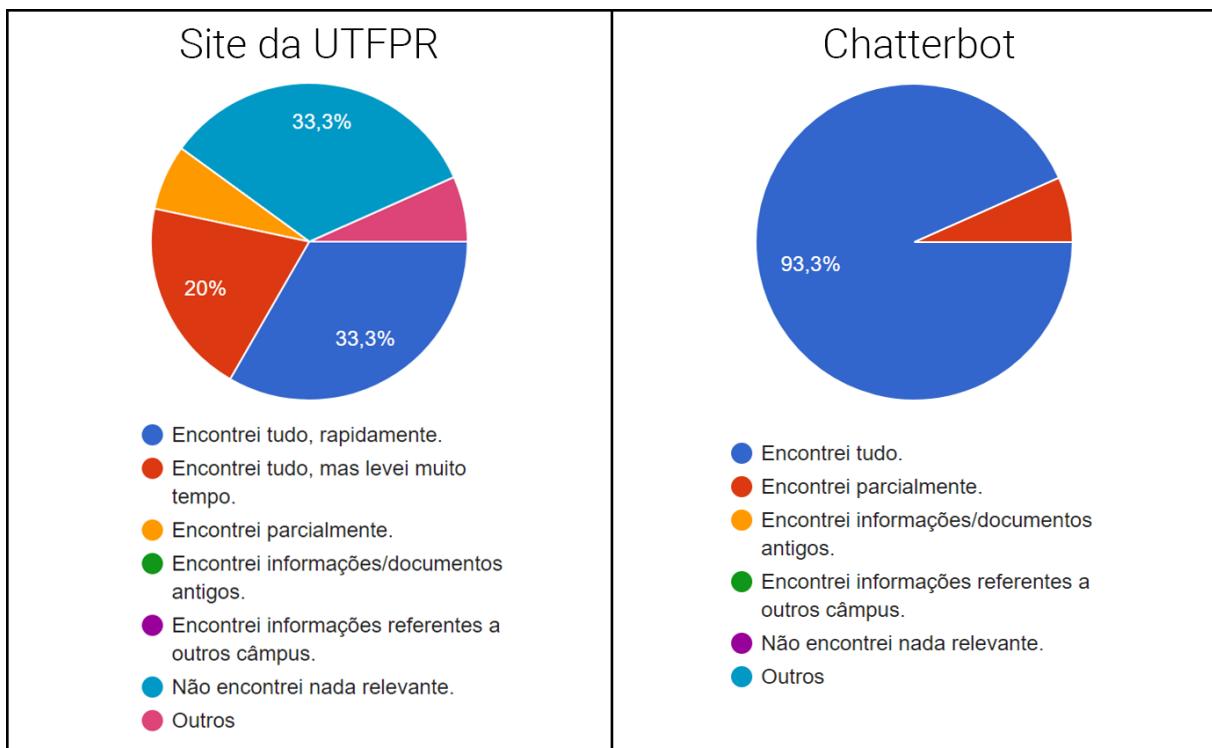


Figura 38 – Cancelamento de Matrícula.

Fonte: Autoria própria.

do aprendizado de máquina do Api.ai para o Português Brasileiro. Ainda não é possível utilizá-lo nesse idioma e com isso algumas entradas dos usuários podem não retornar os dados esperados. Caso o usuário informe algo diferente dos exemplos cadastrados de entradas, o *chatterbot* pode não compreender e com isso retornar um erro.

5.2 EMISSÃO DE CERTIFICADOS

Nessa pergunta foi solicitado aos usuários para que eles emitissem um certificado de participação em um evento. Conforme explicado nas funcionalidades do *chatterbot*, os certificados de eventos ficam armazenados em um sistema da instituição onde é possível acessá-lo por um site. Porém, esse sistema não fica disponível em algum local de fácil acesso no site da UTFPR. Conforme por ser visto na Figura 39, menos de 7% dos respondentes conseguiram emitir o certificado utilizando o site principal da UTFPR para encontrar o sistema de emissão de certificados.

Com o *chatterbot*, o resultado dessa pesquisa mostrou que 60% dos usuários conseguiram emitir o certificado. Esse número é quase 10 vezes maior comparando com o resultado do site da UTFPR.

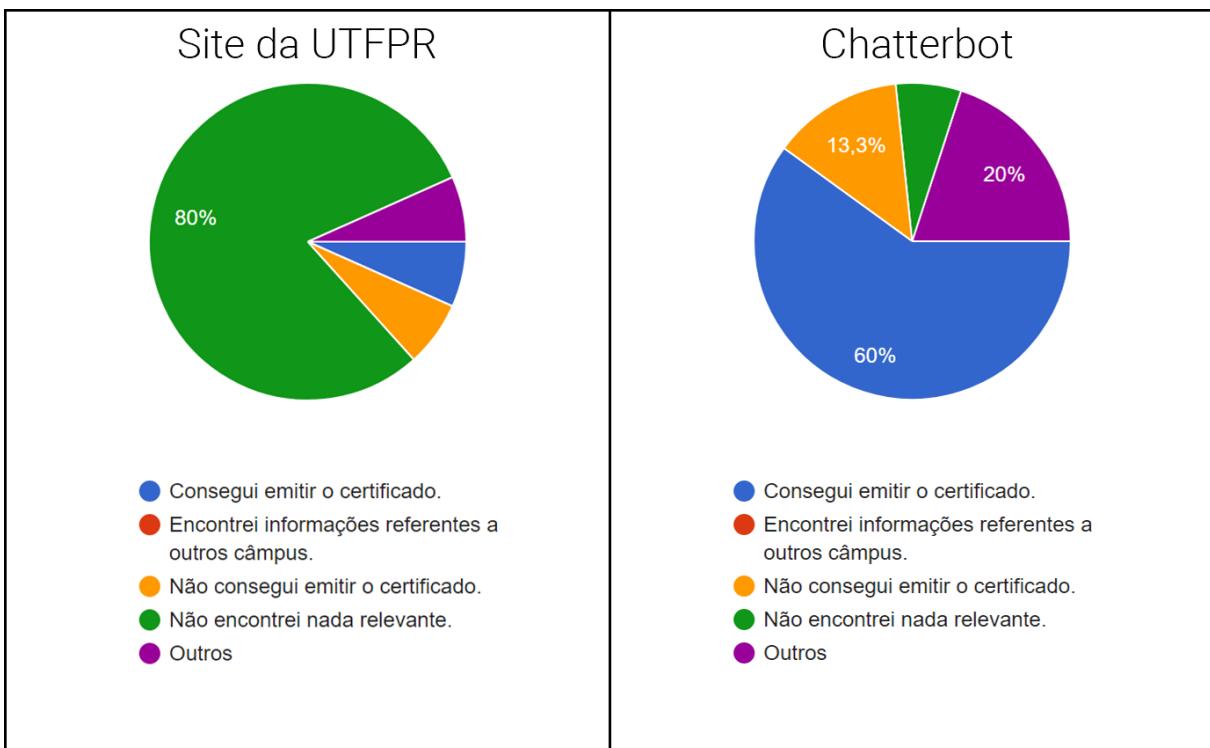


Figura 39 – Emissão de Certificados.

Fonte: Autoria própria.

5.3 INFORMAÇÕES SOBRE MONITORIA

Nessa pergunta foi solicitado ao respondente que ele pesquisasse os horários da monitoria do semestre atual (2017/01). A Figura 40 mostra o resultado da pesquisa, onde apenas 33% dos usuários conseguiram encontrar os horários da monitoria do semestre atual. A resposta do *chatterbot* mostra que 93% dos usuários conseguiram recuperar esses horários dos monitores, um número satisfatório quando se trata de uma informação importante para os alunos.

Outro problema que os alunos enfrentam com as informações da monitoria é sobre a organização das informações que podem ser encontradas no site. Uma simples busca por "Horários Monitoria" no site da UTFPR retorna vários resultados desatualizados com horários da monitoria de semestres passados e editais do programa. Para conseguir encontrar esses horários é necessário seguir seguintes os passos: Site da UTFPR -> Alunos -> Cornélio Procópio -> Monitoria. Mesmo encontrando essa página com o documento com os horários, o usuário pode se sentir confuso com o nome da página, data de atualização e até mesmo o link da página, o que pode ser visto na Figura 41.

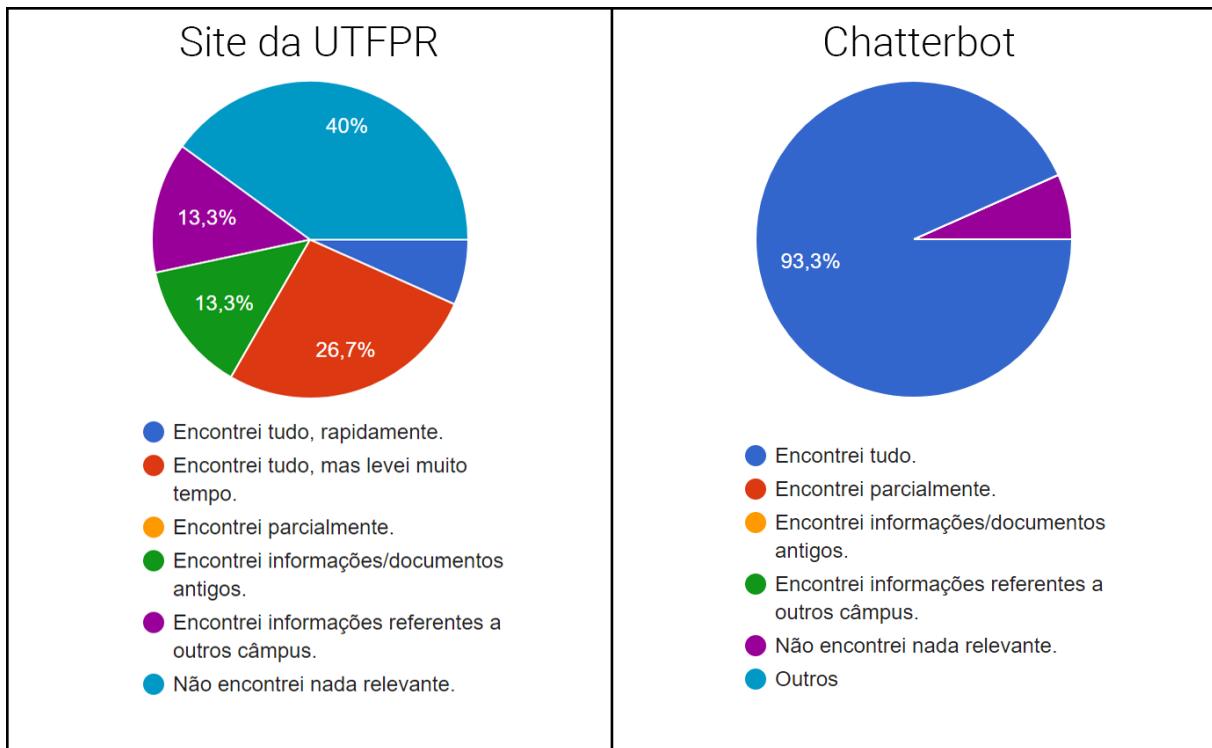


Figura 40 – Informações sobre Monitoria.

Fonte: Autoria própria.



Figura 41 – Confusões visuais da página da Monitoria.

Fonte: Autoria própria.

6 CONSIDERAÇÕES FINAIS

Conforme discutido anteriormente neste trabalho, o site da UTFPR enfrenta alguns problemas de precisão e facilidade em seu sistema de busca, no qual muitos usuários acabam procurando formas alternativas de pesquisa. O problema principal é o difícil acesso às informações e aos documentos que estão no repositório principal da universidade. Com isso, vários usuários não conseguem cumprir seus objetivos de busca e, desta maneira, não encontram os dados desejados.

As pesquisas realizadas no início do trabalho mostraram-se importantes para entender o problema e formar hipóteses de como solucioná-lo. A partir disso, o principal foco do trabalho foi a implementação de um meio alternativo de recuperação de dados e informações, utilizando uma interação natural através de *chatterbots*. A maioria das sugestões dos usuários foram implementadas no *chatterbot*. O questionário realizado no final do trabalho mostra que o *chatterbot* pode ser considerado uma alternativa no qual os usuários usariam e conseguiriam encontrar as informações necessárias com um fácil acesso e um rápido retorno.

Analizando a pesquisa final, pode ser visto que os usuários conseguiram encontrar as informações e documentos de uma maneira muito mais fácil utilizando o *chatterbot*, em comparação com o site da UTFPR. Além de encontrar essas informações, a implementação do *chatterbot* possibilitou os usuários a encontrarem coisas que estão além do site da universidade, como as informações sobre o número do escaninho dos professores.

A plataforma de linguagem natural utilizada neste trabalho, o Api.ai, possui o funcionamento dos agentes baseado no aprendizado de máquina supervisionado, onde é necessário adicionar conteúdo na base de dados para que ele possa ser treinado. Com isso, após o treinamento é possível analisar as saídas e classificar se elas são consistentes com o ambiente trabalhado, caso não seja é necessário realizar outros treinamentos com o agente até que as saídas sejam satisfatórias.

6.1 LIMITAÇÕES

Durante o desenvolvimento do trabalho foram encontradas algumas limitações, sendo que algumas delas dependem de funcionalidades de terceiros. Por exemplo, algumas informações que o *chatterbot* recupera, estão contidas no site da UTFPR e com isso é necessário acessá-las utilizando um *crawler* e algumas dessas requisições podem falhar ou exceder o tempo limite. Além disso, o Api.ai define um limite de 5 segundos de processamento do Webhook, e as vezes esses *crawlers* que são acionados pelo Webhook demoram mais do que isso para retornar algum resultado.

A utilização dos *crawlers* foi necessária para contornar o fato de que a instituição não possui algum *web service* para que as informações possam ser consumidas utilizando algum tipo de requisição. Os conteúdos do cardápio do dia, por exemplo, são extraídos através de um *crawler* para que possa ser possível mostrar esse conteúdo para o usuário do *chatterbot*.

Outra limitação que já foi citada anteriormente, é fato de que o Api.ai não possuir um suporte completo para o Português Brasileiro. O aprendizado de máquina nesse idioma não funciona. Portanto, torna-se necessário cadastrar muito mais exemplos para as entradas dos usuários nos *Intents* e, muitas vezes, se o usuário digitar alguma informação diferente ou alguma palavra incorreta, a plataforma do Api.ai não consegue identificar o *Intent*.

6.2 TRABALHOS FUTUROS

A tecnologia de *chatterbot* permite adicionar várias funcionalidades que podem ir melhorando ou auxiliando algum problema existente, sendo assim, possível adicionar mais funcionalidades no sistema com base no seu ambiente.

Outra funcionalidade que o Api.ai também suporta é a integração de um clique para outras plataformas, onde é possível integrar o *chatterbot* nas principais plataformas de mensagens instantâneas, como o Skype, Telegram e Facebook Messenger. Com isso, não é necessário que o usuário acesse uma página web, podendo utilizar o *chatterbot* através de um mensageiro instantâneo.

O *chatterbot* também poderia ser adicionado a página da UTFPR ou qualquer outra página da instituição adicionando uma pequena janela de chat, similar ao do Facebook. Essa integração facilitaria o uso tanto do *chatterbot*, quanto do sistema de busca da universidade.

Para otimizar as consultas das páginas através dos *crawlers* seria possível deixá-los executando em *background* para armazenar as informações em um banco de dados. Com isso, quando o Webhook necessitar dessas informações, ele recuperaria através do banco de dados, otimizando o processamento e diminuindo o tempo da requisição.

REFERÊNCIAS

- ANGULARJS. **AngularJS**. 2017. Disponível em: <<https://angularjs.org/>>. Citado na página 23.
- API.AI. **Conversational UX Platform for Bots, Apps, Devices, Services - Api.ai**. 2016. Disponível em: <<https://api.ai/>>. Citado 3 vezes nas páginas 20, 21 e 23.
- BARBOSA, Simone; SILVA, Bruno. **Interação humano-computador**: Elsevier Brasil, 2010. Citado na página 12.
- BECK, Kent. **Programação Extrema (XP) Explicada**: Bookman Editora, 2009. Citado na página 32.
- CAPEK, Karel. **Rossum's Universal Robots**. 1921. Disponível em: <<http://preprints.readingroom.ms/RUR/rur.pdf>>. Citado na página 13.
- COPPIN, Ben. **Inteligência Artificial**: LTC - Livros Técnicos e Científicos Ltda, 2010. Citado 3 vezes nas páginas 13, 14 e 15.
- CYBIS, Walter; BETIOL, Adriana Holtz; FAUST, Richard. **Ergonomia e Usabilidade**: Novatec, 2010. Citado na página 12.
- DAHLBERG, Ingetraut. Teoria do conceito. **Ciência da informação**, v. 7, n. 2, 1978. Citado na página 10.
- FERREIRA, Guilherme Fischmann et al. Arquitetura da informação no desenvolvimento de aplicação web. 2015. Citado na página 23.
- FIELDING, Roy Thomas. **Architectural styles and the design of network-based software architectures**. 2000. Tese (Doutorado) — University of California, Irvine, 2000. Citado na página 24.
- FIREBASE. **Produtos do Firebase**. 2017. Disponível em: <<https://firebase.google.com/products/>>. Citado na página 24.
- FLANAGAN, David. **JavaScript: the definitive guide**: "O'Reilly Media, Inc.", 2006. Citado na página 23.
- GALVÃO, Adjimir de Moura. Persona-aiml: Uma arquitetura para desenvolver chatterbots com personalidade. Universidade Federal de Pernambuco, 2003. Citado 3 vezes nas páginas 9, 14 e 16.
- GNECCO, Bruno Barberi et al. Desenvolvimento de interfaces naturais de interação usando o hardware kinect. **Tendências e Técnicas em Realidade Virtual e Aumentada**, v. 2, p. 37–62, 2012. Citado na página 12.
- HEWETT et al. **ACM SIGCHI Curricula for Human-Computer Interaction**. 1992. Disponível em: <<http://old.sigchi.org/cdg/index.html>>. Citado na página 12.
- KARADIMCE, Aleksandar; BOGATINOSKA, Dijana Capeska. Using hybrid mobile applications for adaptive multimedia content delivery. In: IEEE. **Information and Communication Technology, Electronics and Microelectronics (MIPRO), 2014 37th International Convention on**, 2014. p. 686–691. Citado na página 23.

- LEONHARDT, Michelle Denise. Dorothy: um chatterbot para treinamento de profissionais atuantes no gerenciamento de redes de computadores. 2005. Citado 3 vezes nas páginas 9, 13 e 15.
- MAHMUD, Dewi Mariati; ABDULLAH, Nur Atiqah Sia. Mobile application development feasibility studies: A case study in universiti teknologi mara. In: IEEE. **Open Systems (ICOS), 2014 IEEE Conference on**, 2014. p. 30–35. Citado na página 23.
- NIEDERAUER, Juliano. Desenvolvendo websites com php. **São Paulo: Novatec**, 2004. Citado na página 23.
- PILASTRI, André Luiz; BREGA, José Remo Ferreira. Chatterbot com interatividade ao avatar encapsulado no ambiente virtual second life usando a base de conhecimento em aiml. 2009. Citado 2 vezes nas páginas 9 e 10.
- PREECE, Jennifer; ROGERS, Yvonne; SHARP, Helen. **Design de Interação: Além da interação homem-computador**: Bookman, 2005. Citado 3 vezes nas páginas 9, 12 e 25.
- PRESSMAN, Roger; MAXIM, Bruce. **Engenharia de Software-8ª Edição**: McGraw Hill Brasil, 2016. Citado na página 23.
- PRIMO, Alex FT et al. O uso de chatterbots na educação à distância. In: **IV Workshop de Informática na Educação. Porto Alegre: UFRGS, 12p**, 1998. Citado na página 13.
- RODRIGUEZ, Alex. Restful web services: The basics. **IBM developerWorks**, 2008. Citado na página 24.
- ROTHERMEL, Alessandra; DOMINGUES, Maria José Carvalho de Souza. Maria: Um chatterbot desenvolvido para os estudantes da disciplina métodos e técnicas de pesquisa em administração. **SEGET-SIMPÓSIO DE EXCELÊNCIA EM GESTÃO E TECNOLOGIA, Resende**, 2007. Citado na página 16.
- RUPPEN, Andreas; PASQUIER, Jacques; HÜRLIMANN, Tony. A restful architecture for integrating decomposable delayed services within the web of things. **International Journal of Internet Protocol Technology**, Inderscience Publishers, v. 6, n. 4, p. 247–259, 2011. Citado na página 24.
- RUSSELL, Stuart; NORVIG, Peter. **Artificial Intelligence: a modern approach**: Englewood Cliffs: Prentice-Hall, 1995. Citado 3 vezes nas páginas 13, 14 e 15.
- SENDGRID. **What's a Webhook?** 2017. Disponível em: <<https://sendgrid.com/blog/whats-webhook/>>. Citado na página 24.
- TEIXEIRA, Sérgio et al. Chatterbots em ambientes de aprendizagem—uma proposta para a construção de bases de conhecimento. In: **Anais do Workshop de Informática na Escola**, 2005. v. 1, n. 1. Citado na página 13.
- WIT.AI. **Wit - landing**. 2016. Disponível em: <<https://wit.ai/>>. Citado 3 vezes nas páginas 20, 21 e 23.
- WOOLDRIDGE, Michael; JENNINGS, Nicholas R. Intelligent agents: Theory and practice. **The knowledge engineering review**, Cambridge Univ Press, v. 10, n. 02, p. 115–152, 1995. Citado na página 13.