



Faculdades Integradas de Taquara - Faccat
Av. Oscar Martins Rangel, 4.500
Taquara, RS, CEP 95600-000

Curso de Sistemas para Internet

MARVIN: PORTAL WEB COM INTEGRAÇÃO DE ATENDIMENTO ATRAVÉS DE CHATBOT

Diego Prass Cardoso
Faculdades Integradas de Taquara – Faccat – Taquara – RS – Brasil
diegoprass@faccat.br

Everton Luis Berz
Professor Orientador
Faculdades Integradas de Taquara – Faccat – Taquara – RS – Brasil
everton@faccat.br

RESUMO

O objetivo deste artigo é apresentar um fluxo alternativo de atendimento ao cliente, com sugestões e referência à empresa Insoft4 da cidade Sapiranga, Rio Grande do Sul. Atualmente muitos colaboradores estão sobrecarregados devido à alta demanda em diferentes canais de atendimento, que geram diferentes demandas, sendo muitas delas um esforço repetitivo, como responder às mesmas dúvidas, o que gera um desperdício de tempo produtivo para a empresa. Nos dias atuais, uma solução para este problema é a adesão a processos automatizados em canais de atendimento ao cliente e, através destas automações, fazer uso do Machine Learning e Processamento de Linguagem Natural para o desenvolvimento de um agente conversacional. Deste modo, torna-se possível atender diferentes necessidades dos clientes e obter como resultado a redução dos esforços repetitivos e a minimização do envolvimento humano nestas situações.

Palavras-chave: Machine Learning. Processamento de linguagem natural. Chatbot.

ABSTRACT

The purpose of this article is to present an alternative flow of customer service, with suggestions and references to the Insoft4 company in the city of Sapiranga, Rio Grande do Sul. In the present time, many employees have been overloaded due to high demands in different service channels, which generates different demands, being many of them a repetitive effort, such as answering questions that causes waste of productive time for the company. Nowadays, a solution to this problem is the accession to automated processes in their service channels and, through these automations, to make use of Machine Learning and Natural Language Processing for the development of a conversational agent. Thus making it possible to meet different customer needs and obtain, as a result, a reduction in repetitive efforts and a minimization of human involvement in these situations.

Keywords: Machine Learning. Natural language processing. Chatbot.

1 INTRODUÇÃO

A dinâmica de atendimento ao cliente vem mudando, empresas estão buscando melhorar seu perfil de atendimento e utilizando estratégias inovadoras para ganhar espaço no mercado, o qual está a cada dia mais competitivo e impactado pelo resultado da atual revolução digital. A maneira com que os serviços são contratados, prospectados e comercializados têm impactado diretamente as decisões dos clientes em seu dia a dia (FACEBOOK, 2018).

Na busca pela satisfação dos clientes, as empresas têm investido no atendimento ao consumidor através do desenvolvimento de automações, evoluindo do modelo de recepções mais clássicas, como visitas físicas ou chamadas telefônicas, para o desenvolvimento de inteligências artificiais com o uso de Agentes Conversacionais (AC) (FERREIRA *et al.*, 2021).

Segundo o Facebook (2018), sete em cada dez empresas pesquisadas acham que estão se comunicando de forma eficaz com os clientes, mas somente dois em cada dez consumidores pesquisados concordam. Atualmente, o modelo comercial resulta em uma alta demanda de atendimento e reunião, por telefone ou *e-mail* e conversas com analistas e equipe de atendimento. Disponibilizar informações através de *Machine Learning* e Processamento de Linguagem Natural é um meio de ajudar o usuário durante o processo de decisão.

Este artigo apresenta os resultados do desenvolvimento de um *portal web* com integração de atendimento através de um *chatbot* como uma alternativa sugerida a comunicação ao cliente e, assim, criando um novo canal de comunicação e relacionamento para o Grupo Insoft4.

O estudo abrange o desenvolvimento dos contextos e base de conhecimento para implantação de chatbot para atendimento de dúvidas e triagem comercial. Integração a diferentes APIs, como também a avaliação de *frameworks* de *chatbot* existentes no mercado em relação a implementação, funcionalidades e desempenho.

O Grupo Insoft4 é atualmente formado por duas empresas, a Insoft4 Informática e a Insoft4 APS. Juntas as empresas contam com 40 colaboradores que são distribuídos entre os setores de desenvolvimento, suporte, consultoria de implantação e administração, onde ficam as áreas de direção, financeiro, marketing e comercial, responsáveis pela parte estratégica operacional.

Atualmente localizada na cidade de Sapiranga, Rio Grande do Sul, o grupo Insoft4 foi fundado no ano 2000. A empresa atua no mercado de *Hardware* e *Software*, comercializando equipamentos de acesso e no desenvolvimento de sistemas para controle de ponto, acesso, segurança e gestão de terceiros. O principal produto da empresa é seu *Software* de processamento de folha de ponto.

2 FUNDAMENTAÇÃO TEÓRICA

2.1 Processamento de linguagem natural

O uso do termo Processamento de Linguagem Natural, no inglês *Natural Language Processing* (NLP), é um dos subcampos de estudo da Inteligência Artificial (IA), que tem como objetivo ensinar as máquinas a entender e compreender a linguagem humana. O NLP explora o conteúdo textual, proveniente da fala humana, e faz o processamento para a linguagem de máquina (FRANCONI, 1994). O tema vem sendo cada vez mais pesquisado, tendo sua aplicação em campos como análise de sentimento para entender as opiniões dos clientes e encontrar *insights* úteis sobre o produto e a experiência do usuário, porém, não é um conceito simples, pois a linguagem humana possui características como a ambiguidade, que dificultam o aprendizado. Dentro do NLP, podemos considerar dois sub-campos: *Natural Language Understanding* (NLU) e *Natural Language Generation* (NLG).

O NLU tem o papel de realizar a análise frase a frase, juntando em grupos de alta similaridade para conseguir entender qual é a intenção humana, enquanto o NLG tem a responsabilidade de identificar e responder de forma apropriada a cada situação.

Em um sentido mais amplo, pode-se dizer que NLP tem o objetivo de fazer o computador se comunicar através da linguagem humana, que é complexa e possui diferentes níveis de compreensão (FRANCONI, 1994). Podemos analisar esses níveis no Quadro 1 a seguir:

Quadro 1 - Técnicas de radicalização

FONÉTICO E FONOLÓGICO	do relacionamento das palavras com os sons que elas produzem
MORFOLÓGICO	da construção das palavras a partir de unidades de significado primitivas e de como classificá-las em categorias morfológicas
SINTÁTICO	do relacionamento das palavras entre si, cada uma assumindo seu papel estrutural nas frases, e de como as frases podem ser partes de outras, constituindo sentenças
SEMÂNTICO	do relacionamento das palavras com seus significados e de como elas são combinadas para formar os significados das sentenças
PRAGMÁTICO	do uso de frases e sentenças em diferentes contextos afetando o significado

Fonte: Gonzalez (2003, p. 4).

A representação do significado de uma sentença, independente de contexto, é obtida através de sua forma lógica. A forma lógica codifica os possíveis sentidos de cada palavra e identifica os relacionamentos entre palavras e frases (FRANCONI, 1994).

2.2 Chatbot

A comunicação entre homens e máquinas tem recebido um papel importante e cada vez maior nos últimos anos. Alguns exemplos são as soluções de *Chatbots* e Assistentes Virtuais Inteligentes. Algumas empresas como Uber, Magazine Luiza, TIM, entre outras, possuem desenvolvimento e implementação de *chatbot* para automatizar os seus serviços internos e proporcionar uma comunicação mais rápida entre seus clientes. Alguns Assistentes Virtuais aprendem de forma inteligente à medida que vão recebendo interações com o Google Assistant, Siri, Alexa e Cortana. Utilizando Assistentes Virtuais e *Chatbots*, o usuário tem a vantagem de receber rápido acesso a informação, solucionando pequenos problemas ou dúvidas. Além disso, um *chatbot* pode estar disponível para atendimento ao cliente 24 horas por dia e 7 dias por semana (GONZALES, 2003).

2.2.1 Arquitetura de um *Chatbot*

A interação entre um *chatbot* e um humano passa por diferentes estágios. O primeiro é iniciado através da interação, em uma janela de *chat*, transformadas em requisição para que

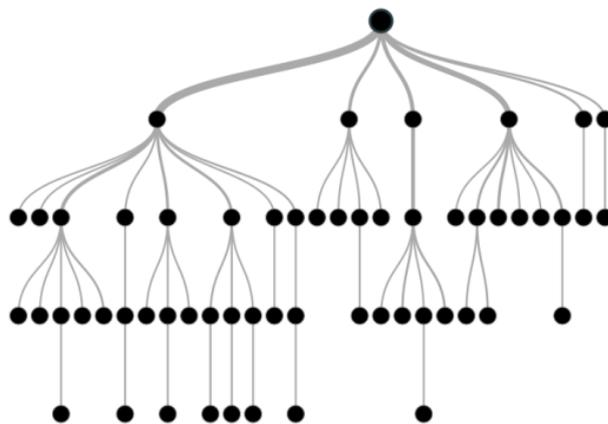
sejam analisadas e automaticamente respondidas. Para garantir que A está se comunicando com B, se utiliza os *webhooks* que, resumidamente, são validações de envio e retorno entre chamadas na *web*, a partir deles é possível garantir a identificação e validação do recebimento das informações entre os dois sistemas (DANTAS, 2021).

Em segundo lugar, o retorno que o usuário recebe depende de um conjunto de modelos. Estes modelos são treinados utilizando uma base de conhecimento desenvolvida através de um conjunto de determinadas intenções que são interpretadas para poder responder de forma correta um questionamento.

2.2.2 Árvore de decisão

Uma árvore de decisão é um meio simples e eficaz de calcular e responder os questionamentos de um usuário para um chatbot, já que se utiliza de um conjunto de conexões baseando-se na abordagem *dividir para conquistar* (QUINLAN, 1993). Repetidas interações são, então analisadas, calculadas e divididas em diferentes conjuntos, que após treinados, são divididos em subconjuntos de classe, até que ocorra a conversão para um único determinado contexto. A Figura 1 apresenta uma classificação baseada em uma árvore de decisão, na qual são ilustradas no topo a *camada de entrada* que representam as interações de um usuário, ao centro as *camadas ocultas de contexto*, que são as ações mapeadas da base de conhecimento, e na base a *camada de retorno*, que representa a ação de um retorno, é uma resposta apresentada ao usuário.

Figura 1 - Exemplo de um classificador usando a árvore de decisão



Fonte: Quinlan (1993), adaptado pelo autor (2021).

Portanto, a árvore de decisão é um mapa de resultados possíveis, “que podem solucionar questões levantadas para um *chatbot*, ou seja, é o que guia um *bot* a encontrar a resposta para as perguntas de clientes” (SILVA, 2021¹), e assim responder a questão de forma aceitável.

2.2.3 Conceitos genéricos utilizados

Com o objetivo de ampliar o alcance da tecnologia de *chatbot* para o mercado, foram definidos alguns conceitos que são utilizados pelas empresas para trabalhar com diferentes níveis e modelos de NLP e aprendizado de máquina. Segundo Cardoso (2017) estes principais conceitos são entidades, intenções e diálogo.

- Entidades: é o substantivo relacionado ao desejo que foi detectado pelo *chatbot*.
- Intenção: é o desejo que o *chatbot* perceberá que o usuário possui ao enviar uma mensagem específica.
- Diálogo: é onde se cria as respostas que o *chatbot* retornará ao detectar uma intenção e/ou uma entidade.

Utilizando sentenças que façam sentido com as entidades, intenções e diálogos criados, o serviço retorna às entidades e a intenção que o *chatbot* reconheceu na sentença, realizando ajustes caso o *chatbot* reconheça entidades ou uma intenção possivelmente incorreta (CARDOSO, 2017). Além destes conceitos principais, são abordados também contextos das *Ações*, *Respostas* e *Digressões* que podem ser vistos de forma resumida de acordo com o Quadro 2.

Quadro 2 - Conceitos de um *chatbot*

Actions (Ações)	Coisas que seu <i>bot</i> pode fazer. ex: fazer uma consulta
Responses (Respostas)	São os textos em si que seu <i>bot</i> pode dizer
Digressions (Digressões)	Ocorre quando o usuário está no meio de um fluxo de diálogo e muda abruptamente para outro assunto

Fonte: IBM CLOUD²

¹ Como o texto original, extraído da Internet, não apresentava paginação, não foi possível, neste trabalho, indicar a página da citação direta (N. A.).

² IBM CLOUD, Controlando fluxo de conversa. Disponível em: “<https://cloud.ibm.com/docs/assistant?topic=assistant-dialog-runtime&locale=pt-BR>”. Acesso em 05 de ago. de 2021.

Ao reconhecer a intenção expressa na entrada de um cliente, um serviço de *chatbot* pode escolher o fluxo de diálogo correto para responder a isso.

Existe mais um conceito que se chama história, do inglês, *story*, uma história é uma representação de uma conversa entre um usuário e um assistente conversacional, convertida em um formato específico onde as entradas do usuário são expressas como intenções (e entidades quando necessário), enquanto as respostas e ações do assistente são expressas como nomes de ações (RASA, 2021). A seguir a Figura 2 representando uma história.

Figura 2 - Exemplo de representação história



```

data > stories.yml
107
108   História: apresentar produto
109     intenção: saudação
110     ação: responder_saudação
111     intenção: iniciar conversa
112     ação: responder_conversa
113     intenção: produto
114     ação: responder_sobre_produto
115     intenção: agendar reunião
116     ação: responder_Agendar_reunião
117     intenção: agradecimento
118     ação: responder_Agradecimento
119
  
```

Fonte: RASA (2021), adaptado pelo autor (2021).

Pode-se dizer então que histórias são exemplos de conjuntos de intenções e respostas, ou seja, o *chatbot* vai escolher a melhor resposta para interagir com os usuários a partir desses exemplos, que nada mais são que uma previsão a uma determinada conversa implementados de forma lógica através de respostas para uma sequência de possíveis pontos de interações anteriormente identificados. (LAPPIS, 2021)

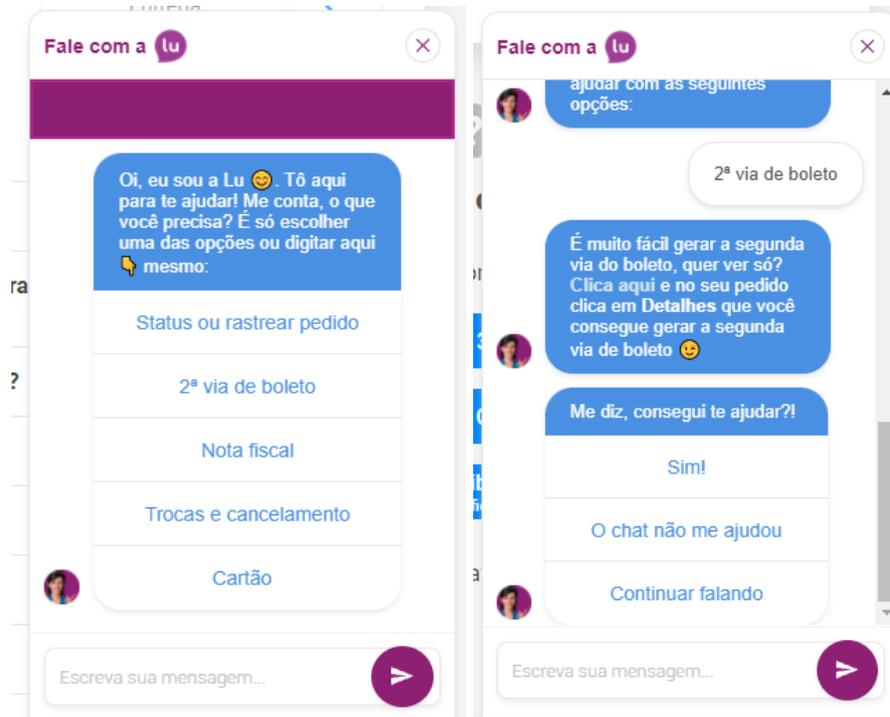
3 PROJETOS RELACIONADOS

Criada pela Magazine Luiza, Lu³ é um *chatbot* que auxilia o cliente com questões básicas. Cabe observe que neste caso, a aplicação já sugere tópicos mais recorrentes para

³ MAGAZINE LUIZA. Portal da Lu. Disponível em: <https://www.magazineluiza.com.br/portaldalu/>. Acesso em: 06 ago. 2021.

auxiliar o usuário e, na Figura 3, podemos ver a continuidade do que foi selecionado: “Segunda via de Boleto”:

Figura 3 - Chatbot Lu – Exemplo da Magazine Luiza



Fonte: Magazine Luiza (2021a).

A Casas Bahia⁴ desenvolveu o “Bahianinho”, nome dado ao seu *chatbot*. A partir dele, o cliente pode escolher por categoria entre diferentes produtos que desejar, como também analisar ofertas de produtos, sendo direcionado via link para a página do produto através da interação por *chat* em seu site, mídias sociais e canais de atendimento. (QUALI, 2020)

O Uber⁵, através da mídia social Facebook Messenger⁶, sem que seja preciso entrar no aplicativo, auxilia o usuário. Isso só é possível com a implantação de um *chatbot* que estipula a comunicação entre os motoristas e clientes mais próximos via rede social.

Estes modelos se enquadram com os *chatbots* orientados a tarefas, similar ao que está sendo proposto neste trabalho como sugestão, o Grupo Insoft4, como uma forma de atendimento alternativo, diversificando assim os modelos de atendimento. A implementação do Marvin, desenvolvido na plataforma IBM Watson Assistance, será um agente

⁴ CASAS BAHIA. Disponível em: <https://www.casasbahia.com.br/?s=bahianinho>. Acesso em: 05 set. 2021.

⁵ UBER. **Quem somos**. Disponível em: <https://www.uber.com/br/pt-br/about/>. Acesso em: 07 set. 2021.

⁶ MESSENGER. Disponível em: <https://www.messenger.com/>. Acesso em: 20 set. 2021.

conversacional para a página do portal *web*. Como uma alternativa de código aberto e atendimento através de integrações das mídias sociais, será analisada a implementação do RASA, ambos através portal *web*.

4 METODOLOGIA

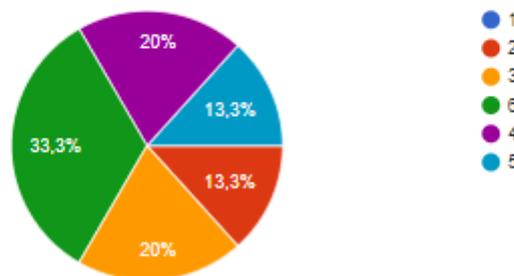
4.1 Entrevistas

Para identificar os processos envolvidos na realização do atendimento, as entrevistas foram realizadas ao longo dos anos de 2020 e 2021 com consultores comerciais, suporte técnico e atendimento ao cliente da empresa Insoft4 na cidade de Sapiranga, Rio Grande do Sul. Elas objetivam detalhar o funcionamento do fluxo de trabalho enquanto é realizado o atendimento a usuários e clientes, identificando o fluxo de preparação, coleta de informações e após-levantamento. A entrevista foi guiada de forma não estruturada e não dirigida, que, segundo Marconi e Lakatos (2003), procura explorar as diversas possibilidades que o contato entre entrevistado e entrevistador pode fornecer.

Na Figura 4 a seguir, podemos observar um exemplo de pergunta realizada aos entrevistados que teve um total de quinze participantes.

Figura 4 - Questionário sobre reunião

Para agendar uma reunião, quantas ligações ou trocas de e-mail você costuma fazer?
15 respostas



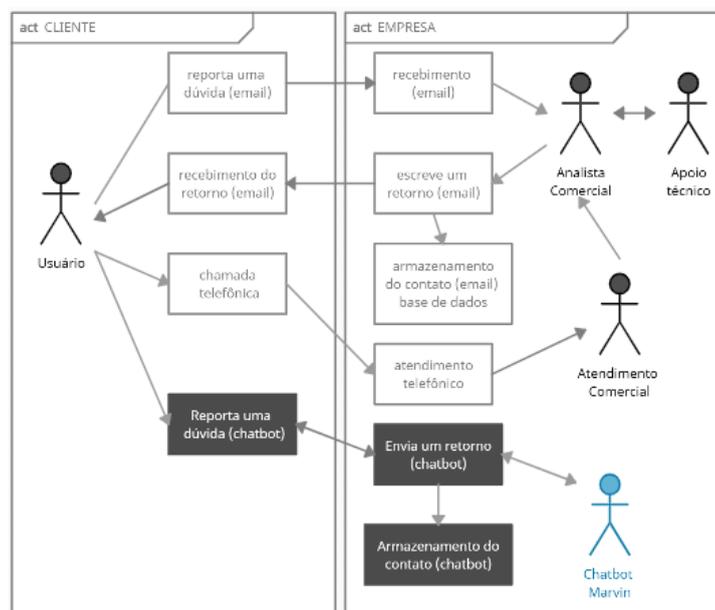
Fonte: Dados da pesquisa (2021).

Pode-se observar, de acordo com os entrevistados, que o modelo de atendimento através de reuniões, que poderiam ter sido realizadas por email, resulta em uma grande perda de tempo, pois responder perguntas de forma repetitiva é algo que demanda energia e não geram resultados satisfatórios ao longo e médio prazo, em consideração as demandas diárias.

4.2 Levantamento e Análise de Requisitos

Após a realização das entrevistas, iniciou-se a observação do processo de trabalho, e a partir desta análise e das percepções obtidas, foi possível desenhar um fluxo, contando com a criação do portal *web*, que auxilie nos contatos e leve informações referentes aos produtos e serviços prestados pela empresa, bem como utilizar dúvida e respondê-las de forma automatizada. Conforme pode ser observado, a seguir, na Figura 5. Onde podemos observar que o *chatbot* atua diretamente com o usuário.

Figura 5 - Fluxo de trabalho proposto



Fonte: Elaborado pelo autor (2021).

Após a definição do modelo foram então levantados requisitos. Eles foram propostos através das informações obtidas como resultado da etapa de entrevista (PRESSMAN, 2011). Os *requisitos funcionais* são RF001: Deve ser possível acessar a *web page*; RF002: Deve ser possível que o usuário abra o *chatbot*; RF003: Deve ser possível pedir informações de produto; RF004: Deve ser possível solicitar falar com um humano/atendente; RF005: Deve ser possível agendar uma reunião; RF006: Deve ser possível pedir informações da LGPD; RF007: Deve ser possível pedir informações básicas da empresa. E os *requisitos não funcionais* são RN001: Deve ser capaz acessar pelo *mobile*; RN002: Disponibilidade em rede HTTPS; RN003: Deve possuir conceitos de responsividade; RN004: O sistema não deve levar mais que 10 segundos para responder; RN005: Notificações por inatividade após 5 minutos.

4.3 Caracterização do Chatbot

Como metodologia para criar a personalidade do *chatbot* foi utilizado o MBTI (do inglês, *Myers-Briggs Type Indicator*), criado durante a segunda guerra mundial pelas professoras norte-americanas Isabel Briggs Myers e sua mãe, Katharine Briggs. (MAGALHÃES, 2021) Para criar a personificação do *chatbot* foi então, inicialmente definido o seu nome e sexo, ele se chamaria Marvin em homenagem ao livro “O Guia do Mochileiro das Galáxias”, de Douglas Adams. Nele existe um robô chamado Marvin que é descrito como o robô mais inteligente da galáxia, um protótipo GPP (genuína personalidade humana). Podemos analisar os detalhes dos papéis atribuídos para *persona* no Quadro 3:

Quadro 3- Definição da persona: Marvin

	Definição visual da persona
Nome:	Marvin
Gênero:	Masculino
Cargo/Ocupação:	Assistente virtual da Insoft
Meio de comunicação:	Chatbot
Objetivo da persona:	Responde a dúvidas e auxilia usuário com agendamento de reunião, falar com o vendedor adequado a seu perfil e informações sobre os produtos.
Desafios da persona:	Aprender as variedades sintáticas e dialetais dos usuários para melhoramento da comunicação, estabelecer uma interação mais natural e confortável para o usuário compreendendo as intenções/dúvidas do usuário. Auxiliando na comunicação de forma rápida e eficiente.

Fonte: Elaborado pelo autor (2021), adaptado de MBTI.

Os principais papéis definidos para serem associados a construção da personalidade das respostas de Marvin foram, seguindo perfis da MBTI⁷, o “Protagonista⁸”, “Executivo⁹” e

⁷ MAGALHÃES, Paulo. **Teste de personalidade MBTI: você conhece essa metodologia?** 2018. Disponível em: <https://blog.trello.com/br/teste-de-personalidade-mbti>. Acesso em: 16 ago. 2021.

⁸ Protagonista: líder inspirador e carismático, que hipnotiza a audiência.

⁹ Executivo: administrador excelente, gerência de coisas e pessoas.

“Consu¹⁰”, que em conjunto em que cada elemento possui diferentes características de personalidade que foram definidas para ser utilizado nas interações do *chatbot* com os usuários.

4.4 Metodologia de desenvolvimento

A metodologia adotada foi a metodologia ágil e para o desenvolvimento foi utilizado o Kanban, ele é um método proveniente do conceito de uma metodologia ágil, este método possui diferentes etapas como a de inspeção, adaptação, interação e incremento. Kanban oferece um método livre para que a pessoa possa organizar tarefas por etapas, mantendo assim a atenção centrada na entrega, segmentando as tarefas por importância e finalidade, possibilitando assim, realizar todo o projeto em pequenas etapas (SILVA; SANTOS; SANTOS NETO, 2013).

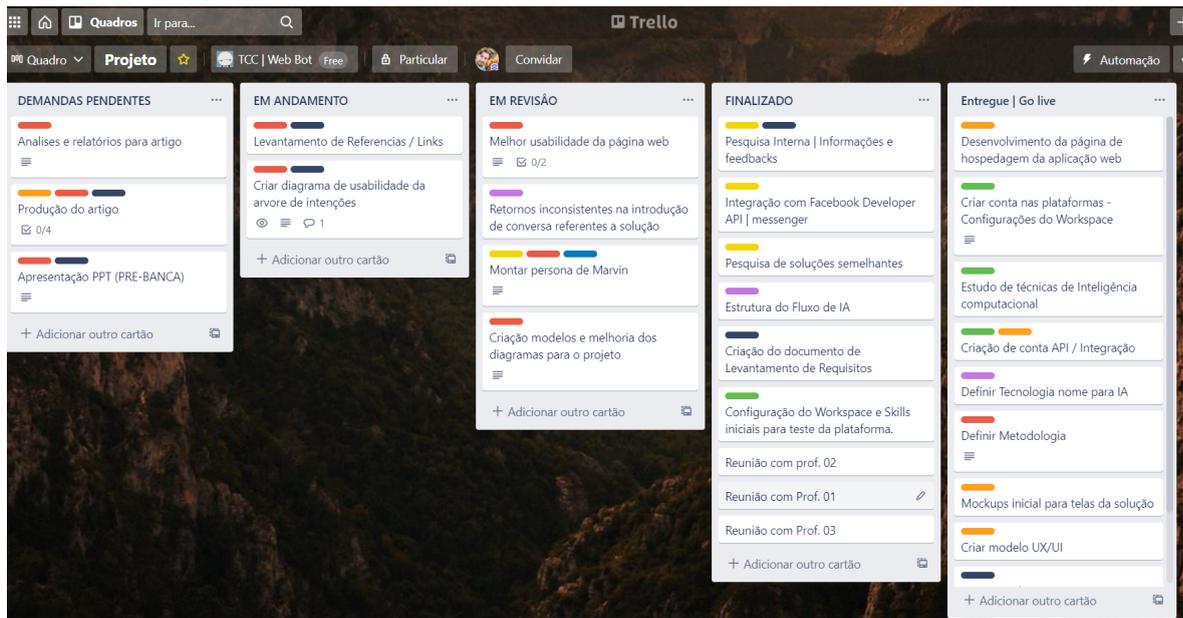
A implementação do Kanban, pode ser resumida em três diferentes etapas: primeiramente é a visualização dos processos como um todo. A segunda etapa é trabalhar a atenção na demanda em foco no processo atual. E a terceira, o gerenciamento dos tempos de entrega, no inglês, “*deadlines*”, ou seja, tempo útil disponível para determinada atividade de uma fase até outra, até que enfim chegue na etapa final que é a entrega (PAHUJA, 2020).

Como forma de operacionalizar o Kanban, foi utilizada a ferramenta Trello¹¹, que pode ser visto na Figura 6 (p.13). As tarefas eram criadas com os “cartões” e seu status atualizado conforme o desenvolvimento, apresentando uma visão geral do andamento em cada etapa do projeto.

¹⁰ Consu: pessoa extremamente atenciosa, social e popular, sempre ponto para ajudar.

¹¹ TRELLO. Disponível em: <https://trello.com/>. Acesso em: 07 ago. 2021.

Figura 6 - Acompanhamento do projeto com o Trello



Fonte: Adaptado de Trello para este trabalho (2021).

Como apresentado na Figura 6, podemos ver as seguintes etapas do fluxo:

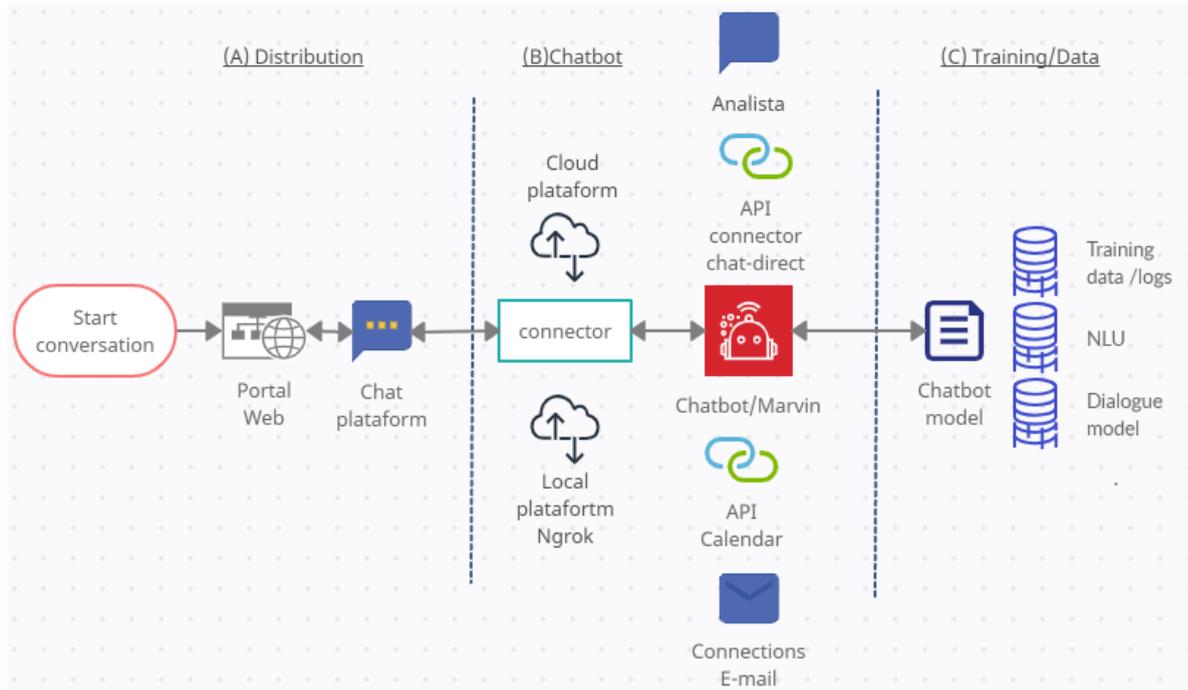
- Ideias: um grupo de ideias e demandas que são selecionadas para entrega.
- Análise de viabilidade: análise levando em relação a viabilidade de implementação.
- Desenvolvimento: trabalho desenvolvido nesta etapa.
- Testes: itens passando por validação de sua qualidade.
- Finalizado: itens que passaram por todo o fluxo e já podem ser considerados finalizados.

4.5 Modelagem do sistema

A partir da análise de requisitos e das percepções obtidas, foi possível imaginar um novo fluxo, contando com a criação e a utilização de um modelo de arquitetura independente do uso de um *framework* para ilustrá-lo. As etapas foram segmentadas em 3 sessões sendo elas: “A” portal *web* integrado a um canal de *chat*, “B” definição dos componentes e configuração de um ambiente de trabalho, local ou em nuvem, por exemplo, para integração local com o *site* foi utilizado o serviço Ngrok¹², que basicamente cria um túnel seguro para rede local, e “C” que seria o local de trabalho onde estão guardados os modelos de treinamento do *chatbot*, como pode ser visto na Figura 7 (p. 14), a seguir:

¹² NGROK. Disponível em: <https://ngrok.com/>, Acesso em: 5 jun. 2021.

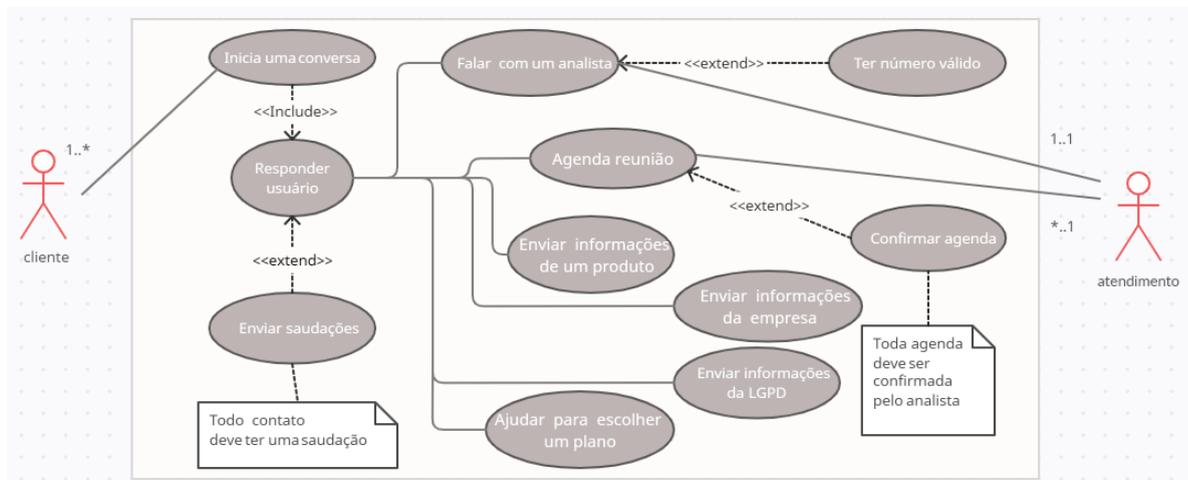
Figura 7 - Fluxo independente de *framework* para o chatbot



Fonte: Elaborado pelo autor (2021).

Olhando o quadro em panorama amplo se pode obter uma ideia geral do desenvolvimento de um todo do projeto, para ilustrar o contexto completo pode-se consultar o diagrama de caso de uso apresentado na Figura 8. Nele é possível ver o papel de cada ator no contexto das interações.

Figura 8 - Diagrama de caso de uso do atendimento



Fonte: Elaborado pelo autor (2021).

Há dois atores que irão interagir com a página *web*, o Usuário e o Analista. O usuário é quem irá realizar as interações, perguntas, com a página. A partir dela será possível abrir a janela de atendimento, onde ele terá algumas de suas intenções mapeadas, como “*Pedir informação sobre um produto*”, “*Pedir informação sobre a empresa*” ou “*Pedir informações sobre a LGPD*” entre outras, e onde a análise da sentença será classificada para retornar uma resposta qualificada ao assunto mencionado.

Alguns cenários principais esperados são: “*falar com uma pessoa*”, que direciona a pessoa através da API¹³ do WhatsApp¹⁴ para iniciar uma conversa com um consultor comercial, pedir “*ajuda para escolher um plano*”, nesse caso o usuário responderá a algumas perguntas para que o grau de necessidade possa ser calculado, recebendo de forma personalizada o plano adequado para ele(a), ou “*criar uma agenda*”, que então utiliza a integração da API do Microsoft Teams¹⁵ através da integração de contas do serviço Callendly¹⁶ para verificar os dias e horários disponíveis da agenda do setor de atendimento e, de forma consultiva, apresentar e demonstrar o benefício do produto ou serviço. O agendamento ocorre mediante confirmação por *e-mail*, após, a data e horário da reunião é confirmada, como ilustrado a seguir no diagrama da Figura 9 (p. 16).

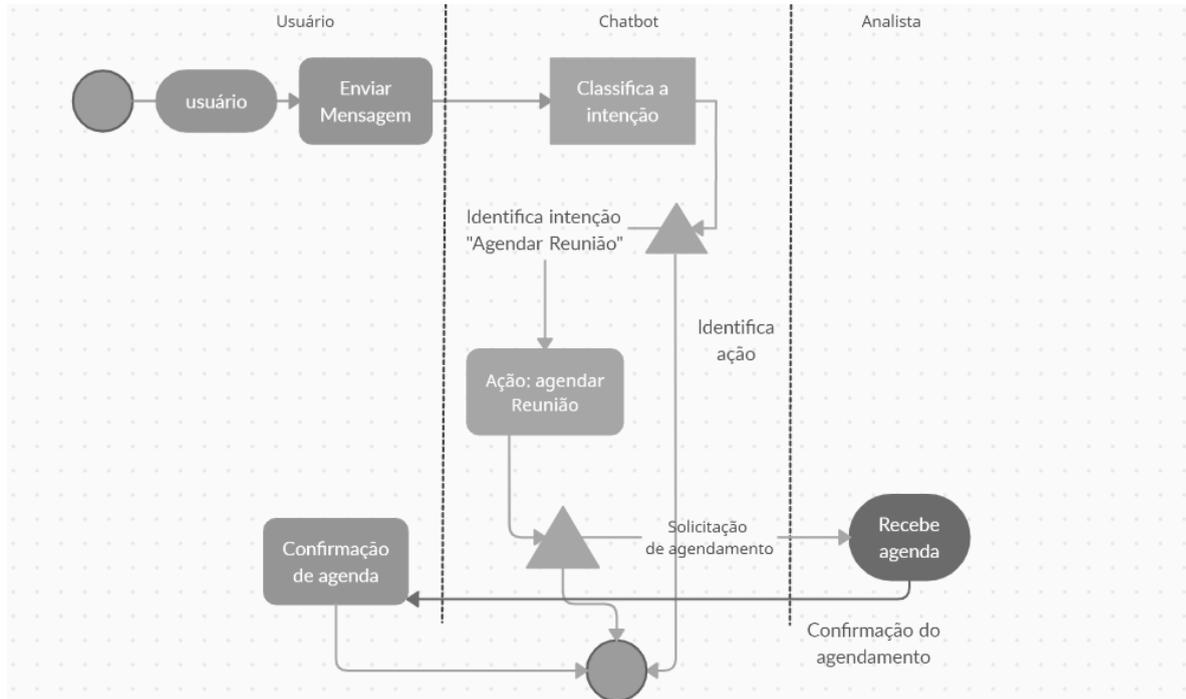
¹³ API, do inglês *Application Programming Interface*.

¹⁴ WHATSAPP. Disponível em: <https://www.whatsapp.com/>. Acesso em: 05 de ago. 2021.

¹⁵ MICROSOFT TEAMS. Disponível em: <https://www.microsoft.com/pt-br/microsoft-teams/>. Acesso em: 12 ago. 2021.

¹⁶ CALLENDLY. Disponível em: <https://calendly.com/pt>. Acesso em: 10 jun. 2021.

Figura 9 - Diagrama de atividade: Agendar reunião



Fonte: Elaborado pelo autor (2021).

4.6 Tecnologias

O projeto Marvin adotou como tecnologias de desenvolvimento para seu portal *web* o HTML, CSS e JavaScript como principais linguagens de implementação, sendo consideradas como as principais estruturas para construção de páginas e documentos eletrônicos para a internet (MAUJOR, 2021).

Para construção dos dados foi utilizado o JSON e YAML. E como ambiente de desenvolvimento foi utilizado Visual Studio Code¹⁷ (IDE)¹⁸, que é uma solução da Microsoft.

4.6.1 Uso do *framework* Bootstrap

O *framework front-end*, Bootstrap é um conjunto de bibliotecas que tem como objetivo simplificar o desenvolvimento de projetos *web* responsivos, ou seja, de página *web* capaz de modelar diferentes tamanhos de resolução (BOOTSTRAP, 2021). O Bootstrap acelera o trabalho do desenvolvedor por meio de padrões pré-definidos de diferentes telas, com uma

¹⁷ VISUAL STUDIO CODE. Disponível em: <https://code.visualstudio.com/>. Acesso em: 15 jun. 2021.

¹⁸ IDE, do inglês *Integrated Development Environment* ou Ambiente de Desenvolvimento Integrado, é um programa de computador que reúne características e ferramentas de apoio ao desenvolvimento de software.

grade de diferentes componentes disponíveis, tais como: botões, barras de navegação, ícones, menus, dentre diversos outros componentes. Assim, é possível desenvolver de forma rápida de pequenos protótipos, até versões finalizadas de projetos. Por essas razões, o presente trabalho utiliza o Bootstrap (BOOTSTRAP, 2021).

4.7 Processamento de linguagem natural com IBM Watson Assistant

O algoritmo de processamento de linguagem natural para o agente conversacional desenvolvido através do IBM Watson Assistant, presente na infraestrutura atual da IBM Cloud¹⁹, foi desenvolvido para atender a crescente demanda de uso da inteligência artificial nas organizações. As funcionalidades do WATSON foram projetadas para diminuir a curva do processo de automação da inteligência artificial, oferecer precisão no processamento de linguagem natural, ampliando assim a aplicabilidade no uso da inteligência artificial (IBM 2021).

O IBM Watson Assistant utiliza técnicas de aprendizado de máquina para responder às interações do usuário com precisão, se utilizando de diversos conjuntos de bases de dados segmentados por cenários. O coração da inteligência artificial, do inglês *CORE*, do Watson Assistant, é projetado para analisar e identificar corretamente as intenções do usuário de modo natural, como em uma conversa entre humanos. Segundo Jeanluca (2020) algumas habilidades do IBM Watson Assistant são Discovery, Reading Comprehension, FAQ Extraction e o Modelo de classificação de intenções (IBM, 2021).

Atualmente, o IBM Watson possui suporte a vários idiomas, entre eles o PT-BR. A IBM vem ampliando suas fronteiras com o objetivo de construir soluções de processamento de linguagem natural para um gama maior de empresas. O algoritmo de detecção de intenções IBM Watson Assistant possui maior assertividade comparado ao de outras soluções proprietárias e comparado às maiores soluções atuais de código aberto, em julho de 2021, foi publicado um artigo que compara o desempenho de forma técnica do IBM Watson Assistant a plataformas semelhantes como do Google, Microsoft e de código aberto RASA. De acordo com os resultados do *benchmark*²⁰, o Watson Assistant possui 92.9 pontos percentuais de

¹⁹ IBM. **Plataforma Watson**. Disponível em: <https://us-south.assistant.watson.cloud.ibm.com/>. Acesso em: 22 ago. 2021.

²⁰ Em computação, *benchmark* é o ato de executar um programa de computador, um conjunto de programas ou outras operações, a fim de avaliar o desempenho relativo de um objeto, normalmente executando uma série de testes (QI *et al.*, 2021).

precisão, enquanto seu concorrente de linguagem aberta RASA possui 88.1 pontos, segundo os testes de performance e treinamento realizados (QI *et al.*, 2021).

4.8 Processamento de linguagem natural com RASA

O RASA é um *software* que foi desenvolvido na Alemanha, suas primeiras versões foram publicadas na comunidade de código aberto por volta de 2016. É uma ferramenta de inteligência artificial utilizada para criar *chatbots*. Como muitas outras tecnologias, o RASA vem evoluindo de maneira constante, e sendo implementado por diversas empresas como Adobe, Toyota e BMW (RASA, 2021).

O núcleo da inteligência artificial, do inglês *core*, do RASA, tem a função de escolher a melhor ação a ser tomada a partir do histórico da conversa e da atual intenção do usuário, seja ela uma resposta dentro do contexto atual da conversa ou uma ação realizada em um novo contexto. Isso garante que o usuário esteja sempre no controle da conversa, ao invés de ser conduzido por uma árvore de decisões (LAPPIS, 2021).

Algumas informações sobre o projeto RASA é que ele foi totalmente programado em *python* e é mantido em repositórios de código aberto. Isso quer dizer que o projeto recebe apoio de pessoas, não remuneradas, que de forma voluntária ajudam no desenvolvimento de melhorias para o projeto, enviando, por exemplo, correções e sugestões de novas funcionalidades para o projeto. Atualmente, ele possui um fórum na internet com mais de 12 mil pessoas trocando experiências, dúvidas e informações (RASA, 2021).

5 DESENVOLVIMENTO

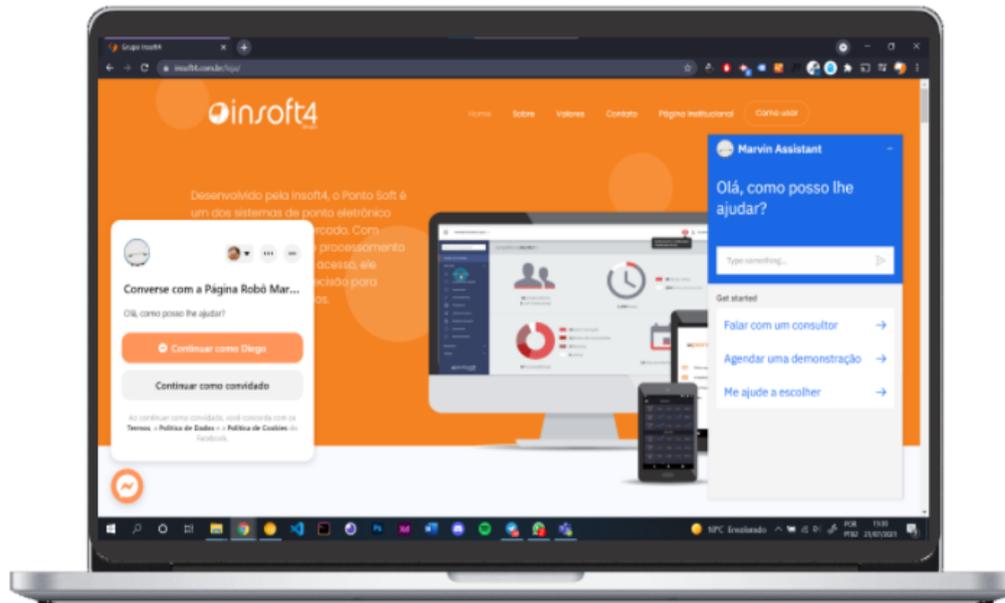
5.1 Integração do Agente Inteligente com *Website*

A medida em que o Marvin foi ficando cada vez mais inteligente e com uma base de conhecimento maior, surgiu a necessidade de construir um *Website* para poder apresentar o projeto em funcionamento, recebendo interações diretamente dos usuários finais.

O Portal *web* teve duas versões, a primeira foi apenas um escopo, feito apenas com HTML e CSS, somente uma versão de validação dos *Scripts* de integração com a Interface de programação de aplicações (*API*) do IBM Watson e definido questões de usabilidade e

navegação. Na segunda versão foi incluído Javascript e Bootstrap, nessa versão o *site* foi dividido em seções, sendo:(1) Home, (2)Sobre, (3) Planos e (4)Contato. Na Figura 10, pode ser vista a sessão (1) Home.

Figura 10 - Apresentação do Portal *Web*



Fonte: Elaborado pelo autor (2021).

5.2 Criação e Inserção da Base de Conhecimento na plataforma

5.2.1 Configuração para o Watson Assistant

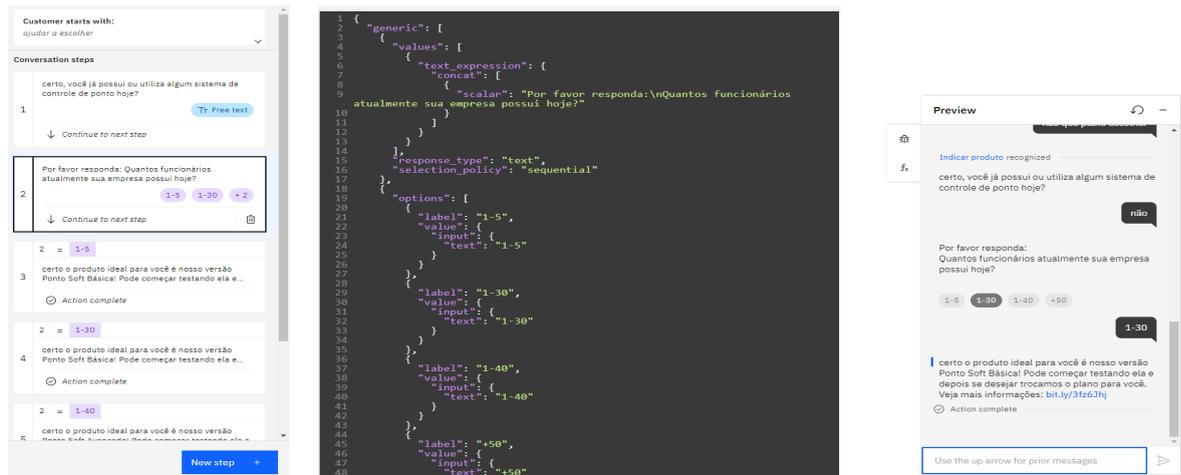
Inicialmente é necessário criar uma conta na plataforma da IBM para obter seu IBMid. A opção do plano LITE permite a utilização de 50 horas de transação de dados por mês com até mil usuários ativos, integração através de API de canal personalizado e dados salvos diretamente na plataforma da IBM Cloud, de forma gratuita.

Após as configurações iniciais é definido o nome para os atributos de habilidade, do inglês “*Skill*”, “*Atendimento ao Cliente*”, e o idioma para as interações definido como “Brazilian Portuguese”, este detalhe é crucial para o restante das configurações, pois os contextos serão analisados levando em consideração o idioma, e este não pode ser alterado posteriormente.

Começasse então a criar as ações de interação na plataforma, à medida que as ações são criadas, para cada uma delas é necessário desenvolver os passos de interação. Estes

podem ser escritos através do Sistema de Gerenciamento de Conteúdo ou diretamente na fonte, utilizando as marcações da extensão JSON²¹, na janela de construção do passo em progresso, como ilustrado na Figura 11, logo abaixo, onde está sendo apresentado o nó da intenção *escolher plano*, que considera apresentar o contexto demonstração de um produto para o usuário.

Figura 11 - Exemplos para intenção de #escolher_plano



Fonte: Elaborado pelo autor (2021).

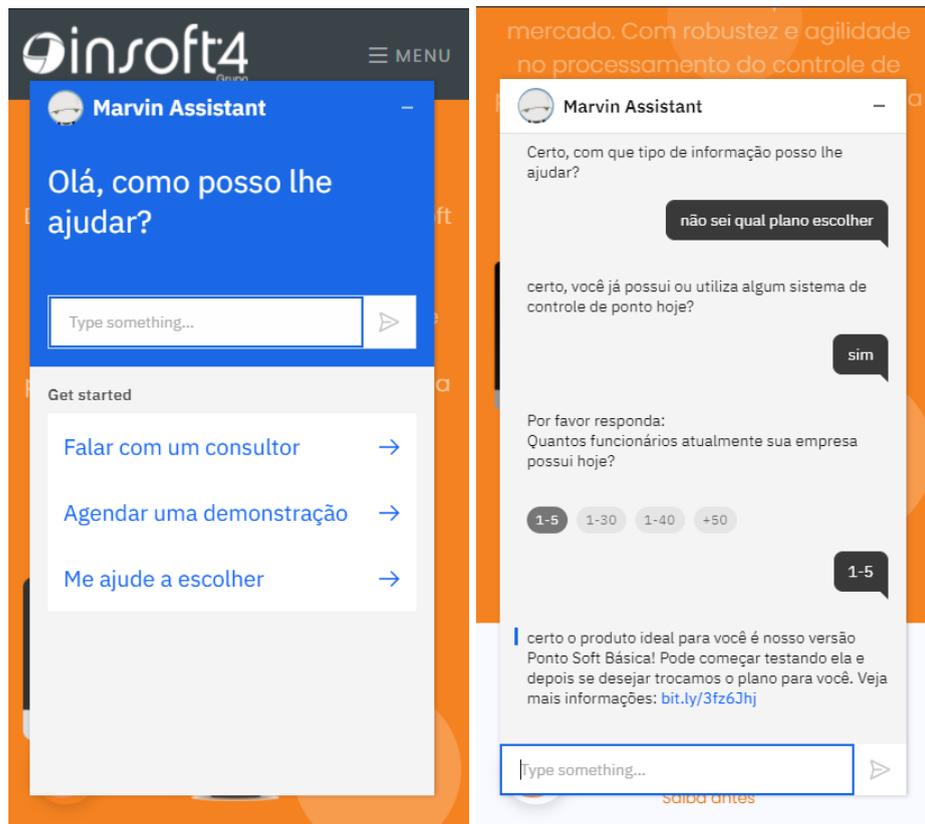
O modelo utiliza dados reais de conversação, na forma das “histórias” em seu treinamento. Como já mencionado na página 7 e exemplificado na Figura 2.

5.3 Microsserviço de comunicação e processamento de mensagem

A interface de comunicação com o usuário se dá por meio dos microsserviços. São o meio de comunicação que se utiliza do protocolo de rede *Transmission Control Protocol* (TCP) para obter a resposta da entrada fornecida pelo usuário. Assim que a solicitação é processada, esse serviço interpreta a resposta e envia para o usuário solicitante, finalizando sua etapa no processo, assim na Figura 12 (p. 21), podemos ver um exemplo de interação do usuário através da interface conversacional.

²¹ JSON, significado do inglês, *JavaScript Object Notation*, é um formato de troca de informações entre sistemas.

Figura 12 - Interação da aplicação com o usuário-interface via Watson



Fonte: Elaborado pelo autor (2021).

A comunicação com o IBM Watson Assistant ocorre por meio da biblioteca *ibm-watson*, versão 5.3.0 disponível no gerenciador. A interpretação de resposta recebida pelo serviço da IBM foi realizada segundo a documentação do Assistant na versão V2. Na figura a seguir, na Figura 13, podemos ver como foi realizada a integração do IBM Watson Assistant ao portal *web*.

Figura 13 - Exemplo de requisição

```

30 <script>
31   window.watsonAssistantChatOptions = {
32     integrationID: "e8984dc3-...", // The ID of this integration.
33     region: "us-south", // The region your integration is hosted in.
34     serviceInstanceID: "d4fda3fc-...", // The ID of your service instance.
35     onLoad: function (instance) { instance.render(); }
36   };
37   setTimeout(function () {
38     const t = document.createElement('script');
39     t.src = "https://web-chat.global.assistant.watson.appdomain.cloud/loadWatsonAssistantChat.js";
40     document.head.appendChild(t);
41   });
42 </script>

```

Fonte: Elaborado pelo autor (2021), utilizando Watson Assistant (IBM, 2021).

5.3 Infraestrutura para o RASA

Como alternativa de código aberto, ao IBM Watson, optou-se ao RASA e para construção de sua infraestrutura foi utilizada a mesma base de conhecimento da implantação do Watson, RASA se utiliza dos mesmos conceitos de Processamento de Linguagem natural e Aprendizagem de Máquina mencionados nas páginas 3 a 7 deste trabalho.

Foi utilizado como referência, para implantação, a versão 2.8.12 do RASA, versão estável mais recente no período de desenvolvimento deste trabalho. Nela foram executados os modelos, como ser visto na Figura 14, onde é apresentado o carregamento de um modelo para o *chatbot*.

Figura 14 - Execução e análise via console

```
[state 1] user intent: saudacoes | previous action name: action_listen
[state 2] user intent: saudacoes | previous action name: utter_saudacoes
[state 3] user intent: iniciar_conversa | previous action name: action_listen
[state 4] user intent: iniciar_conversa | previous action name: utter_iniciar_conversa
2021-10-25 23:12:21 DEBUG rasa.core.policies.rule_policy - There is no applicable rule.
2021-10-25 23:12:21 DEBUG rasa.core.policies.ted_policy - TED predicted 'action_listen' based on user intent.
2021-10-25 23:12:21 DEBUG rasa.core.policies.ensemble - Predicted next action using policy_2_TEDPolicy.
2021-10-25 23:12:21 DEBUG rasa.core.processor - Predicted next action 'action_listen' with confidence 1.00.
2021-10-25 23:12:21 DEBUG rasa.core.processor - Policy prediction ended with events '[]'.
2021-10-25 23:12:21 DEBUG rasa.core.processor - Action 'action_listen' ended with events '[]'.
2021-10-25 23:12:21 DEBUG rasa.core.lock_store - Deleted lock for conversation '10c0e7b6615b41d0af957313c53de1a2'.
Certo, com que tipo de informação posso lhe ajudar?
Your input -> pode me falar sobre o mobile?
```

Fonte: Elaborado pelo autor (2021).

Após a configuração dos contextos dos *Intents*, para os *dialogs*, e seus respectivos *responses*, foram definidas seus totais absolutos, do inglês *utter*, que são uma forma de tratar de forma individual cada *Intent* dentro de uma *Story*. Podemos ver na Figura 15 (p. 23), um exemplo da implementação de todos os passos de análise de uma história no RASA.

Figura 15 - Exemplo de *Story*

```

80 - story: software path
81   steps:
82     - intent: saudacoes
83     - action: utter_saudacoes
84     - intent: greet
85     - action: utter_greet
86     - intent: iniciar_conversa
87     - action: utter_iniciar_conversa
88     - intent: falar_atendente
89     - action: utter_falar_atendente
90     - intent: ponto_soft_mobile
91     - action: utter_ponto_soft_mobile
92     - intent: ponto_soft
93     - action: utter_ponto_soft

```

Fonte: Elaborado pelo autor (2021).

As interações do com *bot* foram realizadas através do *console*, de forma local. Então feita a integração com Facebook Messenger, como podemos ver na Figura 16, a aplicação valida a conexão a ferramenta *Messenger* através de uma chave secreta e um *token* de acesso, que é verificado por uma palavra de segurança definida para a validação do *Webhook*.

Figura 16 - Autenticação para Facebook

```

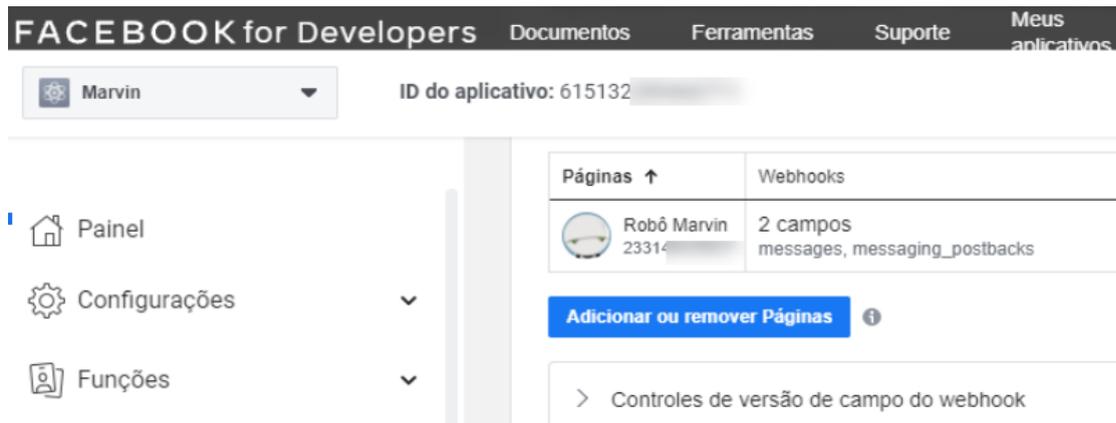
10 facebook:
11   verify: "marvin"
12   secret: "140102452f"
13   page-access-token: "EAAIvdaWaacBAKacDDajsoxhMr1ymSp3bZA9XjN72dB1"
14

```

Fonte – Autor (2021)

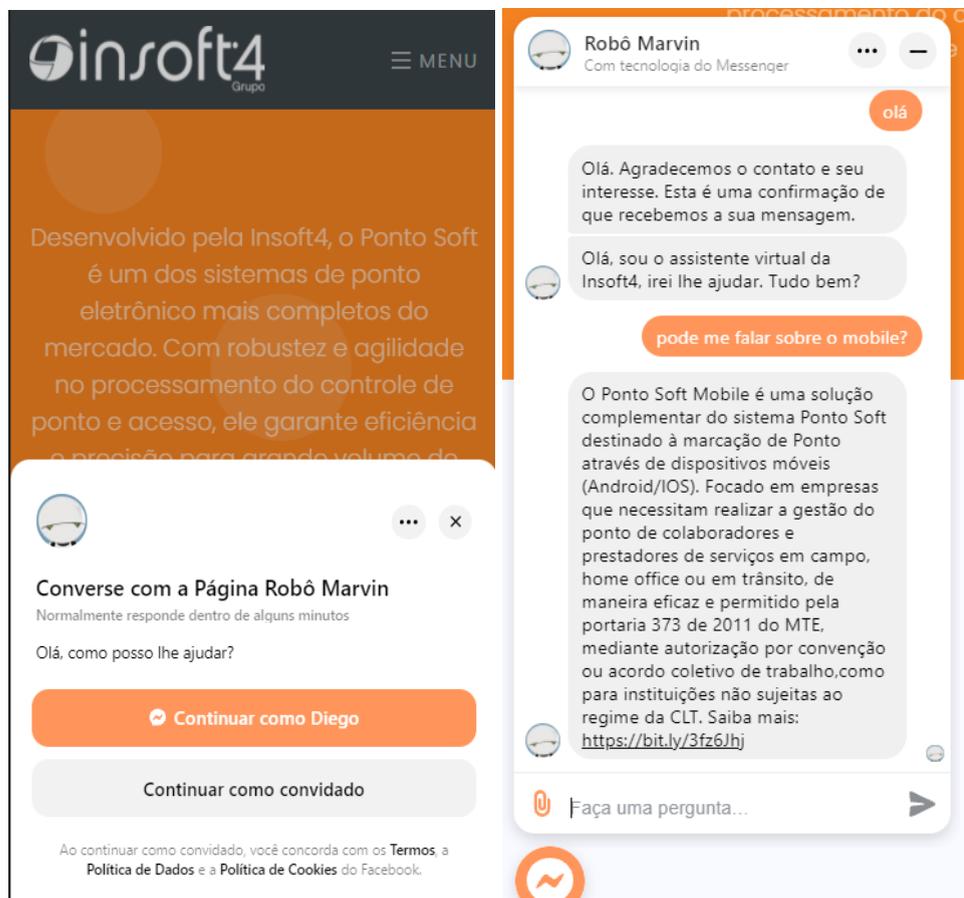
O contexto da plataforma *Facebook for Developer* possibilita a criação de um aplicativo. Este aplicativo tem o contexto de integração com IA e nele são setadas as configurações esperadas que são as mensagens, do inglês *messages*, e os retornos de mensagens, no inglês *messages postbacks*, como pode ser analisado na Figura 17 (p. 24), onde podemos ver também a identidade da aplicação (Id) do aplicativo.

Figura 17 - Aplicação para Facebook



Fonte: Elaborado pelo autor (2021)

E por fim a interface de comunicação com o usuário se dá por meio dos microsserviços, para obter a resposta da entrada fornecida pelo usuário. Neste exemplo a intenção *#ponto_soft_mobile*, da Figura 18, é apresentado como resposta para interação do usuário.

Figura 18 - Segundo Escopo *Website* (Seção Home)

Fonte: Elaborado pelo autor (2021).

6 RESULTADOS

O portal *web* atualmente está online e conta com versões de treinamento para cada plataforma, WATSON e RASA, juntamente com as integrações das API, Facebook, Calendly, Microsoft Teams e Pagadoria. Para acompanhar as interações da plataforma, foram feitos ensaios em modo de depuração comparando as interações realizadas a partir do portal *web* com Watson e com RASA. Em um período de sete dias de análise de logs, as coletas ocorrem em diferentes momentos de diferentes interações com o portal *web*.

No período de estudo ocorreram 12 solicitações de conversas. No que se refere a métrica de quantidade total de solicitações distintas, cada conversa concluiu-se com uma média de 31,8 mensagens com os agentes conversacionais. Estas conversas foram realizadas por 5 usuários diferentes, sendo que cada usuário solicitou uma nova pergunta mais de uma vez com os *chatbots*, totalizando 93 mensagens trocadas.

O algoritmo de processamento de linguagem natural fornecido pelo IBM Watson Assistant obteve uma performance de 9 respostas assertivas²² sobre as mensagens trocadas, tendo-se 6 mensagens sem entendimento, o que corresponde a 18.75%. Destas 31 mensagens, 4 foram classificadas como contexto de digressão e 2 em média necessitam de confirmação.

O algoritmo de processamento de linguagem natural RASA obteve uma performance de 6 respostas assertivas sobre as mensagens trocadas, tendo-se 7 mensagens sem entendimento, o que corresponde a 21.875%. Destas 31 mensagens, 5 foram classificadas como sem resposta relevante.

Finalmente, realizou-se análise do agente conversacional mediante a consultas realizadas, onde foi possível comparar os logs e tentativas de interação com IBM Watson Assistant e RASA, o que se mostrou viável para aplicação e estruturação através do conhecimento adquirido. Tendo o fato de coexistirem várias possibilidades de assuntos distintos que o usuário poderia utilizar para iniciar a interação com os *chatbots*, mesmo com uma base de dados relativamente pequena e com as funcionalidades de detecção de intenção e sugestões através de bases de contexto, os agentes conversacionais conseguiram realizar os atendimentos, o que certamente iria baixar a taxa de solicitação de atendimento humano, resultando assim em otimização de tempo nos atendimentos. As ferramentas também

²² Assertiva, no sentido de que a ação pode ser interpretada através dos modelos dados e treinamento atuais do *chatbot*.

permitem um ambiente de melhoria contínua, pois quanto mais são utilizadas novas informações são obtidas, podendo-se realizar assim um refinamento nos modelos e assim aprimorar a precisão, bem como definir metas de avaliação para o atendimento aos usuários.

7 CONCLUSÃO

O desenvolvimento deste trabalho compreendeu as etapas de aquisição do conhecimento, em que foram obtidos os conhecimentos necessários para criação dos fluxos conversacionais para o desenvolvimento dos protótipos de agente conversacionais. Compreendeu desde a criação dos fluxos de conversa, empregabilidade e comparação da infraestrutura e implementação do processamento de linguagem natural utilizando as soluções IBM Watson Assistant e RASA, como também a utilização dos microsserviços responsáveis pela comunicação e interpretação dos agentes conversacionais, assim como a concepção do portal *web* utilizada pelos microsserviços.

Como projetos futuros, fica a proposta de integração de novas funcionalidades, como retornos referentes a promoções, solicitações de informação de pagadoria e automação do serviço de acompanhamento para o portal *web* como Google Analytics. A sistematização dos processos e melhoria das interações entre o *chatbot* exige uma curadoria contínua, sendo melhorada à medida que novos modelos são treinados, podendo assim criar novas ações para plataformas, e assim alcançar novos resultados. Fica então a sugestão a criação de um projeto de curadoria das plataformas de agentes conversacionais.

REFERÊNCIAS

BOOTSTRAP. **Portal do ecossistema Bootstrap**. Disponível em: <http://getbootstrap.com>. Acesso em: ago. de 2021.

CARDOSO, Rodrigo “Pokemaobr”. **Conceitos que você deve conhecer para criar chatbots utilizando ferramentas online**. 2017. Disponível em: <https://imasters.com.br/desenvolvimento/conceitos-que-voce-deve-conhecer-para-criar-chatbots-utilizando-ferramentas-online>. Acesso em: 02 ago. 2021.

CALENDLY. Disponível em: <https://calendly.com/pt>. Acesso em: 10 jun. 2021.

CASAS BAHIA. Disponível em: <https://www.casasbahia.com.br/?s=bahianinho>. Acesso em: 05 set. 2021.

DANTAS, Rodrigo. **Webhook**: o que é, como funcionam e implementam webhooks? 2021. Disponível em: <https://blog.vindi.com.br/o-que-sao-webhooks/>. Acesso em: 17 set. 2021.

DIAS, Maria Abadia Lacerda; MALHEIROS, Marcelo de Gomensoro. **Estudos de técnicas de radicalização para a Língua Portuguesa**. 2005. Disponível em: https://www.researchgate.net/profile/Marcelo-Malheiros/publication/242193490_Estudo_de_tecnicas_de_radicalizacao_para_a_Lingua_Portuguesa/links/58e2b51392851c1b9d6a055a/Estudo-de-tecnicas-de-radicalizacao-para-a-Lingua-Portuguesa.pdf?origin=publication_detail. Acesso em: 12 maio 2021.

FACEBOOK. Três maneiras pelas quais as mensagens estão transformando o caminho até a compra. 2018. Disponível em: <https://www.facebook.com/business/insights/series/messaging-moves-business>. Acesso em: 25 fev. 2021.

FERREIRA, Juliana Costa *et al.* Using Chatbot as an after-sales service strategy in personal insurance. **Revista Gestão & Tecnologia**, [S.l.], v. 21, n. 2, p. 211-238, abr./jun. 2021.

FRANCONI, Enrico. **Description logics for natural language processing**. In: NATIONAL CONFERENCE ON ARTIFICIAL INTELLIGENCE, 12., 1994, Seattle. Proceedings [...]. Menlo Park, California: AAAI Press, 1994. Disponível em: <https://www.aaai.org/Papers/Symposia/Fall/1994/FS-94-04/FS94-04-005.pdf>. Acesso em: 26 out. 2021.

GONZALES, Marco. **Recuperação de Informação e Processamento da Linguagem Natural**. Porto Alegre. 2003.

IBM. **Plataform Watson**. Disponível em: <https://us-south.assistant.watson.cloud.ibm.com/>. Acesso em: 22 ago. 2021.

JEANLUCA *et al.* Jeanluca Fernandes Pereira. **Processamento de Linguagem Natural Aplicada na Construção de uma Agente Conversacional**. Criciúma. 2020.

LAPPIS. **Projeto Tais**. Disponível em: <https://lappis-unb.github.io/tais/documentacao/arquitetura/>. Acesso em: 05 jun. 2021.

MAGALHÃES, Paulo. **Teste de personalidade MBTI**: você conhece essa metodologia? 2018. Disponível em: <https://blog.trello.com/br/teste-de-personalidade-mbti>. Acesso em: 16 ago. 2021.

MAGAZINE LUIZA. Portal da Lu. Disponível em: <https://www.magazineluiza.com.br/portaldalu/>. Acesso em: 06 ago. 2021b.

MARCONI, Marina de Andrade; LAKATOS, Eva Maria. **Fundamentos da metodologia científica**. 5. ed. São Paulo: Atlas, 2003.

MAUJOR. **Conceitos de desenvolvimento web**. Disponível em: <https://www.maujor.com/>. Acesso em: 16 ago. 2021

MAZON, Stéfany. **Seja bem-vindo ao Watson Assistant!** 2018. Disponível em: <https://developer.ibm.com/br/articles/bem-vindo-ao-watson-assistant/>. Acesso em: 25 set. 2021.

MESSENGER. Disponível em: <https://www.messenger.com/>. Acesso em: 20 set. 2021.

MICROSOFT TEAMS. Disponível em: <https://www.microsoft.com/pt-br/microsoft-teams/>. Acesso em: 12 ago. 2021.

NORONHA, Odilio. Disponível em: <https://medium.com/rapaduratech/rasa-2-0-criando-um-chatbot-com-uma-poderosa-ferramenta-open-source-289430afcfb5>. Acessado em 20 set. 2021

NGROK. Disponível em: <https://ngrok.com/>, Acesso em: 5 jun. 2021.

PAHUJA, S. **What is Scrumban**. 2020. Disponível em: <https://www.agilealliance.org/what-is-scrumban>. Acesso em: 15 mar. 2021.

PRESSMAN, Roger S. **Engenharia de Software: Uma abordagem Profissional**. 7. ed. Porto Alegre: AMGH, 2011.

QI, Haode *et al.* **Benchmarking Commercial Intent Detection Services with Practice-Driven Evaluations**. 2021. Disponível em: <https://arxiv.org/abs/2012.03929>. Acesso em: 07 ago. 2021.

QUINLAN, J, Ross. **C4.5: Programs for machine learning**. São Francisco: Morgan Kaufmann Publishers Inc., 1993.

RASA, **The Stories concept**. Disponível em: <https://rasa.com/docs/rasa/stories/>. Acesso em: 10 ago. de 2021.

SILVA, Diogo Vinícius de S.; SANTOS, F. Alan de O.; SANTOS NETO, Pedro. Os benefícios do uso de Kanban na gerência de projetos e manutenção de software. *In*: SIMPÓSIO BRASILEIRO DE SISTEMAS DE INFORMAÇÃO (SBSI), 8. , 2012, São Paulo. **Anais [...]**. Porto Alegre: Sociedade Brasileira de Computação, 2012 . p. 715-725.

SILVA, Douglas da. **Fluxograma chatbot: aumentando a eficiência do atendimento ao cliente**. 2021. Disponível em: <https://www.zendesk.com.br/blog/fluxograma-chatbot/>. Acesso em: 20 set. 2021.

TRELLO. Disponível em: <https://trello.com/>. Acesso em: 07 ago. 2021.

UBER. **Quem somos**. Disponível em: <https://www.uber.com/br/pt-br/about/>. Acesso em: 07 set. 2021.

VISUAL STUDIO CODE. Disponível em: <https://code.visualstudio.com/>. Acesso em: 15 jun. 2021.

WHATSAPP. Disponível em: <https://www.whatsapp.com/>. Acesso em: 05 de ago. 2021.