



FACULDADE DE TECNOLOGIA, CIÊNCIAS E EDUCAÇÃO

Graduação

GRADUAÇÃO EM CIÊNCIA DA COMPUTAÇÃO

Sistema de resposta automatizado *ChatBot* por WhatsApp

Lucas Osse
Lucas Custódio Recco (Orientador)

RESUMO

Neste artigo desenvolvemos um sistema de automação que possibilita um processo de participação promocional completo que inclui processamento de arquivos enviados e uma experiência de conversação com *ChatBot* através da plataforma *WhatsApp*. A popularidade desse aplicativo tomou a atenção de comerciantes, especialmente de pequenos negócios, oferecendo uma plataforma de comunicação fácil e barata compatível com suporte ao consumidor. Seu design voltado para mensagens pessoais significa que não é compatível com a escala de projetos grandes e de alto volume de participação, algo que pode ser resolvido com automação como um *ChatBot*, comumente utilizado no *Facebook*. Entretanto, apesar de muito popular, ele não disponibiliza uma API aberta para desenvolvedores, mas sim somente para testes internos em números pré-aprovados. Uma alternativa oficial, denominada *WhatsApp for Business*, lida somente com mensagens de texto, ou respostas de ausência, ainda necessitando de um usuário humano.

Palavras-chave: ChatBot. WhatsApp. Respostas Automatizadas. Conversação.

ABSTRACT

In this article we'll be speaking about an automation system that enables a complete promotional participation process that includes uploaded file processing and a chatbot conversation experience through the WhatsApp platform. The popularity of this application took the attention of entrepreneurs, especially those of small businesses, offering an easy and cheap communication platform compatible with customer support. But its personal message driven design means it is not compatible with large scale promotions of high participation

volume, something that can be solved with automation such as a chatbot, commonly used in Facebook. However, despite being very popular, it does not have an open API available to developers, only one for pre-approved internal testing. An official alternative, named WhatsApp for Business, deals entirely on text messaging or away replies, still requiring a human user.

Keywords: ChatBo. WhatsApp. Automated Responses. Conversation.

Introdução

O WhatsApp é uma plataforma que cresceu rapidamente desde sua inepção, com o propósito de facilitar e baratear custos de mensagens por celulares utilizando internet móvel. Se tornou um dos aplicativos mais populares principalmente nas regiões do Oriente Médio, no Brasil e Índia (STATISTA, 2018). O aplicativo foi logo considerado para usos comerciais como em marketing e interação com clientes.

Entretanto, como seu propósito principal são mensagens particulares, há limitações severas especialmente de escala. Respostas automáticas poderiam ser uma solução para lidar com uma quantia grande de clientela, e apesar de experimentos com software não oficiais serem conduzidos, atualizações no serviço do app os inviabilizaram.

O aplicativo é popular com pequenos empreendedores como forma de comunicação eficiente, negociações podem ser realizadas inteiramente por mensagens particulares dentro dele. Porém isso não seria viável em promoções de alta escala que podem chegar a dezenas de cadastros por segundo e tem disponibilidade 24 horas por dia.

Para resolver esse problema existem soluções de automação, o modelo escolhido para este artigo é o *Chatbot*, que é um software no qual os usuários podem interagir com uma conversa. Plataformas como o *Facebook Messenger* contém *chatbots* oficiais com APIs para facilitar no seu desenvolvimento.

Uma API (do Inglês *Application Programming interface*) é um conjunto de instruções ou funções com o propósito de criar uma ponte entre o software que está sendo desenvolvido e o software alvo a ser manipulado. Em certos casos, como no WhatsApp, há dados e informações que são disponibilizadas somente via APIs, com aprovação do desenvolvedor.

Uma plataforma como essa está disponível no Facebook desde o dia 18 de Julho, porém está em fase de testes, pendentes de aprovação. Infelizmente não foi possível obter acesso a esse método, a empresa na qual trabalho não foi aprovada para o seu uso, sendo necessário caminhos diferenciados para se obter informações e interações cruciais para o funcionamento do ChatBot.

ChatBot é um programa de computador que possibilita usuários a conversarem com ele através das mesmas plataformas que uma típica conversa seria realizada com outras pessoas, como pelo WhatsApp ou outros aplicativos e programas de mensagens. Apesar de ser um tema muito discutido atualmente, existem programas similares há anos, o mais antigo sendo ELIZA de 1966. (WIZENBAUM, 1966, p. 36)

O potencial de chatbots em diminuir custos e ganhos em eficiência é alto, um software desse tipo pode carregar várias funções que um usuário normalmente encontraria num aplicativo ou website, como registro de cadastros, pagamentos, requisição de serviços etc. mas através de uma conversa, que é uma interface mais amigável. Tudo sem necessitar visitar um website ou baixar um novo aplicativo.

ChatBots também fazem uso de Inteligência Artificial, que pode ser restritiva a roteiros a conversações mais naturais. As versões mais avançadas de tais software incluem linguagem natural, e podem até se passar como pessoas reais. Um chatbot que consegue convencer uma pessoa que ele é um ser humano passa o teste de Turing (TURING, 1950, p. 2).

Esse projeto será um programa do Windows com janela, que vai usar de automação de browser, foi encontrado uma API que automatiza funções de browser chamada Selenium Web Browser. Seu propósito original é automação de testes com usuários, simulando um usuário real performando ações no navegador mas em alta velocidade. O funcionamento geral do programa será como uma pessoa real clicando e respondendo mensagens com digitação no WhatsApp Web, mas completamente automático.

A API do Selenium suporta vários navegadores. Foi escolhido o Chrome para esse propósito por ser o navegador com maior suporte e mais funcionalidades no driver. Para a integração com os websites da promoção, serão feitas chamadas REST. Essas chamadas enviarão informações para uma

webpage específica que irá lidar com o cadastramento do usuário em banco de dados.

O Bot em si irá gravar no banco de dados as mensagens que os usuários enviaram e seus números de remetente, também como Logs caso haja erros. Sua Inteligência Artificial será baseada em um roteiro rígido, que aguarda um dado ou resposta específica do usuário e trata outros tipos de envios com uma mensagem de erro.

O controle do fluxo de conversação será gravado em uma tabela específica que guarda valores binários como marcadores que indicam quais passos da conversa foram completados. Esse fluxo se inicia com o usuário enviando informações relevantes a promoção, ou sendo direcionado a fazer isso caso haja envio incorreto.

Um fluxo típico de promoção consiste em:

1. Usuário envia conteúdo
2. Bot pede informações e mídia
3. Usuário responde conforme o roteiro
4. É gravado no banco de dados as informações
5. Bot avisa usuário que seu cadastro está completo

Tendo acesso aos balões de mensagem dos usuários, é possível identificar e fazer um controle dos envios de usuários. Analisando o HTML é possível extrair todo o texto ou identifica conteúdo de mídia como fotos, vídeos, áudios e documentos. Com esse controle, adicionado ao sistema de marcadores é possível manter um fluxo consistente com o usuário e criar uma conversação linear.

Justificativa

A inserção no mercado dos aplicativos é alta. Os quatro maiores aplicativos mensageiros hoje são WhatsApp, Facebook Messenger, WeChat e Line. O programa desenvolvido nesse artigo foi criado conforme nos foi solicitado pela empresa Cover Tecnologia, pois há uma demanda alta de mercado para integrar Chatbots no WhatsApp, que é um App muito popular no Brasil.

A empresa, já tem em seu portfólio um chatbot para Messenger e agora expandiu para o WhatsApp. Aplicativos como o Messenger possuem suporte oficial para a criação de chatbots, incluindo uma API para facilitar o seu desenvolvimento.

Há mais de 300 mil chatbots no Facebook Messenger, destinados às mais variadas funções, como apoio a obtenção de um Green Card para residência nos Estados Unidos como o Visabot (FACEBOOK, 2018). Em comum entre a maioria está o apoio automático ao cliente para vários processos administrativos, burocráticos e repetitivos.

O fluxo desses bots consistem em adicioná-los na sua lista de contatos e selecionar as funções desejadas para interação. O Facebook Messenger é uma plataforma popular, que com apoio oficial obteve mais de 300 mil chatbots (NEALON, 2018). O WhatsApp é uma plataforma similar, mas que não mantém apoio oficial a bots, porém há alta demanda para um sistema de chatbot similar.

Apesar disso, há exemplos de software automatizados no WhatsApp ainda atualmente ativos, um deles é um chatbot pertencente ao Exército Brasileiro (Exército Brasileiro, 2014). Outro exemplo no setor privado é o bot do Submarino, que te envia periodicamente atualização sobre os produtos adquiridos pelo site.



Figura 2: bot do exército brasileiro

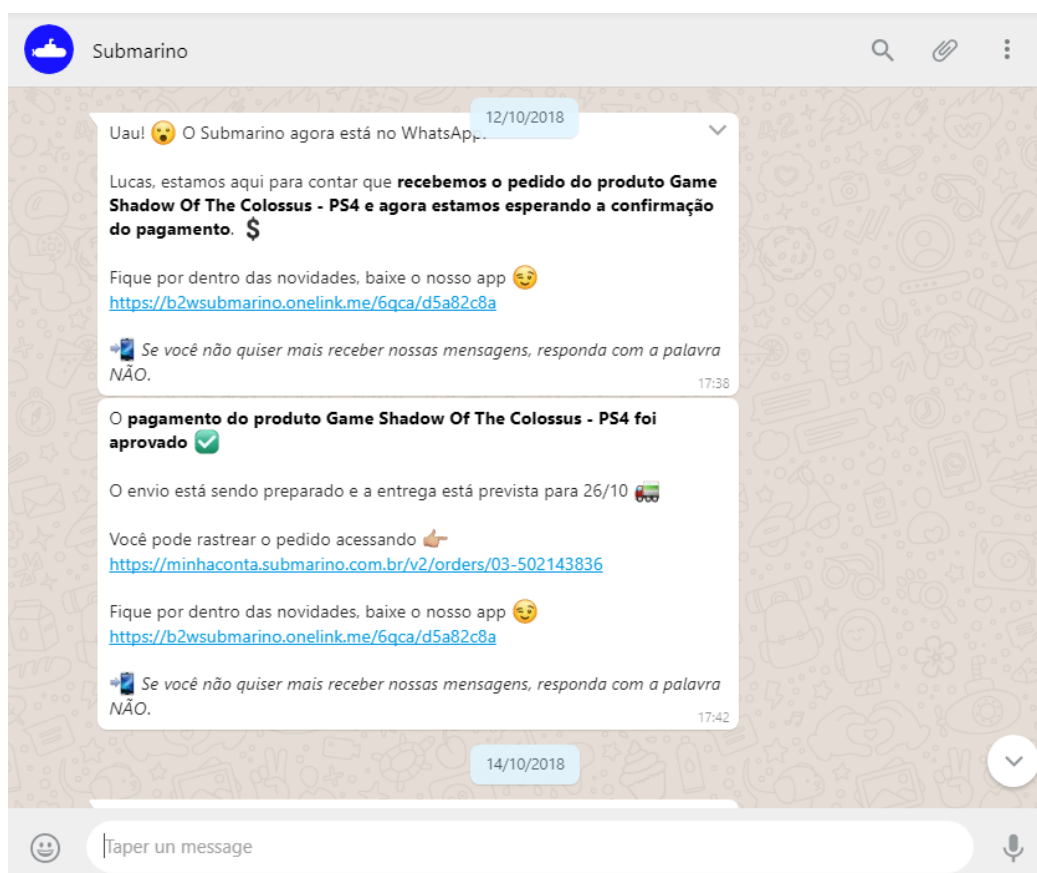


Figura 3: bot do Submarino

Objetivos Gerais

O Objetivo Geral é estabelecer um sistema que processa texto e mídia enviados por usuários, com um tratamento de respostas dependendo do que foi enviado. Um sistema que completa um cadastro comumente visto em websites promocionais.

O software também irá conversar com o usuário, o texto que lhe foi enviado iniciará uma cadeia de eventos em roteiro que envolve requisição de mídia específica relacionada a promoção na qual está associado. Essa conversação irá conter pedidos de envio, e um tratamento de erro caso haja envio de mídia incorreto.

Processamento de mídia seguirá os critérios estabelecidos pela promoção. Usualmente elas necessitam de uma imagem do cupom fiscal, que pode ser baixada no servidor e enviado ao banco de dados. Dito isso, ele pode ser configurado para obter outros tipos de mídia, ou ignorar todos os envios e somente realizar pedidos de informação por texto.

Objetivos Especificos

Com a necessidade dos bots definida, é necessário verificar os requisitos específicos da promoção que irá aderir a ele. Tais requisitos variam entre os tipos de mídia que devem ser tratados e ignorados, ao roteiro de conversação como também todo o fluxo da promoção. Usaremos um modelo médio das promoções como exemplo de uma possibilidade.

Sem API oficial, é necessário realizar engenharia reversa no whatsapp, utilizando-se de sua ferramenta WhatsApp Web. Deve-se encontrar uma maneira de manipulá-lo para receber interações do usuário e enviar respostas.

A engenharia reversa envolverá coletar informações de certas definições do HTML da página, como as tags ou as classes dessas tags.

Definido essas informações, deve ser criado o projeto do programa. Foi definido um software de Windows escrito na linguagem de programação C#, seguindo as diretrizes da empresa com o auxílio do programa Visual Studio, na licença comercial da Cover Tecnologia.

Com o design estabelecido formalmente, é necessário criar um controle operacional que mantém o fluxo das conversações e um registro de logs e conversações. Para esses propósitos o ideal é um banco de dados que pode ser modelado de uma maneira padrão que encaixe em todas as promoções.

Referencial teórico

Devido a alta demanda de chatbots na atualidade, a maior concentração de literatura sobre o assunto é recente. Entretanto, a primeira concepção de um software dessa natureza vem de 1966, chamado ELIZA. Entre as referências temos vários artigos sobre chatbots e inteligência artificial.

Graças ao alto volume de informação recente sobre os chatbots, por causa de sua alta demanda no mercado contemporâneo, há muita informação também descrita em websites. Muitas empresas também disponibilizam PDFs demonstrando a necessidade de integração de um bot no portfólio empresarial.

A intenção de um chatbot é uma conversação similar a uma com um ser humano real, no mercado há demanda para tal software realizar serviços ao consumidor que são burocráticos e repetitivos para seres humanos. Porém, há

bots não comerciais cujo propósito é passar o teste de Turing. Um desses se chama Cleverbot (CARPENTER, 2018).

O Cleverbot é um software que se utiliza de inteligência artificial para aprender com todas as conversas que ele teve, um caso de aprendizado de máquina (NASRABADI, 2007, p. 7). De acordo com o seu design quanto mais conversas ele completa com usuários pela internet, mais convincente ele fica.



Figura 4: Cleverbot

Desenvolvimento

As três ferramentas centrais para a criação desse software foram: A API de automação chamada Selenium, o Visual Studio, e o software SQL Server da microsoft. Graças ao Visual Studio e ao SQL Server, foi possível criar um projeto

de software com janela relacionado a um banco de dados com facilidade. E Graças a API Selenium, toda a interação com o WhatsApp Web, a versão de aplicativo que roda no navegador, foi realizada.

O aplicativo foi desenvolvido na linguagem de programação C#, em conjunto com o Visual Studio. Esta linguagem foi escolhida pois é uma das suportadas e documentadas pelo Selenium. Com o auxílio do Visual Studio, a aplicação foi desenvolvido em base de Windows Forms, com uma janela que disponibiliza informações para monitoração de back-end.

O Selenium WebDriver é uma ferramenta com o propósito de realizar testes automatizados. Ele gera uma instância isolada do navegador de sua escolha, incluindo *Google Chrome*, *Firefox*, *Safari* entre outros. Como no início do projeto não havia uma API viável para interagir com o WhatsApp, o sistema de interação será uma simulação de um usuário clicando no chat e digitando a resposta.

Apesar da necessidade do software em realizar as mesmas etapas que um usuário, devido a sua natureza automática ele é capaz de finalizar essas tarefas em uma velocidade muito maior, com grandes mensagens complexas sendo processadas em menos de um segundo.

Selenium funciona através de seletores, que encontram elementos por suas características do código da página web. Para encontrar os campos relevantes a serem manipulados, foi preciso investigar a página do whatsapp web com a ferramenta de Inspeccionar Elementos, nativa de muitos navegadores. Nesse desenvolvimento foi utilizado o Google Chrome em todos os testes.


Após coletar informações do funcionamento do WhatsApp Web, foi criado um projeto no Visual Studio, uma aplicação com janela. Essa janela apresenta feedback ao usuário, como informações de debugging e o status atual do programa. Cada conta do WhatsApp requer um número de telefone, a janela apresenta o número no qual a conta está logada e mensagens de erro.

A API de Selenium em conjunto com C#, providenciam uma Lista, e um tipo de variável no qual pode se armazenar vários elementos que contém um tipo de característica em comum, como uma classe ou tag.

O controle de mensagens foi feito com o auxílio de um banco de dados. Uma tabela Denominada “tbMensagem” registra tudo que o usuário enviou para

o nosso contato e a data em que foi enviada. Diferentes tags são extraídas dependendo do formato e conteúdo do envio, como imagem, documentos, etc.

Tabela I: Design da tabela tbMensagem

	Column Name	Data Type	Allow Nulls
	Id	int	<input type="checkbox"/>
	dtMensagem	datetime	<input checked="" type="checkbox"/>
	stRemetente	varchar(100)	<input checked="" type="checkbox"/>
	stMensagem	varchar(1000)	<input checked="" type="checkbox"/>
	blFoto	bit	<input checked="" type="checkbox"/>
	tipoMsgs	char(1)	<input checked="" type="checkbox"/>
	stBot	varchar(100)	<input checked="" type="checkbox"/>
			<input type="checkbox"/>

Id: Chave primária da tabela, serve como identidade para tornar cada linha única.

dtMensagem: A data do envio da mensagem do usuário, essa informação é extraída do HTML da página.

stRemetente: O número do usuário que enviou as mensagens.

stMensagem: O conteúdo da mensagem em texto. Se conteúdo diverso como imagens, esse campo informa através de um texto padrão.

blFoto: Se a mensagem desse registro é uma imagem ou não.

tipoMsgs: Controla através de um caractere o tipo de envio.

stBot: O número usado internamente pelas promoções. Foi necessário adicionar esse campo para filtrar por celular as diversas promoções e participações.

1.1 Visual Studio

Devido a parceria oficial da Cover Tecnologia com a Microsoft, foi dada preferência a ferramentas e métodos apoiados pela corporação. Usando a linguagem de programação desenvolvida pela Microsoft chamada C#, em conjunto com o Visual Studio, providenciou um ambiente de desenvolvimento otimizado.

1.2 Selenium Webdriver

A API principal que lida com toda a interação com o website do WhatsApp. Foi necessário uma análise de todo o HTML contido na página web.whatsapp.com. Toda a seleção de elementos da API é feita de maneira hierárquica, com os primeiros elementos que aparecem no HTML, lendo de cima para baixo, e contém o seletor dado sendo escolhidos.

Selenium também pode selecionar uma lista de elementos, contendo todos os itens do HTML que contém o seletor dado. Além de poder selecionar os elementos e gravar seu conteúdo em variáveis, Selenium também disponibiliza várias ações que imitam um típico usuário. Há como clicar, recarregar a página, e enviar teclas do teclado direcionado no elemento selecionado entre outras. (Selenium, 2018)

1.3 Banco de Dados no SQL Server

A empresa na qual trabalho providenciou uma versão licenciada do software SQL Server, da Microsoft. Há uma forte integração com bancos de dados no programa, que fazem controles de conversação, logs de erros e registros de mensagens ambos dos usuários e da promoção.

Todas as mensagens enviadas por usuários são registradas no banco de dados. Dessa maneira é possível acessar as mensagens enviadas anteriormente pelo usuário de maneira mais rápida. Versões mais antigas do software necessitavam da mensagem anterior para determinar a posição da conversa, isso foi otimizado com outro método.

As Tabelas do Banco de Dados final:

tbClienteStatus: Guarda a posição das conversas com o bot.

tbMensagem: Guarda as mensagens enviadas ao bot.

tbSysBotNumeros: Guarda os números de telefone associados a contas que tem o bot.

tbSysLog: Log de erros.

tbSysParam: Parâmetros que contém as mensagens padrão. Elas podem ser alteradas pelo banco sem fechar os bots.

1.4 Celulares

Cada promoção disponibilizou 2 números de telefone para registrar contas. O processo de criar uma conta foi o padrão que todos os usuários fazem. Um chip é atrelado a conta do whatsapp como login e posteriormente usado para logar na interface web. Uma limitação imposta pelo aplicativo é ser necessário que ambos o celular e computador estejam conectados na internet para a interface funcionar.

Os dois números da promoção ficam ligados 24 horas por dia, todos logados no WhatsApp mantendo a conexão com o cliente web viva. Em caso de desconexão com a internet, o software encerra um ciclo de passagem das mensagem e recarrega até a internet voltar.

1.5 Tratamento de Mídia

O WhatsApp permite o envio de texto, fotos, vídeo, áudio e documentos como PDFs. Selenium nos permite coletar o HTML ou partes de um elemento, como seus atributos e classes, e o guardar em uma variável como “string”. String é um formato padrão que representa uma cadeia de caracteres em C#. Esse formato é versátil e pode ser interagido de várias maneiras na linguagem de programação.

O software de chatbot analisa o texto que foi enviado e responde de acordo. A recepção esperada varia de acordo com os critérios da promoção. Quando é necessário guardar a mídia, é reproduzido através do Selenium os mesmos passos que um usuário faria ao salvar o conteúdo.

Em uma imagem, por exemplo, selenium clica nas coordenadas em que está localizada, e com ela em visualização clica no botão de Download. É necessário clicar para visualizar a imagem pois o HTML que contém o botão de download não é gerado até esse evento. Em outras mídias onde o botão já está na página não é necessário reproduzir o conteúdo antes de baixar.

1.6 Integração com serviços externos

O tratamento de imagens envolve uma chamada externa para uma página de serviço da promoção. O propósito dessa chamada é gravar no banco de dados oficial as informações e mídia que foram coletadas com a interação do usuário. Essa chamada ocorre toda vez que o usuário finaliza o cadastro, com todos os dados, gravados temporariamente nas tabelas internas, sendo enviados juntos. A chamada é do tipo POST, que envia todas as informações via um formulário.

1.7 Inteligência

Apesar de lidar com situações imprevistas, ele tem um roteiro fixo e padronizado. Esse roteiro guia o usuário a fazer as ações necessárias para o cadastro na promoção. Se o usuário divergir do roteiro, há um aviso para retomar o curso correto do processo de cadastramento.

O roteiro da maioria das promoções aderidas envolve uma foto. Essa foto representa o cupom fiscal do usuário que está participando da promoção e pode ou não ser o primeiro envio do usuário. Após esse primeiro envio, é coletado a mensagem e esta é gravada no banco de dados interno, em seguida é gravado um novo registro daquele usuário na tabela “tbClienteFluxo” que indica a posição da conversa e o chatbot responde o usuário com instruções para o próximo envio.

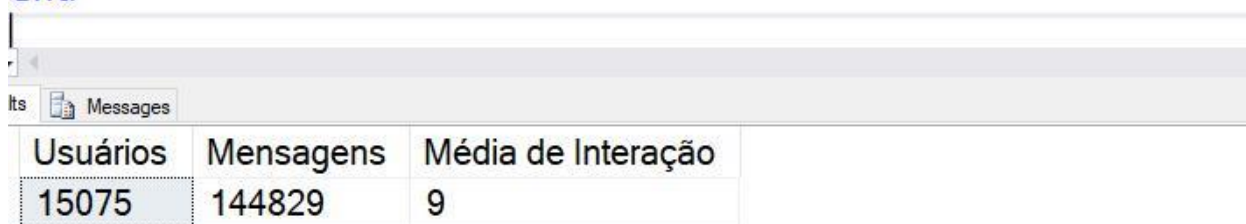
O Fluxo pode se repetir assim indefinidamente, com o registro na “tbClienteFluxo” sendo atualizado até o usuário chegar no ponto final da conversa. Ao atingir a última etapa da conversa, as informações são enviadas para o banco de dados oficial da promoção através de uma chamada POST e o usuário pode reiniciar com uma nova participação, ou ser advertido que atingiu o limite de participação.

Resultados

O software cumpre todas as funções necessárias para ser considerado um “chatbot”, o usuário interage com ele enviando texto ou mídia e é respondido organicamente. O bot é resiliente e capaz de persistir quedas de internet e lidar com aproximadamente 250 mensagens por dia. Foi aceito em várias promoções

oficiais nas quais ele se integra e envia informações para seus banco de dados e atingiu 8905 pessoas que enviaram 89973 mensagens em média de 10 mensagens por pessoa e opera 24 horas por dia sem interrupções.

```
BEGIN
select
count(distinct stRemetente) as 'Usuários'
, count(stMensagem) as 'Mensagens'
, count(stMensagem)/(select count(distinct stRemetente))
as 'Média de Interação'
from tbMensagem
end
```



Usuários	Mensagens	Média de Interação
15075	144829	9

Figura 5: Registros resgatados no banco de dados no dia 10 de Dezembro de 2018.

Considerações Finais

Em conclusão, é possível desenvolver um sistema de resposta automática e conversação para o aplicativo WhatsApp, mesmo sem uma API oficial disponibilizada. A demanda para automação através do WhatsApp é alta e já é utilizada por várias empresas como Submarino. O software resultante do desenvolvimento utiliza uma API de automação de testes para simular a interação de um usuário real mas em alta velocidade. Há diferentes versões do programa em produção, cada uma com roteiro e tratamento de mídia particulares.

Projetos Futuros

O software opera atualmente 24 horas por dia em um servidor, respondendo vários usuários por dia. Entretanto, há possibilidade de várias

melhorias no seu funcionamento. A conversação atual com o programa segue um roteiro rígido que o usuário deve seguir ou ser advertido, deixando claro sua natureza artificial. É possível ambos localmente e através de serviços online melhorar a interação com o usuário de uma maneira mais orgânica com inteligência artificial aprimorada. Há programas com linguagem avançadas o suficiente para enganar o usuário e fazê-lo pensar que é um ser humano, portanto passando o teste de Turing.

A característica mais versátil do software é sua habilidade de realizar chamadas externas. Há uma grande quantidade de APIs externas ambas pagas ou grátis com uma variedade de serviços. Todos podem ser integrados com o chatbot. Por exemplo, um usuário pode enviar sua localização para o software, e após uma chamada de API, o chatbot pode retornar a previsão do tempo daquele local.

A API oficial existente será no futuro disponível para todas as empresas após o período de testes (Facebook, 2018), a vantagem de seu uso é o suporte oficial do Facebook, como um desenvolvimento mais simples sem necessidade de operações complexas como engenharia reversa. A API também pode ser executada sem interface visual dentro de um container *Docker* (Facebook, 2018), o que significa que consome muito menos recursos e pode rodar em ambos Windows e Linux.

Agradecimentos

A esta universidade, seu corpo docente, direção e administração que oportunizaram a janela que hoje vislumbro um horizonte superior, eivado pela acendrada confiança no mérito e ética aqui presentes. Ao meu orientador Lucas Custódio Recco, pelo suporte no pouco tempo que lhe coube, pelas suas correções e incentivos. Aos meus pais, pelo amor, incentivo e apoio incondicional. E a todos que direta ou indiretamente fizeram parte da minha formação, o meu muito obrigado.

Referências

ACCENTURE. **Accenture Chatbots in Customer Service**. 2016. Disponível em: <https://www.accenture.com/t00010101T000000__w__/br-pt/_acnmedia/PDF-45/Accenture-Chatbots-Customer-Service.pdf>. Acesso em: 01 de Nov. de 2018.

AUTOMATE the Planet. **Getting Started with WebDriver C# in 10 Minutes**. 2015. Disponível em: <<https://www.automatetheplanet.com/getting-started-webdriver/>>. Acesso em: 1 nov. 2018.

CARPENTER, Rollo. **Cleverbot**, 2018. Disponível em: <<https://www.cleverbot.com>>. Acesso em: 01 de Nov. de 2018.

DALE, R. The return of the chatbots. **Natural Language Engineering**, v. 22, n. 5, p. 811-817, 2016.

DEVELOPERS. **Introducing Bots on Messenger**. 2016. Disponível em: <<https://developers.facebook.com/videos/f8-2016/introducing-bots-on-messenger/>>. Acesso em: 1 nov. 2018.

_____. **WhatsApp Business API**. 2018. Disponível em: <<https://developers.facebook.com/docs/whatsapp/>>. Acesso em: 1 nov. 2018.

_____. **Availability and Scaling**. 2018 <<https://developers.facebook.com/docs/whatsapp/availability-scaling>>. Acesso em: 1 nov. 2018.

DYNAMSOFT. **Dynamsoft**, 2018. Disponível em: <<https://www.dynamsoft.com/>>. Acesso em: 1 nov. 2018.

EXÉRCITO Brasileiro. **Twitter**, 2014. Disponível em: <<https://twitter.com/exercitooficial/status/504634226671910912/photo/1>>. Acesso em: 1 nov. 2018.

GITHUB. Simple Yet Hackable WhatsApp api. 2018 .Disponível em: <<https://github.com/VISWESWARAN1998/Simple-Yet-Hackable-WhatsApp-api>>. Acesso em: 1 nov. 2018.

ISKANDAR, J. I. Normas da ABNT: comentadas para trabalhos científicos. 3. ed. Jurua Editora, 2009.

NASRABADI, N. M. Pattern recognition and machine learning. **Journal of electronic imaging**, v. 16, n. 4, 2007.

NEALON, G. Using Facebook Messenger And Chatbots To Grow Your Audience. **Forbes**, 2018. Disponível em: <<https://www.forbes.com/sites/forbesagencycouncil/2018/06/04/using-facebook-messenger-and-chatbots-to-grow-your-audience/#64edb64b733b>>. Acesso em: 1 nov. 2018.

RADZIWILL, N. M.; BENTON, M. C. **Evaluating quality of chatbots and intelligent conversational agents**. 2017.

SAENZ, A. Cleverbot Chat Engine Is Learning From The Internet To Talk Like A Human. **Singularity Hub**, 2010. Disponível em: <<https://singularityhub.com/2010/01/13/cleverbot-chat-engine-is-learning-from-the-internet-to-talk-like-a-human/>>. Acesso em: 1 nov. 2018.

SELENIUM. **Introducing the Selenium-WebDriver API by Example**. 2018. Disponível em: <https://www.seleniumhq.org/docs/03_webdriver.jsp#introducing-the-selenium-webdriver-api-by-example>. Acesso em: 1 nov. 2018.

SIMILARWEB. **Whatsapp.com Analytics**, 2018. Disponível em: <<https://www.similarweb.com/website/whatsapp.com>>. Acesso em: 1 nov. 2018.

STATISTA. **Whatsapp**: mobile usage penetration by country. Disponível em: <<https://www.statista.com/statistics/291540/mobile-internet-user-whatsapp/>>. Acesso em: 1 nov. de 2018.

TURING, A. M. **Computing Machinery and Intelligence**. 1950.

VISABOT. **Facebook**, 2016. Disponível em: <<https://www.facebook.com/visabot/>>. Acesso em: 1 nov. 2018.

WEIZENBAUM, J. Eliza: a computer program for the study of natural language communication between man and machine. **Communications of the ACM**, v. 9, n. 1, p. 36-45, 1966.

WEIZENBAUM, J. Computer power and human reason: from judgment to calculation. 1976.