



CURSO DE SISTEMAS DE INFORMAÇÃO

**DESENVOLVIMENTO DE UM CHATBOT PARA AUXILIO NOS
TREINOS DE ALUNOS EM ACADEMIAS DE MUSCULAÇÃO**

JOÁS CASTRO COSTA

Palmas

2019



CURSO DE SISTEMAS DE INFORMAÇÃO

DESENVOLVIMENTO DE UM CHATBOT PARA AUXILIO NOS TREINOS DE ALUNOS EM ACADEMIAS DE MUSCULAÇÃO

JOÁS CASTRO COSTA

Trabalho apresentado ao Curso de Sistemas de Informação da Fundação Universidade do Tocantins - UNITINS como parte dos requisitos para a obtenção do grau de Bacharel em Sistemas de Informação, sob a orientação do professor Me. Silvano Malfatti.

Palmas

2019



CURSO DE SISTEMAS DE INFORMAÇÃO

DESENVOLVIMENTO DE UM CHATBOT PARA AUXILIO NOS TREINOS DE ALUNOS EM ACADEMIAS DE MUSCULAÇÃO

JOÁS CASTRO COSTA

Trabalho apresentado ao Curso de Sistemas de Informação da Fundação Universidade do Tocantins - UNITINS como parte dos requisitos para a obtenção do grau de Bacharel em Sistemas de Informação, sob a orientação do professor Me. Silvano Malfatti.

Prof. Me. Silvano Malfatti
Orientador

Prof. Me. Janio Elias Teixeira Junior

**Prof. Me. Marco Antonio Firmino de
Sousa**

Palmas
2019

Dados Internacionais de Catalogação na Publicação (CIP)
Biblioteca da Universidade Estadual do Tocantins
Câmpus Graciosa – Palmas-TO

C837d Costa, Joás Castro.
Desenvolvimento de um chatbot para auxílio nos treinos de alunos em academias de musculação / Joás Castro Costa. – Palmas, TO, 2019.
46f. il.col.
Inclui CD-ROM.

Monografia (TCC) – Universidade Estadual do Tocantins, Câmpus Graciosa, Curso de Sistemas de Informação, 2019.
Orientador (a): Silvano Malfatti.

1. Sistemas de informação. 2. Tecnologia da informação. 3. Inteligência artificial. 4. Chatbot. I. Título.

CDD 006.3

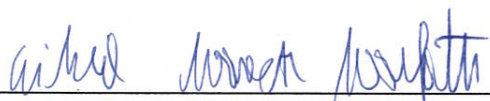
Bibliotecária: Cacilda Martins Madureira CRB-2 / 0561

Todos os Direitos Reservados – A reprodução parcial, de qualquer forma ou por qualquer meio deste documento é autorizado desde que citada à fonte. A violação dos direitos do autor (Lei nº 9.610/98) é crime estabelecido pelo artigo 184 do código penal.

**UNIVERSIDADE ESTADUAL DO TOCANTINS
CURSO DE SISTEMAS DE INFORMAÇÃO**

ATA DE DEFESA DE TRABALHO DE CONCLUSÃO DE CURSO

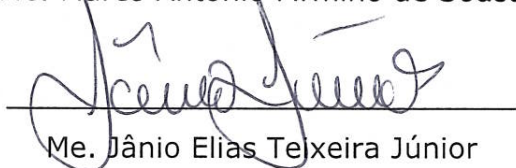
Aos vinte e oito dias do mês de junho de 2019, reuniu-se a banca examinadora do trabalho apresentado como Trabalho de Conclusão de Curso em **Sistemas de Informação** de **JOÁS CASTRO COSTA**, intitulada: **DESENVOLVIMENTO DE UM CHATBOT PARA AUXÍLIO NO TREINO DE ALUNOS EM ACADEMIAS DE MUSCULAÇÃO**. Compuseram a banca examinadora os professores **Me. Silvano Maneck Malfatti (Orientador)**, **Me. Marco Antônio Firmino de Sousa** e **Me. Jânio Elias Teixeira Júnior**. Após a exposição oral, o (a) candidato(a) foi arguido(a) pelos componentes da banca que reuniram-se reservadamente, e decidiram, APROVAR, com a nota 7,5. Para constar, redigi a presente Ata, que aprovada por todos os presentes, vai assinada por mim, Coordenadora de TCC do curso de Sistemas da Informação, e pelos demais membros da banca.




Me. Silvano Maneck Malfatti



Me. Marco Antônio Firmino de Sousa



Me. Jânio Elias Teixeira Júnior



Joás Castro Costa

Este trabalho é dedicado à minha família, pelo apoio incondicional.

Agradecimentos

Agradeço primeiramente a Deus pela oportunidade impar de concluir um curso de graduação em uma instituição de ensino pública e gratuita. Aos meus pais que sempre me ajudaram e prestaram apoio incondicional em todos os momentos e que foram essenciais na formação do meu caráter e motivação para que não desistisse nos momentos difíceis. Aos meus amigos de jornada, pelo companheirismo e ajuda nos momentos difíceis que passamos ao longo desses anos de curso. Aos amigos de fora da faculdade que de alguma forma contribuíram para que eu pudesse estar vivendo esse momento. É impossível dimensionar o imenso prazer que estamos sentindo nesse momento de alegria, mas se fosse resumir o meu sentimento em uma palavra depois desses longos anos de dedicação a minha área de formação, a palavra seria GRATIDÃO.

*“Não vos amoldeis às estruturas deste mundo,
mas transformai-vos pela renovação da mente,
a fim de distinguir qual é a vontade de Deus:
o que é bom, o que Lhe é agradável, o que é perfeito.
(Bíblia Sagrada, Romanos 12, 2)*

Resumo

Com o avanço da tecnologia da informação, as empresas estão cada vez mais aderindo as novas tecnologias com o intuito de modernizar o seu negócio e oferecer diversos serviços aos seus clientes. O presente trabalho apresenta o desenvolvimento de um chatbot com o objetivo de auxiliar alunos nos seus treinos e demais atividades em academias de musculação. Considerando a demanda crescente de software nas academias de musculação e devido a quantidade de pessoas que estão aderindo o hábito de se exercitar, esse software vai auxiliar o aluno no seu desenvolvimento bem como possibilitar que o seu personal trainer ou instrutor da academia possa acompanhar uma quantidade maior de alunos do que se fosse feito presencialmente. O principal resultado obtido a partir deste trabalho foi o desenvolvimento de um chatbot que auxilia alunos de academias de musculação no seus treinos diários.

Palavras-chaves: Tecnologia da Informação. Chatbot. Inteligencia Artificial.

Abstract

With the advancement of information technology, companies are increasingly adhering to new technologies in order to modernize their business and offer various services to their customers. The present work presents the development of a chatbot with the purpose of assisting students in their training and other activities in bodybuilding academies. Considering the increasing demand for software in bodybuilding academies and due to the number of people who are getting into the habit of exercising, this software will help the student in his development as well as enable his personal trainer or instructor of the academy to accompany to quantity higher than if it were done in person. The main result obtained from this work was the development of a chatbot that assists students of bodybuilding academies in their daily workouts.

Key-words:Information Technology. Chatbot. Artificial Intelligence.

Lista de ilustrações

Figura 1 – Arquitetura Geral do Watson Assistant	21
Figura 2 – Ciclo de Vida da Atividade	24
Figura 3 – Arquitetura Geral do Trabalho	28
Figura 4 – Fluxo de Dialogo	31
Figura 5 – Gavetas de Navegação - Navigation Drawer	32
Figura 6 – Telas do Aplicativo	33
Figura 7 – Bibliotecas - Watson Assistant	33
Figura 8 – Método de Autenticação com Watson Assistant	34
Figura 9 – Exemplo de Conteúdo - QRCODE	34
Figura 10 – Diagrama de Atividade - Fluxo Resumido	35
Figura 11 – Diagrama de Atividade - Fluxo Completo	36
Figura 12 – Webservice Rest	37
Figura 13 – Método Repository - getExercicioPordia	37
Figura 14 – Método Service - listarExercicio	38
Figura 15 – Mensagem Inicial <i>Chatbot</i>	39
Figura 16 – Reconhecimento da Intenção de Saudação	40
Figura 17 – Reconhecimento da Intenção - Falar com instrutor	41
Figura 18 – Reconhecimento da Intenção - Treino	41
Figura 19 – Reconhecimento Intenção treino e leitura do QRCODE	42
Figura 20 – Reconhecimento da Intenção Quem é você	43
Figura 21 – Reconhecimento da Intenção - Desrespeito	43
Figura 22 – Intenção Não Reconhecida	44

Lista de abreviaturas e siglas

TI - Tecnologia da Informação.

API - Application Programming Interface.

IA - Inteligencia Artificial.

IDE - Integrated Development Environment.

UML - Unified Modeling Language.

HTTP - HyperText Transfer Protocol

UDDI - Universal Description Discovery and Integration

WSDL- Web Service Description Language

XML - Extensible Markup Language

JSON - JavaScript Object Notation

SOAP - Simple Object Access Protocol

OEM - Original Equipment Manufacturer

Sumário

1	INTRODUÇÃO	14
1.1	Objetivos	15
1.2	Objetivos Gerais	15
1.3	Objetivos Específicos	15
1.4	Justificativa	15
2	REFERENCIAL TEÓRICO	17
2.1	Chatbot	17
2.2	APIs de desenvolvimento de <i>Chatbot</i>	18
2.2.1	Watson Assistant - IBM	18
2.2.2	Microsoft Bot Framework - Microsoft	18
2.2.3	Dialogflow api.ai - Google	18
2.2.4	Amazon Lex - Amazon	18
2.3	Engine Chatbot: IBM Watson Assistant	19
2.3.1	Arquitetura Watson Assistant	20
2.4	Inteligencia Artificial	21
2.5	Aplicativos para Dispositivos Móveis Android	22
2.5.1	Ciclo de vida de uma Activity	23
2.6	Webservice: integração e interoperabilidade:	25
3	METODOLOGIA	26
3.1	Métodos e Técnicas de pesquisa	26
3.2	Procedimentos Metodológicos	27
3.3	Materiais e Softwares	27
3.4	Arquitetura do trabalho	28
4	DESENVOLVIMENTO	30
4.1	Implementação no Watson Assistant	30
4.1.1	Definição das intenções	30
4.1.2	Fluxo de Dialogo	31
4.2	Aplicativo Android	32
4.2.1	Integração com o Watson Assistant	33
4.2.2	Interação via QRCODE	33
4.2.3	Implementação <i>Chatbot</i>	34
4.3	Webservice	36

5	RESULTADOS	39
5.1	Reconhecimento de Intenções	39
5.1.1	Reconhecimento da Intenção de Saudação	40
5.1.2	Reconhecimento da Intenção de Falar com instrutor	40
5.1.3	Reconhecimento da Intenção - Treino	41
5.1.4	Reconhecimento da Intenção - Quem é você?	42
5.1.5	Reconhecimento da Intenção - Desrespeito	42
5.1.6	Não Reconhecimento de Intenção	43
6	CONSIDERAÇÕES FINAIS	45
6.1	Trabalhos futuros	46
	REFERÊNCIAS	47

1 Introdução

O avanço da tecnologia da informação tem proporcionado o surgimento de sistemas de informação cada vez mais inteligentes (BRITO; OLIVEIRA, 2017). A inteligência artificial acompanhou esse avanço e tem contribuído para o desenvolvimento de sistemas inteligentes capazes de simular a capacidade humana e solucionar problemas ou reproduzir o pensamento do ser humano (FILHO, 2009).

Desde que Alan Turing (1950) propôs que o computador poderia se comportar de maneira inteligente, através do teste de turing, a tecnologia vem avançando gradativamente no sentido de tornar a comunicação humano-maquina mais próxima a linguagem natural falada pelos seres humanos (MANFIO, 2015).

Manfio (2015) afirma que "É inquestionável, por exemplo, a praticidade gerada a usuários de nível básico por um sistema por comandos de voz que permite realizar operações sem ajuda de terceiros."

Nesse contexto, o *chatbot* surgiu como uma proposta inovadora considerando seus aspectos de disponibilidade e capacidade de interagir com uma pessoa de forma autônoma e intuitiva. Levando em consideração a sua capacidade de interagir com uma pessoa de forma natural e a satisfação das necessidades pontuais das pessoas de forma interativa e objetiva (PEREIRA; PINHEIRO, 2018), o *chatbot* tem se demonstrado uma ferramenta capaz de atender a necessidade de software em diversos ramos da sociedade.

Rodizio Bento et al. (2013) corrobora no sentido que a TI por meio dos seus diversos tipos de sistemas de informação, contribui de maneira significativa em agilizar a rotina de trabalho de estabelecimentos de vários segmentos em especial as academias de musculação.

Neste sentido, este trabalho propõe o desenvolvimento de um *chatbot* utilizando a API IBM Watson Assistant para auxílio nos treinos de alunos em academias de musculação. Esse *chatbot* vai auxiliar o aluno no momento do seu treinamento, bem como proporcionar ao seu instrutor uma alternativa de supervisionar uma quantidade maior de alunos, em relação ao acompanhamento presencial.

1.1 Objetivos

1.2 Objetivos Gerais

- Desenvolver um *chatbot* para auxílio no treino de alunos em academias de musculação.

1.3 Objetivos Específicos

- Realizar um estudo bibliográfico sobre *Chatbot*
- Fazer um estudo bibliográfico da aplicação da inteligência artificial em *chatbots*
- Definição das intenções básicas do *chatbot*
- Desenvolver um aplicativo para dispositivos moveis Android que será a interface de interação com o *chatbot*.
- Integrar o *chatbot* com uma aplicação WEB para teste das funcionalidades.
- Construir um Webservice para o compartilhamento de dados entre a aplicação Web e o *chatbot*.

1.4 Justificativa

É indiscutível que a pratica de atividades físicas têm se mostrado uma ferramenta de suma importância na busca por uma saúde física e bem-estar ([Rede Movimentar, 2016](#)). Tornou-se forte aliada na prevenção da saúde física, bem como o bem estar das pessoas, entre outros fatores. Sendo assim, a musculação vem conquistando um espaço crescente, por propiciar resultados estéticos, melhorar a performance desportiva e colaborar na prevenção, restauração e reabilitação das mais diversas condições patológicas ([Rede Movimentar, 2016](#)).

A quantidade de pessoas que iniciam atividades físicas em academias de musculação vem crescendo a cada dia. A busca pela melhoria da saúde, o desejo de alcançar os padrões de beleza imposto pela sociedade e o fortalecimento muscular são os principais motivos para adesão dessas pessoas às academias de musculação ([LIZ; ANDRADE, 2013](#)).

Para que essas pessoas tenham sucesso nos seus objetivos, é necessário um acompanhamento adequado por parte dos profissionais da área de educação física. No entanto, verifica-se que pela quantidade de alunos por profissional em uma academia, esse acompanhamento fica comprometido influenciando muitas vezes no resultado do aluno ou levando o aluno a desistência.

Neste sentido, a solução a ser desenvolvida por este trabalho tem a proposta de expandir o acompanhamento do aluno pelo instrutor e ao mesmo tempo proporcionar ao aluno uma ferramenta de acompanhamento do seu treino diário através da interação com o *chatbot*, dentre outras funcionalidades que serão fundamentais no desenvolvimento do aluno e no feedback para o instrutor.

2 Referencial Teórico

2.1 Chatbot

O termo *chatbot* originou-se do inglês onde *chat* significa conversador e *bot* é uma abreviação da palavra *robot* que significa robô. O *chatbot* é um programa de computador que deve ser capaz de simular a conversa de um ser humano (TEIXEIRA; MENEZES, 2003). Ele deve agir de tal forma que a pessoa que está interagindo com o *chatbot* tenha a impressão de estar conversando com uma pessoa e não um programa de computador. O primeiro programa de computador considerado *chatbot* foi desenvolvido em 1966 pelo professor Joseph Weizenbaum do Instituto de Tecnologia de Massachusetts e foi chamado de ELIZA (WEIZENBAUM, 1966). O objetivo do ELIZA é simular a conversação entre um psicólogo e um paciente.

A principal característica de um *chatbot* é sua capacidade de interagir com um usuário como se fosse um ser humano. Isso tem sido possível graças ao emprego das técnicas da inteligência artificial. Com base nas técnicas de inteligência artificial empregadas, Rothmel e Domingues (2007) identifica três gerações de *chatbots*, a saber:

- A primeira geração, é baseada no casamento de padrões e regras gramaticais e tem como maior representante o ELIZA.
- A segunda geração, baseada nas técnicas de redes neurais e regras de produção, tem como representante o JULIA, desenvolvido por Michael Mauldin.
- A terceira e mais recente geração é baseada na utilização de linguagens de marcação na construção de bases de conhecimento. Seu principal representante é o ALICE

Com a evolução da tecnologia e principalmente da IA foram desenvolvidos diversos *chatbots* dotados de inteligência para atender a demanda de vários segmentos. Galvão (2003) considera que os *chatbots* são utilizados atualmente como facilitadores principalmente nos seguintes seguimentos: entretenimento, ensino a distância, comércio eletrônico e atendimento ao consumidor.

2.2 APIs de desenvolvimento de *Chatbot*

Para facilitar desenvolvimento de *chatbots* algumas empresas disponibilizam APIs que são plataformas onde uma pessoa pode construir um *chatbot* com conhecimento técnico de programação ou não. A seguir, será apresentado alguns exemplos dessas plataformas.

2.2.1 Watson Assistant - IBM

O Watson Assistant é uma API de desenvolvimento de *chatbots* criado pela empresa IBM. A empresa define que "O Watson Assistant é uma oferta para criar interfaces de conversação em qualquer aplicativo, dispositivo ou canal¹". Enfatiza ainda o fato de que não é necessário ser um desenvolvedor para usar a sua plataforma devido ao seu editor de diálogo visual. Essa plataforma será apresentada de forma mais detalhada posteriormente

2.2.2 Microsoft Bot Framework - Microsoft

A Microsoft Bot Framework é uma plataforma de criação de *chatbots* da empresa Microsoft. A empresa define que "É uma estrutura abrangente para criar experiências de conversação em AI de nível corporativo²". A plataforma utiliza os serviços cognitivos denominado "Azure Bot Service" que segundo a empresa: "fornece um ambiente integrado desenvolvido especificamente para o desenvolvimento de *bots*, permitindo que você crie, conecte, teste, implante e gerencie bots inteligentes, tudo em um só lugar.³"

Empresas como Laliga e Adobe atualmente são clientes e possui *chatbots* construído por meio desse serviço.

2.2.3 Dialogflow api.ai - Google

Segundo Brandes (2017), "O Dialogflow é uma plataforma para construir interfaces de conversação para bots, aplicativos e dispositivos." Em setembro de 2016 a Google adquiriu essa plataforma, o que pode evidenciar o poder da plataforma e o seu potencial de evolução (BRANDES, 2017). Por meio do site oficial a empresa informa que "O Dialogflow incorpora a experiência e os produtos de aprendizado de máquina do Google, como o Google Cloud Speech-to-Text.⁴."

2.2.4 Amazon Lex - Amazon

Segundo Amazon (2019): "O Amazon Lex é um serviço para a criação de interfaces de conversa em qualquer aplicativo usando voz e texto"

¹ <https://www.ibm.com/cloud/watson-assistant/>

² <https://dev.botframework.com/>

³ <https://docs.microsoft.com/en-us/azure/bot-service/?view=azure-bot-service-4.0>

⁴ <https://dialogflow.com/>

Amazon (2019) informa também que:

"O Amazon Lex disponibiliza funcionalidades avançadas de aprendizado profundo de Automatic Speech Recognition (ASR – Reconhecimento automático de fala), para a conversão de fala em texto, e Natural Language Understanding (NLU – Compreensão de linguagem natural), para o reconhecimento da intenção do texto, o que permite criar aplicativos com experiências do usuário altamente envolventes e interações conversacionais realistas."

A plataforma apresenta no seu site oficial alguns casos de uso de *chatbots* que a Amazon Lex pode ser aplicada: bots de call center, de informações, aplicações, de produtividade corporativa e bots conectados aos segmentos da Internet das Coisas.

2.3 Engine Chatbot: IBM Watson Assistant

Entre os componentes de um *chatbot* o mais importante é o que desempenha a função de *engine* do sistema. A engine é o componente responsável por realizar a conversão da linguagem natural em uma ação inteligível por máquinas. O desenvolvimento de mecanismos que funcionam como a engine do chatbot é complexo e utiliza diversos modelos de processamento de linguagem natural e aprendizado de máquina para fornecer níveis consideráveis de precisão. (KAR; HALDAR, 2016).

Com o objetivo de facilitar o desenvolvimento de *chatbots* algumas empresas já disponibilizam o recurso de processamento de linguagem natural no modelo de Software como Serviço SaaS (Software as a service) e é aplicado em *chatbots* por meio de API's (KAR; HALDAR, 2016).

O Watson Assistant associa aprendizado de máquina com interpretação de linguagem natural e ferramentas de diálogos integradas para extrair intenções e entidades de um diálogo com a finalidade de estabelecer um fluxo de conversação entre os aplicativos e seus usuários¹. Para entender o funcionamento do Watson Assistant é necessário apresentar os principais conceitos contido nessa engine.

a) Intenções:

Intenções são os objetivos que se espera dos usuários quando interagirem com o *chatbot*. Para cada objetivo que possa ser identificado na entrada de um usuário é necessário definir uma intenção². Por exemplo, em um chatbot de pedidos de pizza pode ser definida

¹ IBM. *Assistente Watson v1 - Documentos da API da nuvem IBM*. Disponível em: <<https://console.bluemix.net/apidocs/assistant?language=node#introduction>>. Acesso em: 20 nov. 2018.

² IBM. *IBM CLOUD DOCS / IBM ASSISTANT*. Disponível em: <<https://console.bluemix.net/docs/services/assistant/index.html#sobre>>. Acesso em: 20 nov. 2018.

a intenção "pedirPizza" para identificar a intenção do usuário de realizar um pedido.

b) Entidades:

Entidades são informações necessárias para compreender o contexto e retirar a duplicidade em uma intenção. Elas facilitam a identificação de parâmetros necessários para executar uma ação específica (KAR; HALDAR, 2016). É possível definir entidades comuns que podem ser usadas em cenários diferentes. Por exemplo: Data, Numero, Dinheiro. Bem como entidades específicas de uma intenção. Por exemplo, em um *chatbot* para *help desk* se a intenção for consertar equipamento pode-se definir entidades para distinguir o tipo de equipamento em que o usuário deseja consertar.

c) Contexto:

O contexto representa a habilidade de manter o estado de uma conversa para identificar a intenção do usuário. Ele é usado também para lidar com situações de ambiguidade de sentidos ou quando a entrada do usuário seja imprecisa (KAR; HALDAR, 2016). Através do contexto é possível associar uma declaração do usuário aparentemente vaga em relação a uma referência anterior.

d) Dialogo:

O Dialogo representa a sequência de um fluxo de conversa que incorpora as intenções e entidades definidas dentro de um contexto. De acordo com a entrada do usuário e a identificação da intenção e entidade será estabelecido uma forma de diálogo.

2.3.1 Arquitetura Watson Assistant

A Figura 1 demonstra a arquitetura geral do Watson Assistant e o seu funcionamento. Os usuários interagem por meio dos pontos de integração. Os pontos de integração por sua vez, são plataformas de mídias sociais onde é possível publicar um *chatbot*. Por exemplo, o Messenger do Facebook ou o chat do Slack. No momento em que Watson recebe a entrada do usuário ele encaminha a uma área de trabalho, que é um contêiner para seu fluxo de diálogo e dados de treinamento³.

Dentro da área de trabalho onde contem o fluxo de dialogo, a entrada do usuário é interpretada pelo Watson que direciona para um fluxo de conversa e coleta as informações necessárias para responder ou executar uma ação para o usuário⁴.

³ IBM. *IBM CLOUD DOCS / IBM ASSISTANT*. Disponível em: <<https://console.bluemix.net/docs/services/assistant/index.html#sobre>>. Acesso em: 20 nov. 2018.

⁴ IBM. *IBM CLOUD DOCS / IBM ASSISTANT*. Disponível em: <<https://console.bluemix.net/docs/services/assistant/index.html#sobre>>. Acesso em: 20 nov. 2018.

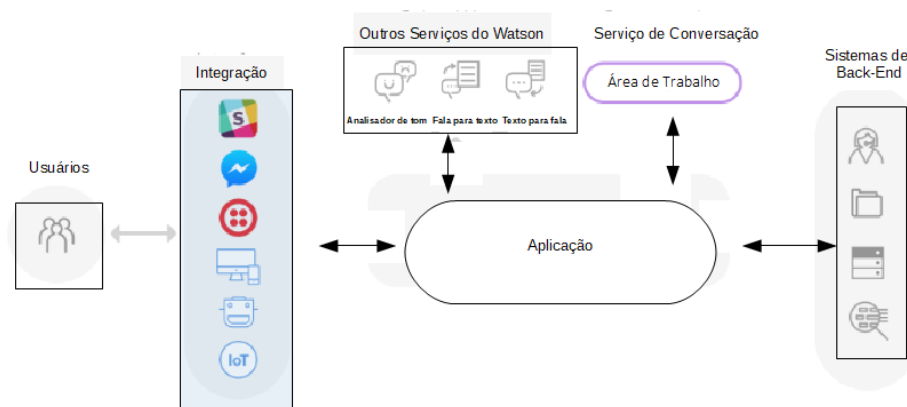


Figura 1 – Arquitetura Geral do Watson Assistant
 Fonte: <https://console.bluemix.net/docs/services/assistant/index.html>sobre

2.4 Inteligencia Artificial

Para que o *chatbot* possa se comportar de maneira inteligente ele deve ser dotado de conceitos e técnicas da inteligência artificial. Conceituar a inteligência artificial é uma tarefa difícil devido as suas características e os seus variados ramos. No entanto, os autores de grande renome na literatura assim conceituam a IA:

“É o estudo de como fazer os computadores realizarem tarefas que, no momento são feitas melhor por pessoas.” (NASCIMENTO JR.; YONEYAMA, 2004 apud RICH, 1983, p.2)

“É o campo de conhecimento onde se estudam sistemas capazes de reproduzir alguma das atividades mentais humanas.” (NASCIMENTO JR.; YONEYAMA, 2004 apud NILSSON, 1986, p.2)

A primeira ideia de um sistema inteligente foi proposto por Alan Turing(1950) através do teste de turing. O teste de turing foi projetado para identificar se o computador poderia se comportar de maneira inteligente (RUSSEL; NORVING, 2010). O Teste de turing consiste em utilizar um interrogador humano que se comunica com um voluntario humano e um computador apenas com trocas de mensagens. O teste propõe que caso o interrogador não consiga diferenciar as repostas do voluntario e do computador as suas questões formuladas, de fato o computador manifesta inteligencia (NASCIMENTO JR.; YONEYAMA, 2004).

Dentro da IA existem diversos subcampos que são empregados com o intuito

de prover habilidades inteligente aos softwares, se tratando de *chatbots*, destacam-se o aprendizado de máquina e o processamento de linguagem natural.

Segundo [ANTONIO DAS NEVES \(2018\)](#): “O aprendizado de máquina é a área da inteligência artificial responsável pelos métodos e algoritmos que possuem a capacidade de aprender com informações obtidas através de uma base de dados.”

[Witten e Frank \(2005\)](#), descreve quatro tipos diferentes de aprendizagem de máquina: aprendizado por classificação, onde o esquema de aprendizagem é apresentado com um conjunto de exemplos que se esperam que aprendam uma forma de classificar exemplos não vistos; aprendizado por associação onde deseja encontrar qualquer associação entre recursos, não apenas aqueles que possui um certo valor de classe; aprendizagem por agrupamento, no qual grupos de exemplos semelhantes de acordo com um critério estabelecido são agrupados e por fim aprendizado por meio da previsão numérica, onde o resultado a ser previsto não se trata de uma lição discreta, mas um valor numérico.

Outro subcampo da IA e amplamente utilizado em *chatbots* é o processamento de linguagem natural, pois segundo [Carrilho Junior \(2007\)](#) “é o conjunto de métodos com o objetivo de analisar textos e gerar frases escritas em um idioma humano.” O processamento de linguagem natural pode ser dividido em duas abordagens: a baseadas em texto e a baseada em dialogo ([Carrilho Junior, 2007](#)). Enquanto a abordagem baseada em texto está relacionado a busca de documentos e a compreensão de textos, a baseada em dialogo tem sido objeto de interesse nas interfaces entre máquinas e seres humanos, pois estas estão voltadas cada dia mais para uma forma de comunicação mais humana ([Carrilho Junior, 2007](#)).

2.5 Aplicativos para Dispositivos Móveis Android

Nos últimos anos a forma de acesso a internet vem sendo modificada pois os computadores *desktops* e *notebooks* foram perdendo espaços para os dispositivos moveis como *tablets* e *smartphones*. Isso se deve ao fato de que os dispositivos moveis possuem um custo mais acessível, são mais fáceis de utilizar, além de executar diversas tarefas e ser facilmente transportado ([FONSECA; ALENCAR, 2016](#)). Motivado por estas vantagens, o mercado de aplicativos para aparelhos celulares mostra-se em crescimento rápido ([SILVA; SANTOS, 2014](#)).

Segundo [Silva e Santos \(2014\)](#) "Atualmente, há diversas plataformas para aparelhos celulares, tais como: Android (Google), IOS (Apple Inc), Windows Mobile (Microsoft Corp), entre outros." Cada um desses sistemas necessita que seus aplicativos nativos sejam desenvolvidos por linguagens de programações específicas ([James White, 2013](#)).

Dentre essas plataformas destaca-se o sistema operacional Android. Segundo [An-](#)

[droid Developers \(2019\)](#) "O Android é um sistema operacional de código aberto para dispositivos móveis e um projeto de código aberto correspondente liderado pelo Google".

[Android Developers \(2019\)](#) afirma que :

"O Android é uma pilha de software de código aberto criada para uma ampla variedade de dispositivos com diferentes formatos. O principal objetivo do Android é criar uma plataforma de software aberta disponível para operadoras, OEMs e desenvolvedores para transformar as ideias inovadoras deles em realidade e apresentar um produto real e bem-sucedido que aprimora a experiência móvel dos usuários."

Aplicativos para Dispositivos moveis são desenvolvidos por meios de IDEs, que são ambientes integrais de desenvolvimento de aplicações. O Android Studio é a IDE oficial de desenvolvimento de aplicativos para sistemas operacionais Android ([Android Developers, 2019](#)).

Basicamente os aplicativos Android são constituído de *Activities* que é um componente visual do aplicativo que fornece uma tela onde os usuários poderão interagir com o aplicativo. Cada *Activity* recebe uma janela onde exibe uma interface para o usuário ([Android Developers, 2019](#)).

2.5.1 Ciclo de vida de uma Activity

Quando uma *Activity* é interrompida com o início de uma nova *Activity* ela é notificada sobre essa alteração do estado por meio de métodos de retorno do ciclo de vida da Activity.

A Figura 2 mostra como se comporta o ciclo de vida de uma *Activity* e a maneira que seus métodos estão organizados.

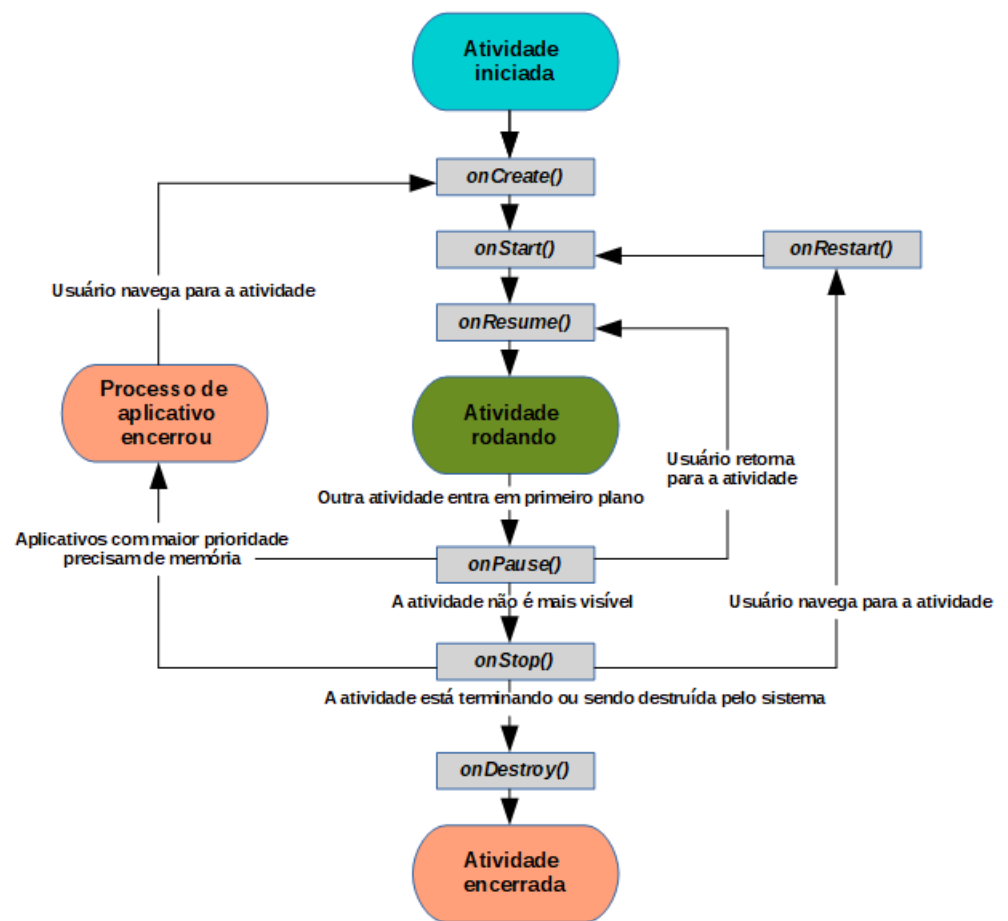


Figura 2 – Ciclo de Vida da Atividade
 Fonte: Adaptado de ([Android Developers, 2019](#))

O ciclo de vida é composto por 7 métodos que representa os estados de uma *Activity*:

- `onCreate()`: Chamado quando é iniciado a *Activity* pela primeira vez. É o método principal e deve ser obrigatoriamente criado
- `onRestart()`: Chamado sempre que uma *Activity* tiver sido interrompida
- `onStart()`: Chamado antes da *Activity* ficar visível para o usuário.
- `onResume()`: Chamado antes da *Activity* interagir com o usuário
- `onPause()`: Chamado quando o aplicativo estiver prestes a exibir outra *Activity*
- `onStop()`: Chamado quando a *Activity* não está mais visível para o usuário.
- `onDestroy()`: Chamado antes da *Activity* ser destruída. É a última chamada do aplicativo aquela *Activity*

2.6 Webservice: integração e interoperabilidade:

O conceito de *webservices* surgiu da necessidade de integrar diversos tipos de sistemas, a maioria deles operando em plataformas diferentes e desenvolvidos por diversas linguagens. Segundo [FREITAS \(2006\)](#) *Web service* é um padrão para comunicação entre diferentes softwares e aplicações com arquiteturas baseadas em serviços, que estão hospedados em diferentes plataformas. A implementação de um *web service* tem como base um conjunto de protocolos e linguagens padrões da internet, entre eles o HTTP, SOAP, WSDL e o UDDI ([FREITAS, 2006](#)).

Os *web service* são classificados em dois tipos: os baseados em SOAP e no estilo REST ([KALIN, 2013](#)). [ARMIGLIATTO \(2017\)](#), define SOAP como uma tecnologia para acessar *webservices* unicamente em forma de mensagens de requisições e respostas feitas em XML para criar uma especie de contrato entre o provedor e o consumidor do serviço através do arquivo WSDL. Os Webservices no estilo REST, segundo [LECHETA \(2015\)](#) é um formato leve de Web Service que são criados sobre o protocolo HTTP utilizando métodos GET, DELETE, POST e PUT como padronização de acesso.

A comunicação com os diferentes tipos de aplicações por meios de *webservices* é realizada utilizando formatos padrões de troca de dados. Os principais utilizados por *webservices* do tipo SOAP e Rest são o XML e JSON ([Dal Moro; DORNELES; Trindade Rebonatto, 2011](#)).

O JSON é um modelo de formato para troca de dados que possui uma sintaxe de alto nível, porém de fácil compreensão e por isso fazem do JSON um formato ideal para transmissão de dados através da rede ([OLIVEIRA, 2012](#)).

3 Metodologia

Neste capítulo será apresentada a metodologia utilizada para elaboração deste trabalho de conclusão de curso, bem como as ferramentas e tecnologias que foram utilizadas para o desenvolvimento do trabalho.

3.1 Métodos e Técnicas de pesquisa

Segundo [Marconi e Lakatos \(2010, p.65\)](#)

O método é o conjunto das atividades sistemáticas e racionais que, com maior segurança e economia, permite alcançar o objetivo - conhecimentos válidos e verdadeiros - traçando o caminho a ser seguido, detectando erros e auxiliando as decisões do cientista.

Ainda de acordo com ([CERVO; BERVIAN, 2002](#)) o método se dá com o conjunto das etapas ou passos necessários para a realização da pesquisa. A pesquisa é uma atividade direcionada para a resolução de problemas teóricos ou práticos com a aplicação de processos científicos ([CERVO; BERVIAN, 2002](#)). A pesquisa pode ser classificada do ponto de vista de sua natureza de duas formas: Pesquisa básica e Pesquisa aplicada.

A pesquisa básica consiste na geração de conhecimentos novos para o avanço da ciência sem a aplicação prática. Enquanto que, a pesquisa aplicada tem como objetivo gerar conhecimentos práticos voltados à solução de problemas específicos ([PRODANOV; FREITAS, 2013](#)). Portanto, levando em consideração esses conceitos o presente trabalho está classificado como uma pesquisa aplicada.

Em relação aos procedimentos técnicos da pesquisa este trabalho classifica-se também como uma pesquisa bibliográfica. Segundo [Cervo e Bervian \(2002\)](#) a pesquisa bibliográfica tem por objetivo conhecer e analisar as contribuições culturais ou científicas do passado em relação a determinado tema, assunto ou problema. A técnica de pesquisa bibliográfica segundo [Marconi e Lakatos \(2010\)](#), consiste em consultar as bibliografias já tornadas públicas sobre o tema a ser estudado. As bibliografias são publicações avulsas, jornais, revistas, livros, pesquisas, monografias, teses, etc.

Por meio desta técnica será possível conhecer a teoria associada aos conceitos que serão apresentados, bem como aprofundar o estudo nesses conceitos de modo a construir um pensamento voltado para a consecução dos objetivos deste trabalho.

3.2 Procedimentos Metodológicos

Este trabalho foi desenvolvido em 4 etapas. Na primeira etapa foi realizada a fundamentação teórica do trabalho, abordando os conceitos relacionados a Inteligência Artificial, *Chatbots*, Dispositivos Moveis, a API Watson Assistant e Webservices. Para esta etapa foi utilizado a técnica de pesquisa bibliográfica.

Na segunda etapa foi realizado o estudo das ferramentas e softwares necessários para o desenvolvimento do objetivo principal deste trabalho. Sendo assim, foram estudados o desenvolvimento de *chatbots* com API Watson Assistant e a sua integração com plataformas de interação, o desenvolvimento e integração de Webservices com aplicativos de dispositivos moveis e o desenvolvimento de um chatbot por meio de um aplicativo do sistema operacional Android.

Na terceira etapa foi realizada a implementação prática das aplicações utilizando todos os conhecimentos e informações adquiridas nas etapas anteriores.

Na quarta e última etapa foram realizados testes das aplicações desenvolvidas simulando interações do *chatbot* em um contexto prático e buscando solucionar os erros e otimizar a aplicação.

3.3 Materiais e Softwares

Todo ambiente de desenvolvimento foi hospedado em um Notebook pessoal da fabricante ASUS¹ MODELO X450C, processador Intel Core i5, 6GB de memória RAM, sistema operacional Windows 10 Home Single Language.

O sistema web utilizado para realizar testes com o *chatbot* foi desenvolvido utilizando a IDE Eclipse ide Photon June 2018².

A aplicação web foi desenvolvida na linguagem JAVA com a especificação JSF para a construção de interfaces de usuário baseadas em componentes para aplicações web.³

Os dados foram persistido no banco de dados através do framework de persistência de dados JPA.

O sistema de banco de dados que foi utilizado para armazenar os dados do sistema web foi o PostgreSQL 10.4⁴.

Os Diagrama de classe do Sistema Web e diagrama de atividade do *chatbot* foram gerados usando o software de modelagem de sistemas Astah UML 7.2.0.⁵

¹ <https://www.asus.com/br/>

² <https://www.eclipse.org/>

³ <https://www.oracle.com/technetwork/java/javase/javaserverfaces-139869.html>

⁴ <https://www.postgresql.org/>

⁵ <http://astah.net/editions/uml-new>

O Aplicativo para sistemas operacionais Android foi desenvolvido por meio da IDE oficial, Android Studio¹, utilizando a linguagem JAVA.

3.4 Arquitetura do trabalho

A arquitetura do trabalho demonstra todos os elementos que serão desenvolvidos e integrados para a consecução do objetivo principal deste trabalho, sendo constituída por:

- Um sistema de gerenciamento de dados com interface web de para teste de integração com o *chatbot*.
- Um Webservice para realizar a comunicação de dados entre a aplicação web, o banco de dados e o *chatbot*.
- Um Banco de dados para armazenamento dos dados
- Um aplicativo desenvolvido para o sistema operacional Android onde o *chatbot* será hospedado e por meio do qual os usuários terão acesso.
- Um recurso de *chatbot* utilizando a API IBM Watson Assistant para o reconhecimento de intenções com uso de inteligência artificial

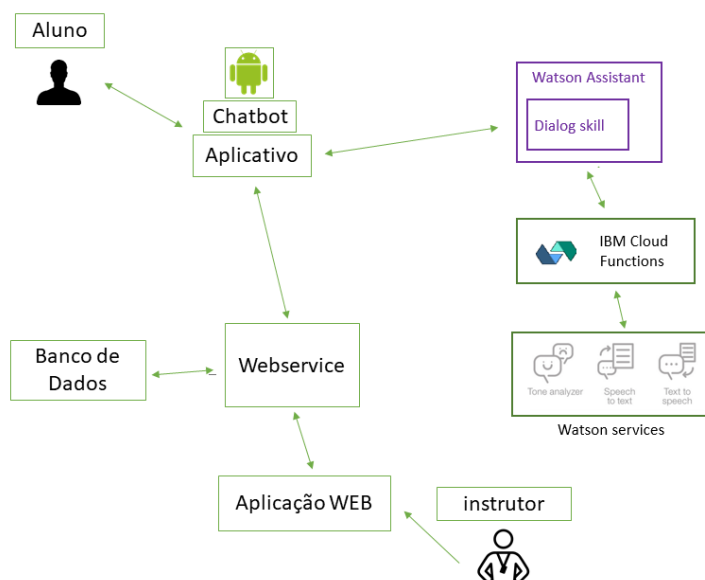


Figura 3 – Arquitetura Geral do Trabalho
Fonte: O Autor

¹ <https://developer.android.com/studio>

A figura 3 demonstra a arquitetura geral do trabalho onde o instrutor insere os dados do aluno na aplicação Web que por sua vez salva no banco de dados. No momento em que o aluno iniciar o *chatbot* no aplicativo, ele fará uma requisição ao Watson para iniciar a conversa. Quando o *chatbot* reconhecer a intenção do aluno de iniciar o treino o aplicativo faz uma consulta ao banco de dados por meio do *webservice* que retornará os dados de treino com as instruções necessárias conforme o treino definido inicialmente pelo instrutor.

4 Desenvolvimento

Neste capítulo serão demonstrados os passos utilizados para o desenvolvimento das soluções propostas

4.1 Implementação no Watson Assistant

4.1.1 Definição das intenções

Por meio da interface gráfica disponibilizada pela IBM para implementação de regras de conversação foram definidas as possíveis intenções do usuário ao interagir com o *chatbot*:

- Iniciar o dialogo / Saudação

Essa é a intenção que inicia o dialogo e saúda o usuário.

- Iniciar o Treino

Intenção utilizada para identificar o momento que o usuário vai iniciar o seu treino.

- Falar com o instrutor

Intenção ativada quando identificado o desejo do usuário de solicitar ajuda a um instrutor/personal.

- Saber quem é o robô

Intenção ativada caso o usuário tenha a curiosidade de saber com quem ele está interagindo.

- Entreter o usuário

Intenção disponível para o chatbot entreter o usuário por meio de piadas ou outros meios.

- Desrespeito

Intenção ativada quando identificado algum tipo de ofensa verbal ou palavras de baixo calão por parte do usuário.

4.1.2 Fluxo de Dialogo

Conforme as intenções anteriormente citadas foi definido o seguinte fluxo de dialogo que demonstra como o *chatbot* vai responder e se comportar quando identificado uma intenção.

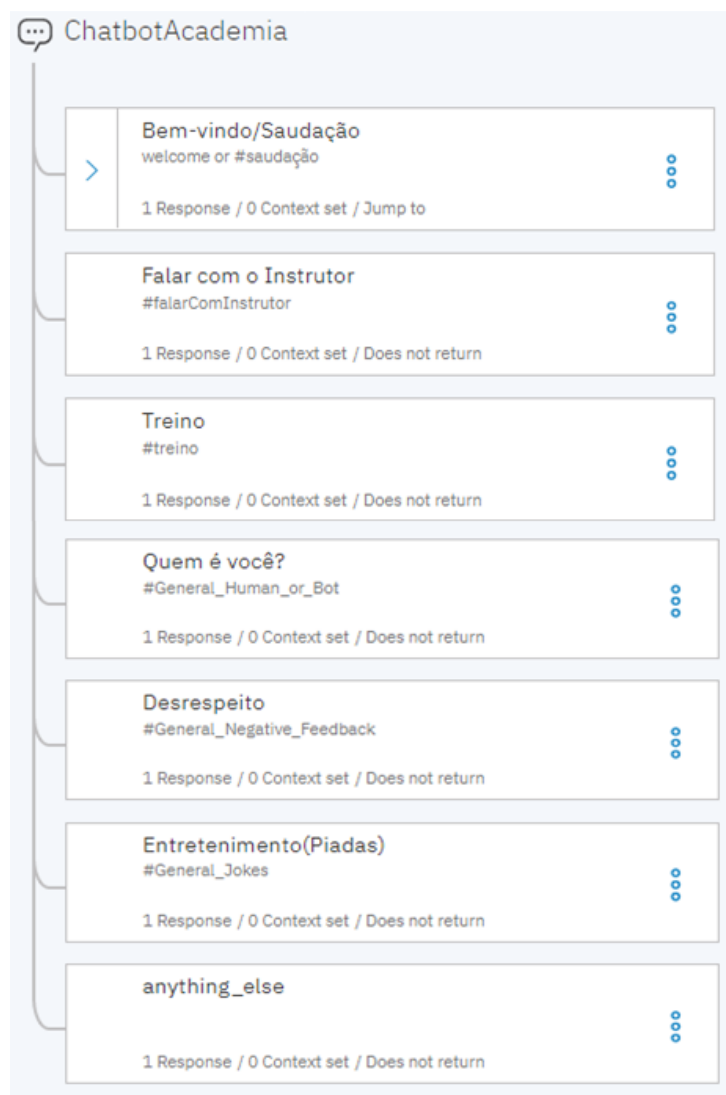


Figura 4 – Fluxo de Dialogo
Fonte: O Autor

Para cada nó do fluxo de dialogo existe uma verificação condicional que controla a entrada ou não no fluxo do respectivo nó. Dessa forma, quando o Watson reconhece uma intenção o fluxo de conversa é direcionado para o respectivo nó que representa a intenção identificada.

4.2 Aplicativo Android

O aplicativo foi desenvolvido baseado no componente visual *Navigation Drawer*¹ que utiliza o conceito de Gavetas de Navegação e Fragmentos² para exibição de conteúdos. A gaveta de navegação é um painel da interface do usuário que mostra o menu de navegação principal do aplicativo. A gaveta é mostrada sempre que o usuário toca no ícone da gaveta na barra de aplicativos ou passa um dedo pela borda esquerda da tela. Os fragmentos estão dispostos dentro dessas gavetas de acordo com a sua funcionalidade no aplicativo.

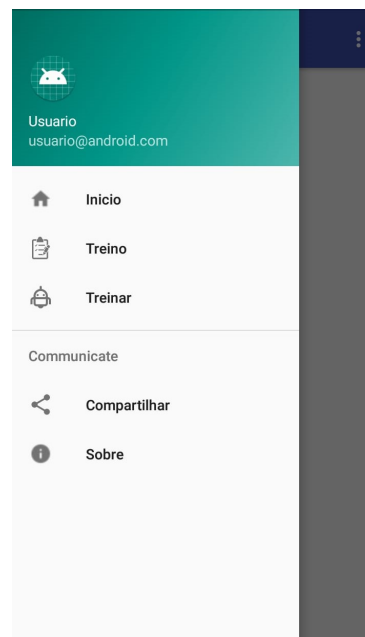


Figura 5 – Gavetas de Navegação - Navigation Drawer

Fonte: O Autor

A tela inicial do aplicativo é a tela de login. a autenticação é realizada por meio do *webservice*. Quando o usuário digitar o nome de usuário e a senha e pressionar o botão "Entrar" o aplicativo realiza uma requisição ao *webservice* buscando esses dados. Caso os dados digitados foram previamente cadastrados e associados a um usuário o *webservice* irá retornar o id, o nome e e-mail do aluno e o aplicativo exibira a tela principal do aplicativo. Além da tela de login o aplicativo possui as telas de treino, treinar e sobre. Cada tela representa os fragmentos criados. A tela treino contém a ficha de treino do usuário, a tela treinar é onde o usuário interage com o *chatbot* e a tela sobre contém as informações do desenvolvedor do aplicativo. A Figura 6 retrata cada uma dessas telas.

¹ <https://developer.android.com/guide/navigation/navigation-ui>

² <https://developer.android.com/guide/components/fragments?hl=pt-br>

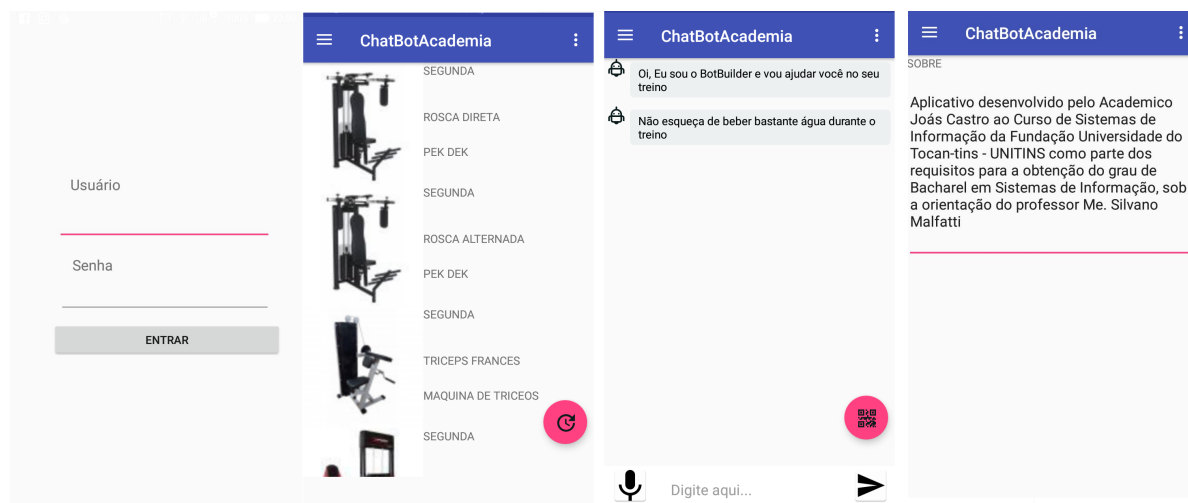


Figura 6 – Telas do Aplicativo
Fonte: O Autor

4.2.1 Integração com o Watson Assistant

O Watson Assistant possui uma biblioteca própria para integração com aplicativos Android. Para tanto, foi necessário adicionar no projeto as anotações no arquivo "build.gradle" conforme mostra a Figura 7.

```
implementation 'com.android.support.design:28.0.0'
implementation 'com.ibm.watson:assistant:7.0.0'
implementation 'com.ibm.watson:text-to-speech:7.0.0'
implementation 'com.ibm.watson:speech-to-text:7.0.0'
```

Figura 7 – Bibliotecas - Watson Assistant
Fonte: O Autor

Para realizar a autenticação com o workspace do *chatbot* criado na página da IBM foi criado o método "createServices()" passando as credenciais do serviço, conforme mostra a Figura 8.

4.2.2 Interação via QR CODE

A interação do aluno com cada aparelho de musculação e a explicação do exercício se dará por meio de instruções fornecidas através da leitura de QR CODEs. Para possibilitar a leitura de QR CODE pelo aplicativo foi utilizado a biblioteca de QR CODE e códigos de barra ZXing¹;

Para fins de padronização foi definido que os QR CODEs devem possuir conteúdos no formato JSON. A Figura 9 apresenta um exemplo de conteúdo de QR CODE apto para ser lido e corretamente mostrado na tela do aplicativo.

¹ <https://github.com/zxing/zxing>

```

104 private void createServices() {
105
106 // Autenticação com o watson usando as credencias do serviço do chatbot
107 watsonAssistant = new Assistant( versionDate: "2018-11-08", new IAMOptions.Builder()
108 .apiKey("a8LCEZkQf2vrLLsaskXQxos87mBWUQWUV_4mUKNyHPf7")
109 .build());
110 watsonAssistant.setEndPoint("https://gateway.watsonplatform.net/assistant/api");
111
112
113 // Autenticação para usar os serviços de texto para fala
114 textToSpeech = new TextToSpeech();
115 textToSpeech.setIAMCredentials(new IAMOptions.Builder()
116 .apiKey("TTS_API_KEY_HERE")
117 .build());
118 textToSpeech.setEndPoint("TTS_URL_HERE");
119
120 // Autenticação para usar os serviços de texto fala para texto
121 speechService = new SpeechToText();
122 speechService.setIAMCredentials(new IAMOptions.Builder()
123 .apiKey("STT_API_KEY_HERE")
124 .build());
125 speechService.setEndPoint("STT_URL_HERE");
126 }

```

Figura 8 – Método de Autenticação com Watson Assistant
Fonte: O Autor

```

{"exercicio": " ", "equipamento": " ": "Instruções": " "}

```

Figura 9 – Exemplo de Conteúdo - QR CODE
Fonte: O Autor

4.2.3 Implementação *Chatbot*

O *chatbot* propriamente dito foi construído dentro de um fragmento relacionado ao *navigation drawer*. No momento em que o usuário seleciona na gaveta de navegação o item de menu relacionado ao *chatbot* o *navigation drawer* exibe o conteúdo do fragmento e assim inicia a interação. O *chatbot* faz uma requisição ao Watson, ele processa a requisição e retorna a mensagem inicial definida e aguarda o usuário digitar alguma mensagem. Se o usuário digitar alguma mensagem o aplicativo realiza uma nova requisição ao Watson passando a mensagem do usuário. O Watson processa a mensagem recebida e responde conforme a intenção identificada. Este fluxo se repete conforme mostra o diagrama de atividade da Figura 10.

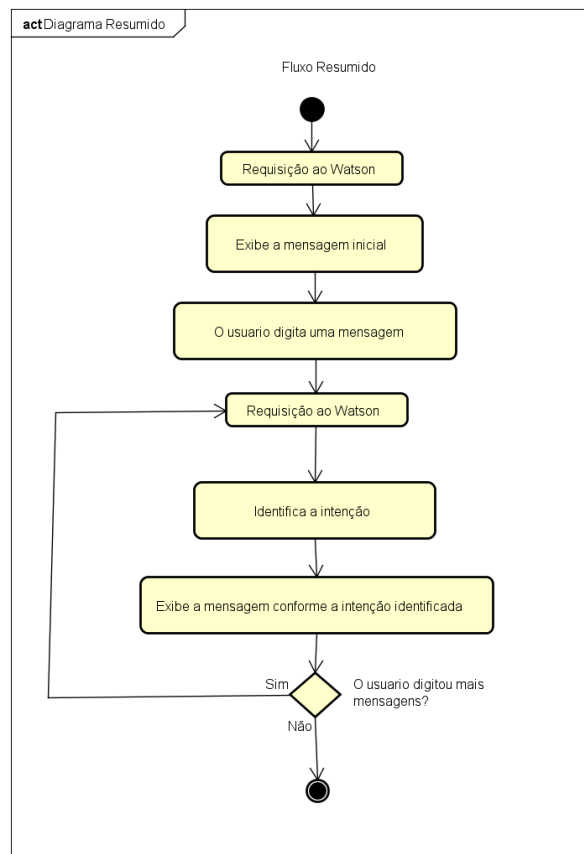


Figura 10 – Diagrama de Atividade - Fluxo Resumido
Fonte: O Autor

Quando o usuário digita uma mensagem que o Watson interpreta como a intenção de iniciar o treino, o aplicativo envia uma requisição para o *webservice* buscando no banco de dados o treino do aluno daquele dia. O *webservice* recebe a requisição e retorna o treino do aluno cadastrado no banco de dados por meio de um objeto JSON. Dessa forma, o aplicativo recebe o objeto JSON faz a leitura dos dados e retorna como resposta para o usuário o exercício e o equipamento que ele deve se dirigir e a orientação para ler o respectivo QR CODE. O Usuário realiza a leitura do QR CODE e o *chatbot* apresenta as instruções de treino contidas no seu conteúdo. A Figura 11 mostra o fluxo completo incluído as interações da intenção treino.

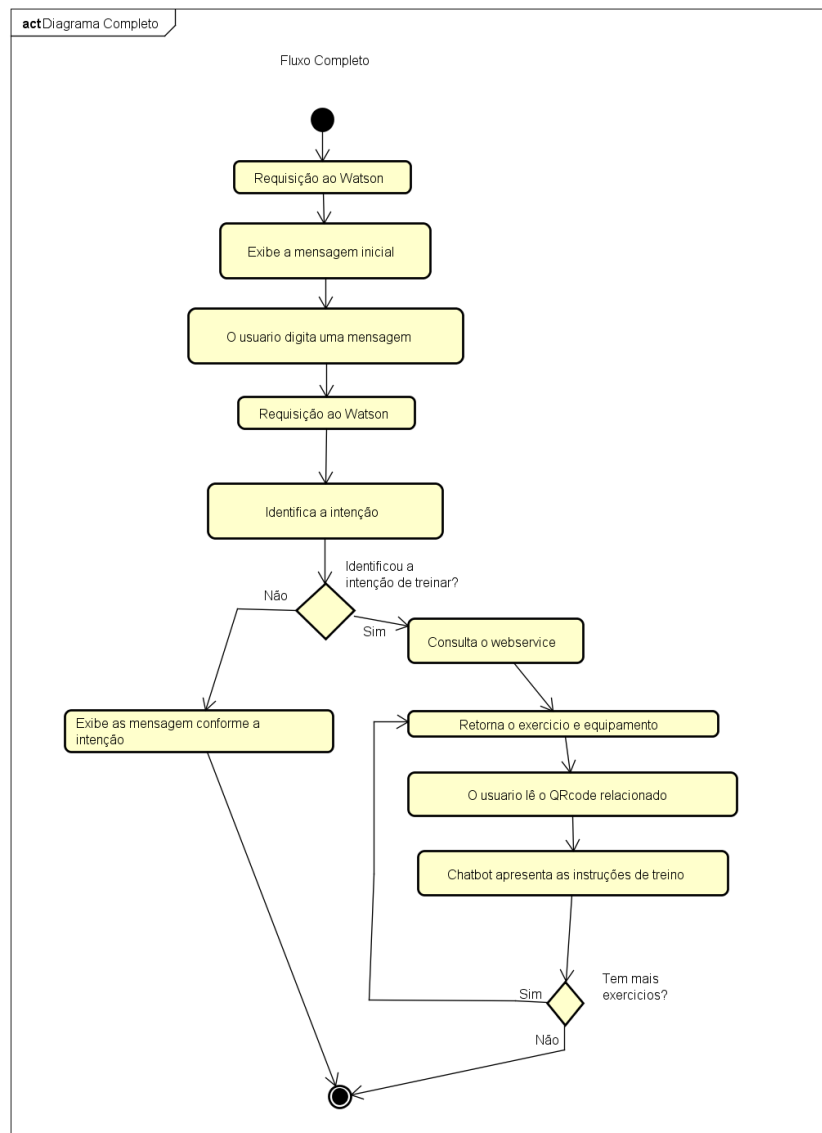


Figura 11 – Diagrama de Atividade - Fluxo Completo
Fonte: O Autor

4.3 Webservice

Foi desenvolvido um *webservice* do tipo RESTFul na linguagem Java. Ele é responsável por realizar as consultas no banco de dados da aplicação web e devolver no formato JSON as respostas ao aplicativo android. O *webservice* foi hospedado localmente pelo servidor de aplicação Tomcat v9.0 instalado internamente no Eclipse. As classes do *webservice* estão organizadas em três pacotes: o modelo, onde estão as classes que representam as entidades, o pacote repository onde contem as classes que buscam as entidades cadastradas no banco de dados e o pacote service onde estão as classes de serviço do *webservice*. A Figura 12 ilustra essa organização.

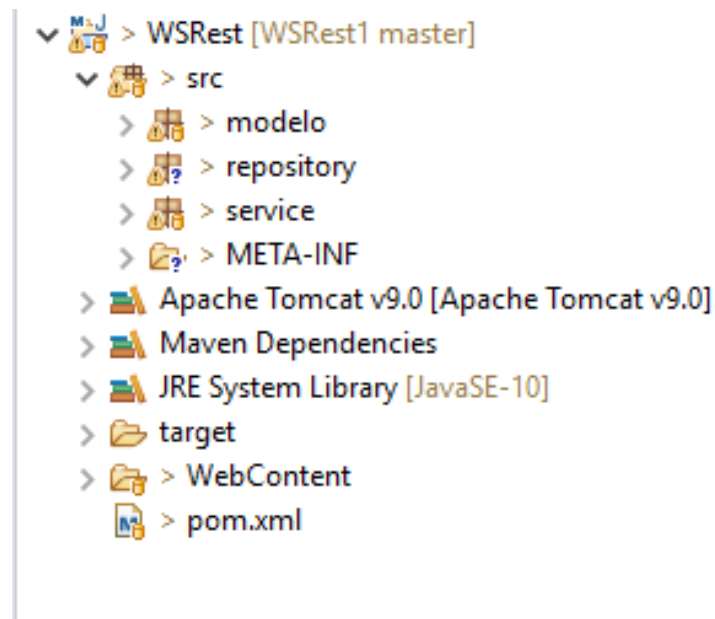


Figura 12 – Webservice Rest
Fonte: O Autor

No pacote repository foi implementados as classes com os métodos que buscam os dados do banco de dados. A Figura 13 mostra como exemplo, o método para buscar a lista de exercícios de determinado aluno.

```

88 // consulta para retornar a lista de exercicio de determinado aluno pelo
89 // dia
90 @SuppressWarnings("unchecked")
91 public List<Treino> getExercicioPordia(Integer aluno, Integer dia) {
92     Query query = getEntityManager().
93     // createQuery("SELECT * FROM treinoaluno a inner join treino t on t.id =
94     // a.idtreino WHERE a.idaluno=:id and t.dia=:dia");
95     createNativeQuery("SELECT exercicio.nome\r\n"
96         + " FROM treinoaluno,treino,treinoexercicio,exercicio,equipamento \r\n"
97         + " where treinoaluno.idtreino = treino.id and treino.id = treinoexercicio.idtreino"
98         + " and"
99         + " treinoexercicio.idexercicio = exercicio.id \r\n"
100         + " and exercicio.idequipamento = equipamento.id\r\n"
101         + " and treinoaluno.idaluno =?aluno and treino.dia =?dia;");
102
103     query.setParameter("aluno", aluno);
104     query.setParameter("dia", dia);
105     List<Treino> lista = query.getResultList();
106
107     if (lista == null)
108         lista = new ArrayList<Treino>();
109     return lista;
110 }

```

Figura 13 – Método Repository - getExercicioPordia
Fonte: O Autor

Para receber essa consulta e converter para o formato JSON foi implementado os respectivos métodos nas classes *services*. Esses são os método que serão chamados pela URL da requisição do *webservice* para obter o retorno. A Figura 14 mostra um exemplo desses métodos.

```

54
55 // metodo que retorna o JSON com a lista de exercicios
56 @GET
57 @Path("/listarExercicio/{aluno}/{dia}")
58 @Produces(MediaType.APPLICATION_JSON)
59 public List<Treino> listarExercicio (@PathParam("aluno") Integer aluno,@PathParam("dia") Integer dia) {
60
61     try {
62         List<Treino> treinoAluno = repository.getExercicioPordia(aluno, dia);
63         return treinoAluno;
64     } catch (Exception e) {
65         throw new WebApplicationException(500);
66     }
67
68
69 }
70

```

Figura 14 – Método Service - listarExercicio
Fonte: O Autor

5 Resultados

Neste capítulo serão apresentados os resultados obtidos com base nos teste realizados.

5.1 Reconhecimento de Intenções

Para verificar se o *chatbot* estava identificando as intenções definidas foram realizados testes enviando mensagens de diversas formas para verificar se é identificado a intenção a ser testada.

Ao iniciar a conversa o *chabot* mostra a mensagem inicial conforme demonstra a Figura 15.

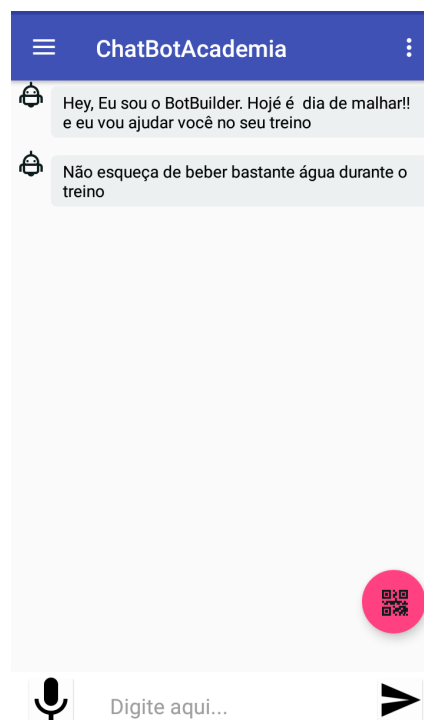


Figura 15 – Mensagem Inicial *Chatbot*
Fonte: O Autor

5.1.1 Reconhecimento da Intenção de Saudação

Para reconhecimento da intenção de saudação que tem a função de saudar o usuário e lhe dar algumas recomendações iniciais foi enviados as seguintes mensagens ilustrada na Figura 16.

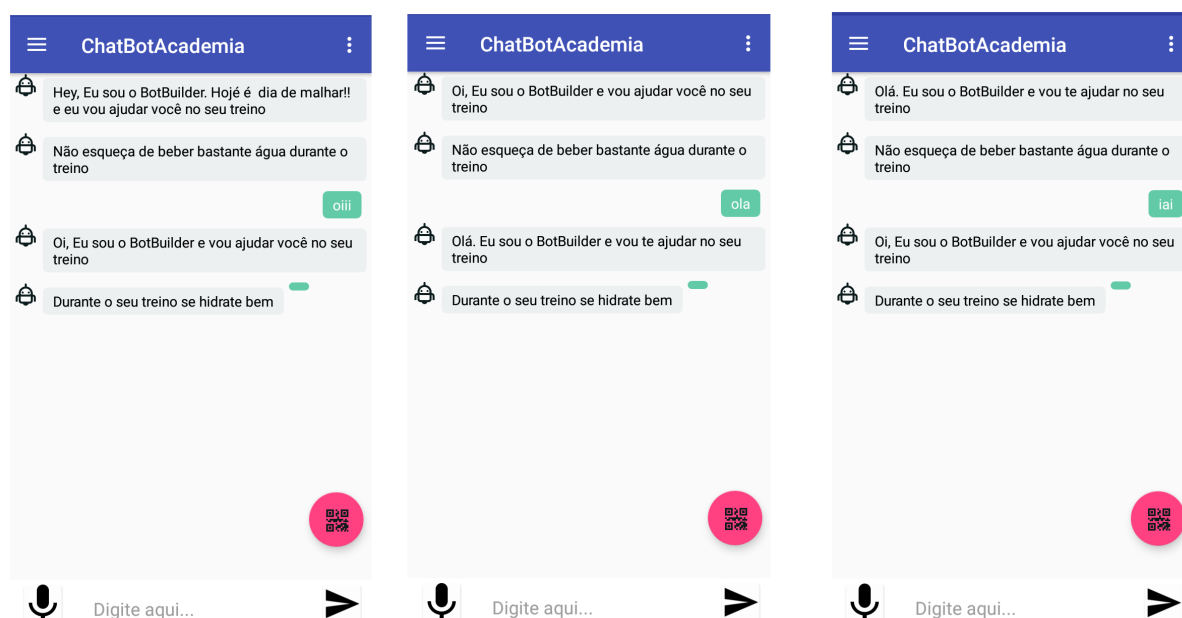


Figura 16 – Reconhecimento da Intenção de Saudação
Fonte: O Autor

5.1.2 Reconhecimento da Intenção de Falar com instrutor

No teste da intenção "Falar com o instrutor" enviamos algumas mensagens no sentido de o usuário solicitar ajuda para uma pessoa no caso o instrutor da academia ou *personal trainer*. Na Figura 17 que mostra os resultados do teste, percebe-se que quando o usuário digita a expressão "preciso de ajuda" o *chatbot* não reconhece corretamente a intenção desejada. Sendo assim, para corrigir é preciso treinar o Watson para que ele reconheça como a intenção desejada.

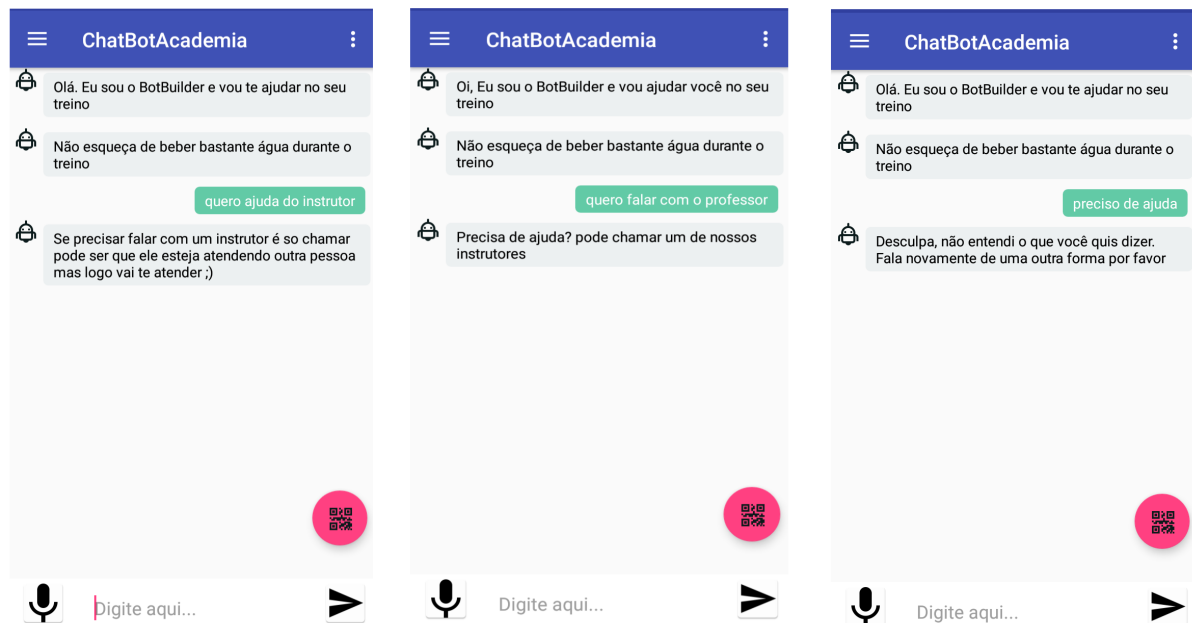


Figura 17 – Reconhecimento da Intenção - Falar com instrutor
Fonte: O Autor

5.1.3 Reconhecimento da Intenção - Treino

O *chatbot* reconhece a intenção "treino" quando o usuário demonstra interesse de iniciar o seu treino daquele dia. A Figura 18 apresenta os resultados do reconhecimento dessa intenção mediante diferentes tipos de mensagens do usuário.

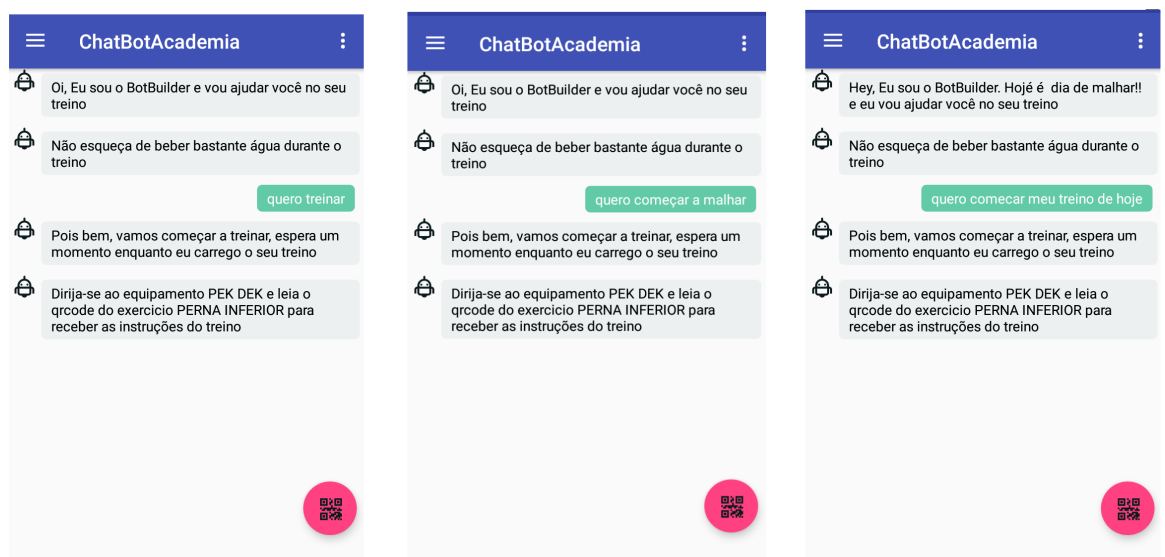


Figura 18 – Reconhecimento da Intenção - Treino
Fonte: O Autor

Quando é reconhecido a intenção de treinar o aplicativo busca no *webservice* o treino do aluno cadastrado no banco de dados para aquele dia e apresenta a mensagem

orientando o aluno ir até o equipamento e realizar a leitura do QRcode relacionado ao exercício.

Para isso o usuário deve tocar no botão que fica no canto inferior direito da tela para ler o QRcode. Assim que o usuário lê o QRcode, o *chatbot* mostra as instruções de treino contidas no conteúdo do QRcode como mostra a Figura 19.

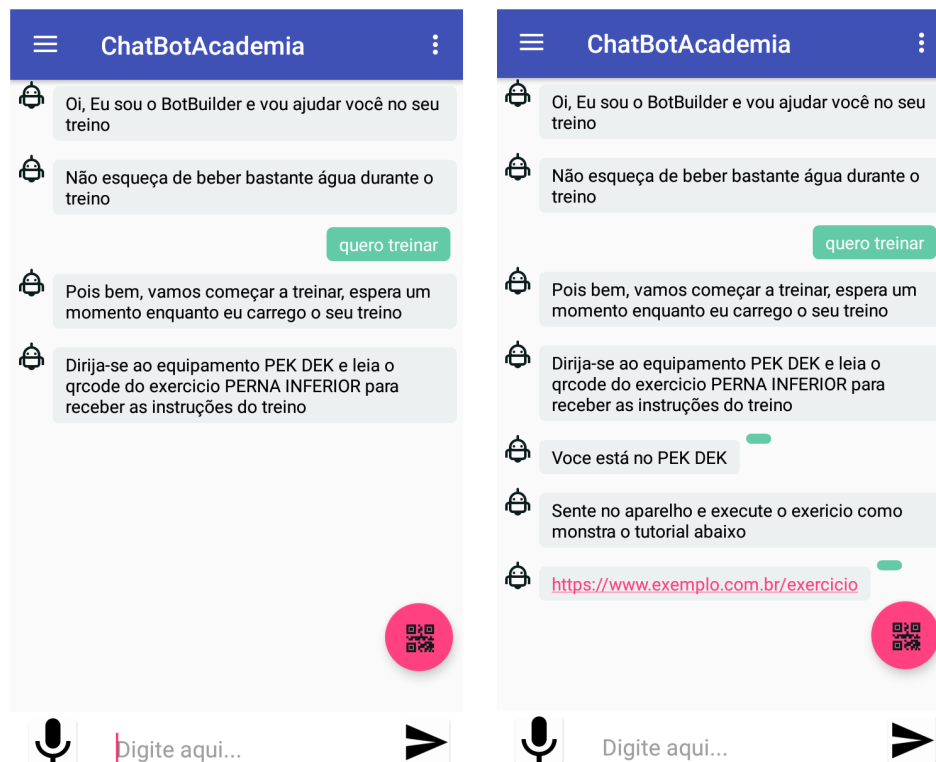


Figura 19 – Reconhecimento Intenção treino e leitura do QRcode
Fonte: O Autor

5.1.4 Reconhecimento da Intenção - Quem é você?

Quando o usuário por curiosidade deseja saber com quem está conversando ou algo semelhante o *chatbot* reconhece a intenção e direciona para o fluxo de conversa denominado "Quem é você?". A Figura 20 mostra o teste realizado com essa intenção.

5.1.5 Reconhecimento da Intenção - Desrespeito

o *chatbot* foi desenvolvido para reconhecer também quando o usuário digita palavras, palavras de baixo calão e insultos em gerais. Quando isso acontece é direcionado para um fluxo de conversa específico para esses casos. A Figura 21 mostra o *chatbot* reconhecendo esses casos.

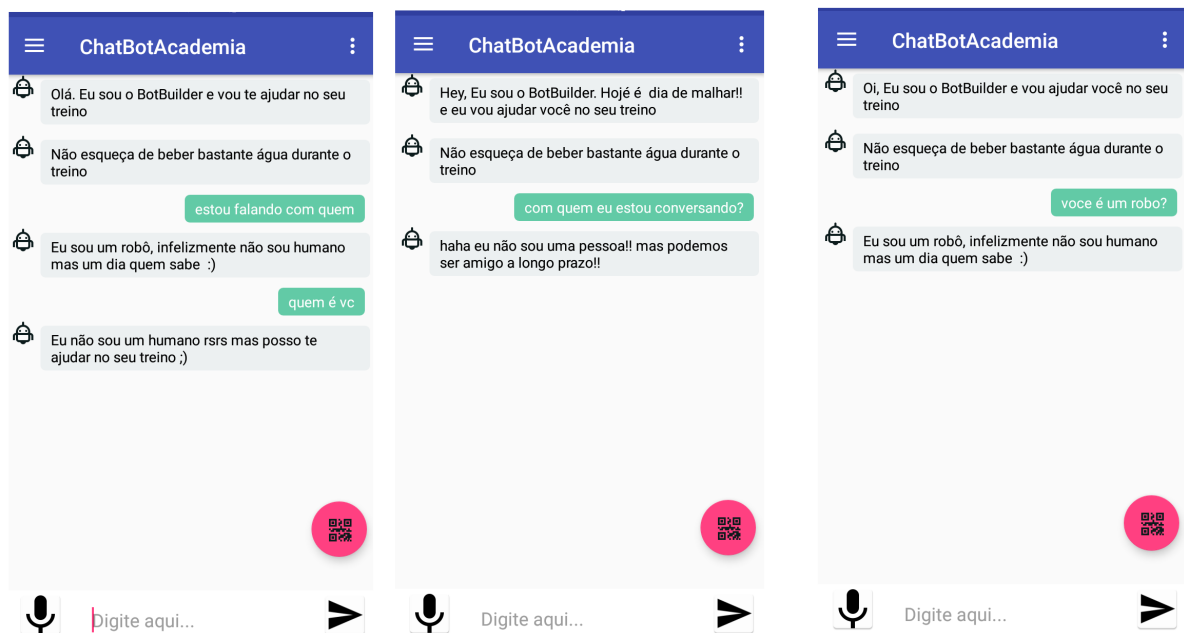


Figura 20 – Reconhecimento da Intenção Quem é você
Fonte: O Autor

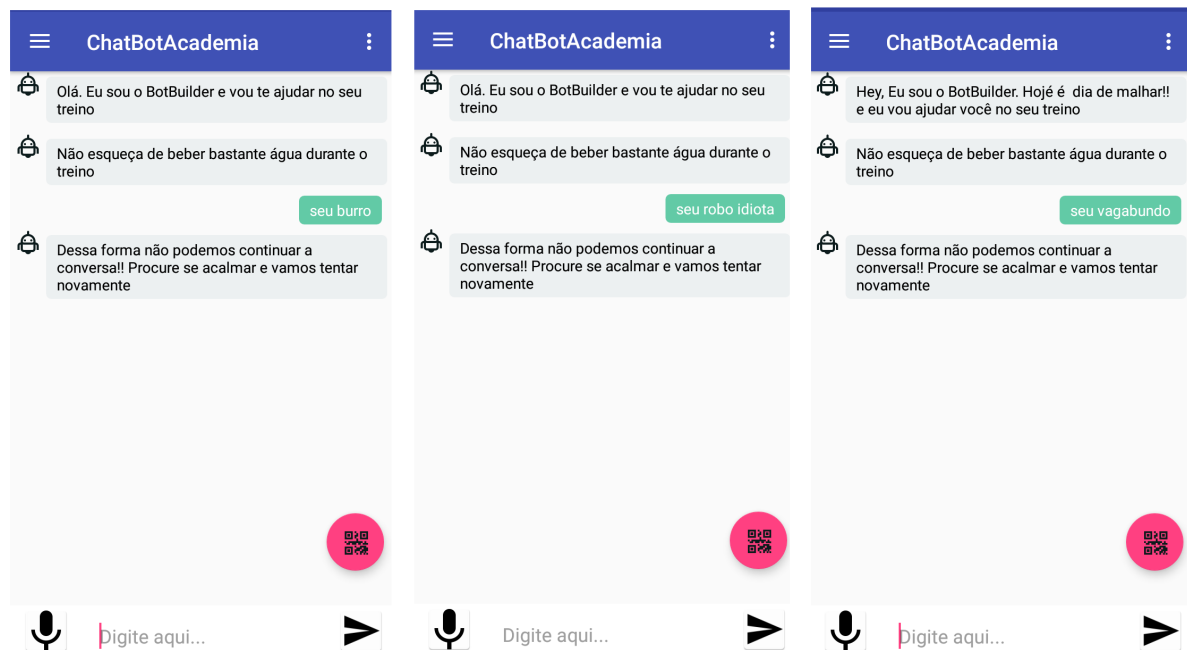


Figura 21 – Reconhecimento da Intenção - Desrespeito
Fonte: O Autor

5.1.6 Não Reconhecimento de Intenção

Quando o usuário digita alguma mensagem que não é relacionada as intenções definidas o *chatbot* direciona para um fluxo específico informando que ele não entendeu. A Figura 22 mostra alguns casos em que isso pode acontecer.

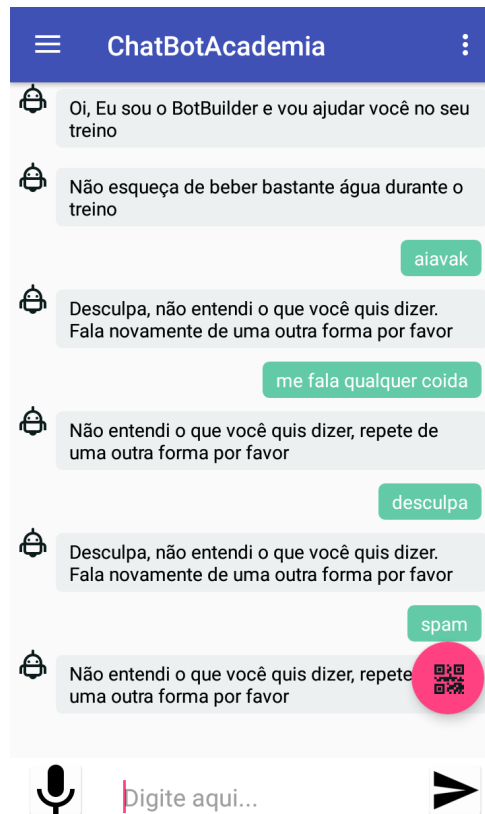


Figura 22 – Intenção Não Reconhecida
Fonte: O Autor

6 Considerações Finais

A finalidade deste trabalho consistiu em desenvolver um *chatbot* para auxílio nos treinos de alunos de academias de musculação. Com base nos resultados deste trabalho podemos concluir que o software desenvolvido é capaz de auxiliar um usuário a realizar os seus exercícios cotidiano em uma academia de musculação e ainda a possibilidade de dinamizar a rotina do treino.

A API utilizada demonstrou uma grande capacidade em reconhecer a intenção do usuário diante das diversas formas que ele possa se expressar, devido a sua habilidade de reconhecimento de linguagem natural advindo da inteligência artificial.

Apesar dos resultados apresentados o contexto em que o *chatbot* está inserido pode ser muito imprevisível, muitas variáveis podem influenciar na consolidação e no efetivo uso do *chatbot*, por exemplo, a resistência no uso de tecnologias por profissionais da área de educação física.

6.1 Trabalhos futuros

Pretende-se como trabalhos futuros

- Definir novas intenções que o *chatbot* seja capaz de reconhecer.
- Desenvolver a capacidade de reconhecer e tratar o contexto da conversa.
- Agregar novas funcionalidades ao aplicativo.

Referências

- Amazon. *Página Oficial*. 2019. Disponível em: <<https://aws.amazon.com/pt/lex/>>. Acesso em: 20 jun. 2019.
- Android Developers. *Site Oficial*. 2019. Disponível em: <<https://developer.android.com/?hl=pt-BR>>. Acesso em: 20 jun. 2019.
- ANTONIO DAS NEVES, S. *Técnicas de Aprendizado de Máquina Aplicadas a Classificação da Qualidade de Pavimentos Asfálticos utilizando Smartphones*. João Monlevade: [s.n.], 2018. Disponível em: <http://www.monografias.ufop.br/bitstream/35400000/799/1/MONOGRAFIA_TécnicasAprendizadoMáquina.pdf>. Acesso em: 20 mai.2019.
- ARMIGLIATTO, G. M. *REST usa JSON e SOAP usa XML, certo?* 2017. Disponível em: <<http://www.matera.com/blog/post/rest-usa-json-e-soap-usa-xml-certo>>. Acesso em: 20 Jun. 2019.
- BRANDES, B. *Dialogflow (api.ai) — Breve introdução da plataforma*. 2017. Disponível em: <<https://medium.com/botsbrasil/api-ai-breve-introducao-da-plataforma->>. Acesso em: 20 Jun. 2019.
- BRITO, F. N.; OLIVEIRA, M. A. de. Desenvolvimento de um Chatterbot para a página web de um curso de nível superior. 2017. Disponível em: <<http://www.repositoriobib.ufc.br/00003a/00003acc.pdf>>. Acesso em: 27 mai. 2019.
- Carrilho Junior, J. R. *Desenvolvimento de uma Metodologia para Mineração de Textos*. 2007. Disponível em: <http://www.lambda.maxwell.ele.puc-rio.br/11675/11675{_}1.>>. Acesso em: 10 mai. 2019.
- CERVO, A. L.; BERVIAN, P. A. *METODOLOGIA CIENTÍFICA*. 5ª. ed. [S.l.: s.n.], 2002. 242 p. ISBN 85-87918-15-X.
- Dal Moro, T. et al. *Web services WS-* versus Web Services REST*. Passo Fundo: [s.n.], 2011. Disponível em: <<https://seer.ufrgs.br/reic/article/view/22140/12928>>. Acesso em: 15 jun. 2019.
- FILHO, E. C. P. O Uso do Processamento de Linguagem Natural na Construção de Chatterbots. p. 46, 2009. Disponível em: <<https://dcc.catalao.ufg.br/up/498/o/Eustaquio2009.pdf>>. Acesso em: 10 mar. 2019.
- FONSECA, A. R.; ALENCAR, M. S. d. M. O uso de aplicativos de saúde para dispositivos móveis como fontes de informação e educação em saúde. *XIX Seminário Nacional de Bibliotecas Universitárias – SNBU*, 2016.
- FREITAS, L. C. D. *WEB SERVICES*. 2006. Disponível em: <<http://www.leonardofreitas.eti.br/download/TCC-LEONARDO-WEBSERVICES.pdf>>. Acesso em: 15 jun. 2019.
- GALVÃO, A. d. M. “ Persona-AI ML : Uma Arquitetura para Desenvolver Chatterbots. 2003. Disponível em: <<https://repositorio.ufpe.br/handle/123456789/2473>>. Acesso em: 07 nov. 2018.

IBM. *Assistente Watson v1 - Documentos da API da nuvem IBM*. Disponível em: <<https://console.bluemix.net/apidocs/assistant?language=node#introduction>>. Acesso em: 20 nov. 2018.

IBM. *IBM CLOUD DOCS / IBM ASSISTANT*. Disponível em: <<https://console.bluemix.net/docs/services/assistant/index.html#sobre>>. Acesso em: 20 nov. 2018.

James White. Going native (or not): Five questions to ask mobile application developers. *Australasian Medical Journal*, 2013. Acesso em: 20 jun. 2019.

KALIN, M. *Java Web Services: Up and Running*. 2ª. ed. [S.l.: s.n.], 2013. ISBN 978-1-449-36511-0.

KAR, R.; HALDAR, R. *Applying Chatbots to the Internet of Things: Opportunities and Architectural Elements*. 2016. Disponível em: <<http://thesai.org/Publications/ViewPaper?Volume=7{&}Issue=11{&}Code=ijacsa{&}Se>>. Acesso em: 07 nov. 2018.

LECHETA, R. R. *Web Services Restful: Aprenda a Criar web Services RESTful em Java na Nuvem do Google*. 1ª. ed. [S.l.: s.n.], 2015. ISBN 978-8575224540.

LIZ, C. M. D.; ANDRADE, A. Motivação de Aderentes e Desistentes da Musculação em Academias. 2013. Disponível em: <<http://congressos.cbce.org.br/index.php/conbrace2013/5conice/paper/download/5123/2529>>. Acesso em: 27 nov. 2018.

MANFIO, E. R. *A EVOLUÇÃO DOS CHATTERBOTS: PLN, I.A. E DIFUSÃO CULTURAL*. 2015. Disponível em: <<https://www.inbot.com.br/artigos/educacional/A-EVOLUCAO-DOS-CHATTERBOTS-PLN-IA-FATEC.pdf>>. Acesso em: 20 jun. 2019.

MARCONI, M. d. A.; LAKATOS, E. M. *fundamentos de metodologia científica*. 7ª. ed. São Paulo: [s.n.], 2010. 297 p. ISBN 9788522457588.

NASCIMENTO JR., C. L.; YONEYAMA, T. *INTELIGÊNCIA ARTIFICIAL EM CONTROLE E AUTOMAÇÃO*. 1ª. ed. São Paulo: [s.n.], 2004. 218 p. ISBN 9788521203100.

NILSSON, N. J. *Principles of Artificial Intelligence*. 1ª. ed. [S.l.: s.n.], 1986. ISBN 978-3-540-11340-9.

OLIVEIRA, R. R. *Avaliação de manutenibilidade entre as abordagens de web services RESful e SOAP-WSDL*. São Carlos: [s.n.], 2012. Disponível em: <<http://www.teses.usp.br/teses/disponiveis/55/55134/tde-24072012-164751/publico/RRrevisada.pdf>>. Acesso em: 17 jun. 2019.

PEREIRA, G.; PINHEIRO, M. A. Conversando com robôs: O uso de chatbots na comunicação de marcas no Facebook Messenger. *Intercom – Sociedade Brasileira de Estudos Interdisciplinares da Comunicação*, 2018. Disponível em: <<http://portalintercom.org.br/anais/nordeste2013/resumos/R37-0415-1.pdf>>. Acesso em: 08 nov. 2018.

PRODANOV, C. C.; FREITAS, E. C. de. *Metodologia do Trabalho Científico: Métodos e Técnicas da Pesquisa e do Trabalho Acadêmico*. 2ª. ed. [S.l.: s.n.], 2013. ISBN 978-85-7717-158-3.

Rede Movimentar. *MUSCULAÇÃO: A IMPORTÂNCIA DO ACOMPANHAMENTO ADEQUADO*. 2016. Disponível em: <<https://redemovimentar.com.br/detalhe.php?id=27804{&}cs=b32389{&}url=MUSCULACAO:-A-IMPORTANCIA-DO-ACOMPANHAMENTO-AD>>. Acesso em: 20 jun. 2019.

RICH, E. *Artificial Intelligence*. New York, NY, USA: McGraw-Hill, Inc., 1983. ISBN 0-07-052261-8.

Rodizio Bento, A. et al. Importância da tecnologia no acompanhamento das atividades dos alunos de academia. 2013. Disponível em: <http://convibra.com/upload/paper/2013/29/2013{_}29{_}81>. Acesso em: 27 nov. 2018.

ROTHERMEL, A.; DOMINGUES, M. J. C. d. S. MARIA : Um chatterbot desenvolvido para os estudantes da disciplina “ Métodos e Técnicas de Pesquisa em Administração ”. *SEGeT - Simpósio de Excelência em Gestão e Tecnologia*, p. 1–12, 2007.

RUSSEL, S.; NORVING, P. *Inteligência Artificial*. 3ª. ed. Rio de Janeiro: [s.n.], 2010. . 1324 p. ISBN 978-85-352-3701-6.

SILVA, M. M. da; SANTOS, M. T. P. Os Paradigmas de Desenvolvimento de Aplicativos para Aparelhos Celulares. 2014. Disponível em: <<http://revistatis.dc.ufscar.br/index.php/revista/article/view/86/80>>. Acesso em: 20 jun. 2019.

TEIXEIRA, S.; MENEZES, C. S. de. Facilitando o uso de Ambientes Virtuais através de Agentes de Conversação. *Brazilian Symposium on Computers in Education (Simpósio Brasileiro de Informática na Educação - SBIE)*, 2003. Disponível em: <<http://www.br-ie.org/pub/index.php/sbie/article/view/276>>. Acesso em: 06 nov. 2018.

WEIZENBAUM, J. ELIZA—A Computer Program For the Study of Natural Language Communication Between Man and Machine. 1966. Disponível em: <<http://portal.acm.org/citation.cfm?doid=365153.365168>>. Acesso em: 07 nov. 2018.

WITTEN, I. H.; FRANK, E. *Datamining. Practical Machine Learning Tools and Technicals with Java Implementations*. 2ª. ed. San Francisco: [s.n.], 2005. ISBN 0-12-088407-0.