

FACULDADE DE ENGENHARIA DA UNIVERSIDADE DO PORTO

Assistentes Virtuais para Comunicação Empresarial

João Carlos Gonçalves Costa



FEUP FACULDADE DE ENGENHARIA
UNIVERSIDADE DO PORTO

Mestrado Integrado em Engenharia Informática e Computação

Orientador: Henrique Daniel de Avelar Lopes Cardoso

24 de Julho de 2017

Assistentes Virtuais para Comunicação Empresarial

João Carlos Gonçalves Costa

Mestrado Integrado em Engenharia Informática e Computação

Resumo

Nos últimos anos, o número de dispositivos com acesso à Internet aumentou, assim como a eficiência da comunicação remota. Não obstante o facto de existirem hoje inúmeros canais de comunicação, como *chat*, *email*, vídeo, as pessoas continuam a considerar as chamadas de voz como o canal de comunicação que oferece uma experiência mais pessoal em tempo real. Contudo, no mundo empresarial, as pessoas estão constantemente ocupadas com reuniões e tarefas, não conseguindo dar resposta imediata a todos os contactos que recebem. Por outro lado, cada vez mais empresas optam pela terceirização de certas tarefas, assim como permitem aos seus funcionários trabalharem remotamente a partir de sua casa. Isto levou a um aumento das audiokonferências telefónicas, chamadas telefónicas com mais do que apenas dois participantes. Estas permitem a realização de reuniões entre equipas de trabalho localizadas em diferentes locais.

No seguimento do contexto apresentado, esta dissertação foi desenvolvida partindo de dois objetivos iniciais. O primeiro centrou-se no desenvolvimento de um sistema que, controlado exclusivamente por voz, fosse capaz de gerir diversas funcionalidades necessárias à realização de uma audiokonferência telefónica, como por exemplo, o agendamento ou a ligação à mesma. O sistema tira partido de um assistente virtual que comunica com o utilizador de forma semelhante à humana. Conjuntamente, e este foi o segundo objetivo desta dissertação, desenvolveu-se um protótipo de um sistema para, tirando partido de uma aplicação Android, responder a todas as chamadas de voz nativa recebidas por um utilizador que se encontra ocupado. Uma chamada nativa consiste numa chamada de telefone "regular" usando a *public switched telephone network (PSTN)*. A aplicação desenvolvida detecta uma tentativa de chamada de voz nativa e inicia prontamente uma sessão de comunicação com a pessoa que iniciou a chamada, através de um assistente virtual (*bot*). O assistente virtual pergunta ao utilizador qual foi o motivo da chamada e interactivamente oferece algumas opções à pessoa que tentou contactar. Refira-se que, à data de término desta dissertação, ambos os assistentes virtuais desenvolvidos funcionam de forma independente.

Já existem hoje em dia ferramentas e dispositivos de reconhecimento de fala, assim como sistemas que fazem o processo inverso, sistemas texto-voz. Na elaboração dos assistentes virtuais tirou-se partido das ferramentas referidas, nomeadamente do *Amazon Lex* e *Amazon Alexa* para entendimento de linguagem natural e criação de fluxos conversacionais e do *Google Cloud Speech API* para conversão de áudio em texto. O dispositivo *Amazon Echo Dot*, uma coluna *wireless* que permite receber controlos por voz, foi usado para tirar partido das capacidades da *Amazon Alexa* e permitir uma comunicação por voz com o utilizador. Utilizou-se também a plataforma de comunicações *Twilio*, que permite receber, controlar e monitorizar chamadas telefónicas.

A avaliação da eficiência dos protótipos realizados foi conseguida com base em inquéritos de satisfação dos utilizadores e em testes de usabilidade. Com ambos sistemas desenvolvidos conseguiu-se dar resposta às questões que se levantaram. Contudo, o sistema de gestão de audiokonferências telefónicas não apresentou uma eficácia e recetividade por parte dos utilizadores tão elevada como o sistema criado para dar resposta às chamadas recebidas por um utilizador que se encontra ocupado.

Abstract

In recent years, the number of devices with Internet access as well as the efficiency of remote communications have increased. Despite the fact that today there are numerous communication channels such as chat, email and video, people still consider voice calls as the communication channel that offers a more personal experience in real time. However, in the business world, people are constantly busy with meetings and tasks, failing to respond immediately to all contacts they receive. On the other hand, more and more companies choose to outsource certain tasks as well as allow their employees to work remotely from their home. This made an increase in telephone audioconferences, telephone calls with more than just two participants. The audioconferences allow meetings between work teams located in different locations.

Following the presented context, this dissertation was developed starting from two initial objectives. The first focused on the development of a system which, controlled exclusively by voice, would be able to manage several functionalities necessary to perform a telephone conference call, such as scheduling it or establishing its connection. The system takes advantage of a virtual assistant which communicates with the user in a way similar to a human.

Together, and this was the second objective of this dissertation, a prototype of a system was developed to take advantage of an Android application for responding to all the native voice calls received by a busy user. A native call consists of a "regular" telephone call using the public switched telephone network (PSTN). The developed application detects a native voice call attempt and promptly starts a communication session with the person initiating the call through a virtual assistant (bot). The virtual assistant asks the user for the reason for the call and interactively offers some options to the person they tried to contact. It should be noted that at the end of this dissertation, both developed virtual assistants work independently.

Nowadays there are tools and devices for speech recognition as well as tools for speech synthesis. The development of the virtual assistants took advantage of the mentioned tools, namely Amazon Lex and Amazon Alexa for understanding natural language and creating conversational flows, and the Google Cloud Speech API for converting audio to text. The Amazon Echo Dot device, a wireless speaker which allows the user to receive voice controls, was used to take advantage of Amazon's capabilities and enable voice communications with the user. The Twilio communication platform was also used, which allows, through an API, to receive, control and monitor telephone calls.

The evaluation of the efficiency of the realized prototypes was achieved based on user satisfaction surveys and usability tests. With both developed systems, the questions that arose were able to be answered. However, the telephone conference management system has not been as effective and responsive to users as the system created to respond to calls received by a busy user.

Agradecimentos

Gostaria de agradecer, em primeiro lugar, a toda a minha família que me apoiou em todo o meu percurso académico e certamente me continuará a apoiar, em especial aos meus pais e ao meu irmão.

Agradeço também ao meu orientador da Faculdade de Engenharia da Universidade do Porto, Henrique Lopes Cardoso, que sempre se mostrou disponível para ajudar no que fosse necessário.

Quero agradecer também à empresa *Wit-Software*, proponente desta dissertação, por me ter permitido desenvolver esta dissertação dentro das suas instalações e pelo acompanhamento próximo que sempre me foi dado. Agradeço também a todos os colaboradores desta empresa, em especial ao Paulo Sousa, ao Pedro Andrade, ao Jorge Sousa e à Simone Guedes.

Gostaria de agradecer também a todos os meus amigos, aos que já conhecia, aos que tive a oportunidade de fazer durante esta dissertação, e a uma amiga especial, que me apoiaram e me deram força para fazer sempre mais e melhor.

João Carlos Gonçalves Costa

*“If you’re not failing every now and again,
it’s a sign you’re not doing anything very innovative.”*

Woody Allen

Conteúdo

Agradecimentos	v
1 Introdução	1
1.1 Contexto	1
1.2 Motivação e Objetivos	2
1.3 Estrutura	4
2 Revisão Bibliográfica	5
2.1 Processamento de Linguagem Natural (<i>NLP</i>)	5
2.1.1 Técnicas para processamento de texto	6
2.2 Ferramentas de <i>NLP</i>	7
2.2.1 The Stanford CoreNLP Natural Language Processing Toolkit	7
2.2.2 <i>Amazon Lex</i>	8
2.2.3 <i>API.AI</i>	9
2.2.4 Comparação de ferramentas	9
2.3 Tecnologias e protocolos de comunicação	10
2.3.1 <i>VoIP</i>	11
2.3.2 Protocolo de iniciação de sessão (<i>SIP</i>)	12
2.3.3 Sistema Interativo de Resposta por Voz (<i>IVR</i>)	13
2.4 Sistemas de audioconferência	14
2.4.1 <i>WebEx</i>	14
2.4.2 Conference Calling by Vastconference	14
2.4.3 Infinite Conferencing	15
2.5 Conclusões	15
3 Wit Bot Platform	17
3.1 <i>Channel Gateway</i>	17
3.2 <i>Bot Engine</i>	18
3.2.1 <i>NLP Engine</i>	18
3.2.2 <i>AIML Engine</i>	18
3.2.3 <i>BPMN Engine</i>	19
3.3 <i>Integration</i>	20
3.4 Conclusões	20
4 Assistente Virtual para gestão de conferências telefônicas	21
4.1 Inovação empresarial e surgimento de novas tecnologias	21
4.2 Sistema de agendamento de audioconferências telefônicas através de um assistente virtual controlado por voz	22

CONTEÚDO

4.3	Implementação	23
4.3.1	Ferramentas e tecnologias utilizadas	23
4.3.2	Arquitetura	23
4.3.3	Amazon Alexa	24
4.3.4	Servidor/ <i>Wit Bot Platform</i>	25
4.3.5	Casos de uso funcionais	27
4.4	Conclusões	29
5	Assistente virtual de chamadas recebidas	31
5.1	Atendimento automático de chamadas	31
5.2	Aplicação móvel e assistente virtual	34
5.3	Implementação	34
5.3.1	Ferramentas e tecnologias adotadas	34
5.3.2	Arquitetura	35
5.3.3	Servidor	36
5.3.4	Amazon Lex	37
5.3.5	Aplicação móvel	37
5.3.6	Comunicação Aplicação móvel Servidor	40
5.3.7	Principais bibliotecas externas utilizadas	40
5.4	Conclusões	41
6	Validação e resultados	43
6.1	Assistente Virtual para agendamento de conferências telefônicas	44
6.1.1	Testes de usabilidade	45
6.1.2	Inquérito sumário	47
6.2	Assistente Virtual de chamadas recebidas	48
6.2.1	Testes de usabilidade	48
6.2.2	Inquérito sumário	48
6.3	Conclusões	49
7	Conclusões e trabalho futuro	51
	Referências	53
A	Interaction Model - Alexa Skill	59
B	Fluxos - Bizagi Modeler	69
B.1	Fluxos criados de raiz	70
B.2	Fluxos adaptados	73
B.3	Fluxos sem alterações	75
C	Inquérito introdutório - Assistente virtual de chamadas recebidas	79
C.1	Inquérito	79
C.2	Respostas	81
D	Inquérito final - Assistente virtual de chamadas recebidas	87
D.1	Inquérito	87
D.2	Respostas	88

CONTEÚDO

E	Inquérito final - Assistente virtual para gestão de audioconferências telefônicas	89
E.1	Inquérito	89
E.2	Respostas	90

CONTEÚDO

Lista de Figuras

2.1	Rede de comunicação <i>SIP</i> [SJ08]	12
2.2	Fluxo de chamadas de sistema <i>IVR</i> [SJ08](adaptado)	13
3.1	Arquitetura da <i>Wit Bot Platform</i>	17
3.2	Exemplo de fluxo conversacional implementado com o <i>Bizagi Modeler</i>	19
4.1	Arquitetura global do sistema	24
4.2	Exemplo de intenção para agendar uma audioconferência com dia semanal e hora	24
4.3	<i>Amazon Alexa Card</i> para associar conta Wit-software e <i>Amazon</i>	25
4.4	Processo de associação de contas	26
4.5	Vista de lista de convidados na <i>Amazon Alexa App</i>	27
4.6	Vista de conferencia agendada na <i>Amazon Alexa App</i>	28
4.7	Vista de resposta ao enunciado "Who is Tiago Ferreira"na <i>Amazon Alexa App</i>	29
5.1	Resultados de utilização e conhecimento das tecnologias de correio de voz e voz para SMS por parte da amostra	32
5.2	Resultados de satisfação com a qualidade do serviço das por parte da amostra	33
5.3	Resultados de serviço preferencial de chamadas de voz	33
5.4	Arquitectura de alto nível	36
5.5	Exemplo de mensagem no formato <i>Twiml</i>	36
5.6	Visualização do menu principal da aplicação	38
5.7	Visualização da lista de chamadas respondidas pelo assistente virtual na aplicação	39
5.8	Visualização da vista de <i>chat</i> da aplicação	39

LISTA DE FIGURAS

Lista de Tabelas

2.1	Comparação de ferramentas [Lex17, Rod17, try17]	10
2.2	Diferenças entre <i>VoIP</i> e <i>PSTN</i>	11
6.1	Heurísticas para <i>chat bots</i>	44
6.2	Resultados da análise dos teste de usabilidade com base nas seis heurísticas para <i>chat bots</i>	46
6.3	Resultados importantes ao inquérito sobre o assistente virtual para chamadas recebidas	47
6.4	Resultados importantes ao inquérito referente ao assistente virtual de camadas recebidas	49

LISTA DE TABELAS

Abreviaturas e Símbolos

ASR	Reconhecimento automático de fala
NLU	Entendimento de linguagem natural
NLP	Processamento de linguagem natural
FOL	Lógica de primeira ordem
DAG	Grafo acíclico dirigido
XML	Linguagem extensível de marcação genérica
RDF	Linguagem para representar informação na Internet
OWL	Linguagem der ontologias na Web
API	Interface de programação de aplicações
WWW	<i>World Wide Web</i>
SIP	Protocolo de iniciação de sessão
IVR	Unidade de resposta audível
HTTP	Protocolo de transferência de hipertexto
PTSN	Rede pública de telefonia comutada

Capítulo 1

Introdução

Neste capítulo será exposto sucintamente o contexto e a motivação do trabalho descrito nesta dissertação, os seus principais objetivos e a estrutura da mesma.

1.1 Contexto

Os telemóveis são os dispositivos de comunicação mais bem-sucedidos de todos os tempos [Día05]. O lançamento comercial destes dispositivos levou muito tempo desde a criação dos primeiros protótipos, mas hoje em dia desempenham um papel indispensável no quotidiano de qualquer pessoa de um país desenvolvido, desde jovens estudantes a empresários de sucesso. Os telemóveis não são apenas um meio de comunicação, mas podem também desempenhar outros papéis devido à sua capacidade de armazenar uma grande quantidade de informação [Día05].

Atualmente já não se fala em telemóveis, mas sim em telefones inteligentes: *smartphones*. O enorme volume de vendas dos *smartphones* demonstra por si só o seu impacto considerável na sociedade. Os fabricantes e o marketing podem ser culpados por esse *hype*, mas não há dúvidas de que estes dispositivos apresentam grandes recursos e capacidades, como o permitir uma conexão rápida à Internet a partir de qualquer lugar, oferecer recursos educacionais e de negócio e, até mesmo, permitir a realização de tarefas que antigamente só eram possíveis com um computador ou *notebook* [SS13]. Assim se percebe a grande importância destes dispositivos no que diz respeito às pessoas que estão ligadas ao mundo empresarial.

Contudo, os trabalhadores despendem imenso tempo em situações que os impossibilitam de responder atempadamente às chamadas recebidas [AB01]. Uma dessas situações, segundo [DS76], são as reuniões empresariais, que cada vez mais se tornam numa constante nas mais diversas empresas. Para além disso, cada vez mais as empresas optam pela terceirização dos seus serviços. Existem também cada vez mais empresas que permitem aos seus funcionários trabalharem remotamente a partir de suas casas [LKW09, Wat17a]. Consequentemente, aumentou o número de

Introdução

audioconferências telefônicas entre funcionários dentro de cada empresa e entre empresas distintas [Wat17b]. Uma audioconferência telefônica consiste numa chamada telefônica onde comunicam mais do que dois participantes, sendo usadas frequentemente para reuniões e debates sobre determinados assuntos.

Existem hoje diversos canais de comunicação, como por exemplo, canais de conversação (*chat*), *email* ou vídeo. Ainda assim, as pessoas na sua generalidade continuam a preferir as chamadas de voz nativas, isto é, chamadas de telefone “regular” usando a rede *public switched telephone network (PSTN)* [Pet03].

Segundo [Día05], parte dos utilizadores continua a evitar o uso de tecnologias como correio de voz (*voice mail*), que permite aos seus utilizadores trocar mensagens de voz quando uma chamada não pode ser estabelecida, e ainda a ferramenta *voice-to-text*, que possibilita converter voz em texto e transmitir este último sob a forma de mensagem textual para a pessoa com quem se pretendia estabelecer contacto mas que não respondeu à chamada de voz [CJP03]. As razões apontadas para relutância dos utilizadores em utilizar o correio de voz estão relacionadas com a sua interpretação do que é móvel. Os utilizadores percebem que o recetor não está disponível quando este último não responde à chamada. E, graças à mensagem de chamada perdida, o recetor saberá que foi contactado e retornará a chamada, pelo que não faz sentido perder tempo e dinheiro com o correio de voz [Día05].

Refira-se que, esta dissertação foi proposta pela empresa Wit-Software. Esta empresa, através dos seus colaboradores, apoiou sempre de perto o desenvolvimento desta dissertação, ajudando a tomar decisões e prestando auxílio sempre que necessário. A Wit-Software é uma empresa que procura criar soluções avançadas e produtos de marca branca para a indústria das telecomunicações móveis. A empresa pretende desenhar o futuro das comunicações móveis e inspirar as pessoas pela diferença que impõe nos produtos que desenvolve.

1.2 Motivação e Objetivos

Atendendo ao enquadramento apresentado, surgem algumas questões a que se pretende dar resposta. Apresenta-se de seguida a listagem de algumas dessas questões relativas ao processo de gestão de audioconferências:

- Como tornar o processo de gestão de audioconferências mais simples e rápido?
- Será que os novos dispositivos de comunicação por voz podem ter um papel importante nesse processo?
- Serão estes dispositivos uma mais-valia? Ou a comunicação por voz nem sempre acelera os processos?

Seguidamente, focando a motivação na constante taxa de ocupação dos funcionários e na sua relutância no uso do correio de voz e nos sistemas *voice-to-text*, surgem novas questões pertinentes:

Introdução

- Existirá alguma utilidade na criação de um sistema que consiga dar resposta a uma chamada sem intervenção do recetor da chamada ou alguém no seu papel?
- De que maneira se conseguirá conceber esse sistema?
- Como garantir que esse sistema autónomo seja simultaneamente eficaz, intuitivo e atrativo para o utilizador?
- Será possível e plausível dar resposta às chamadas recebidas sem a decisão final do utilizador? Que autonomia deverá o sistema possuir?
- Conseguirá este sistema competir ou ser mais vantajoso que as alternativas já existentes?

Assim, este projeto de dissertação tem como objetivos a elaboração de dois sistemas independentes que, tirando partido das ferramentas e dispositivos mais recentes de processamento e entendimento de linguagem natural, consigam responder às questões apresentadas.

Assistente Virtual para gestão de audioconferências telefónicas – O primeiro sistema consiste, tal como o nome indica, num assistente virtual que, através de comunicação por voz, permite gerir todo o processo de gestão de audioconferências telefónicas. Na verdade este sistema é capaz de agendar e listar audioconferências e de ligar o utilizador a uma audioconferência. Entre outras capacidades importantes no processo de gestão de audioconferências. Este assistente comunica com um servidor de audioconferências telefónicas e liberta os utilizadores dos processos de introdução de *PINs* usados regularmente nos processos de gestão de audioconferências telefónicas. Para o desenvolvimento deste sistema utilizou-se o *Amazon Echo Dot*, uma coluna *wireless* que recebe controlos por voz e permite tirar partido das capacidades da *Amazon Alexa*. A *Amazon Alexa* é um serviço que dá aos seus utilizadores capacidades de interagir com dispositivos de forma intuitiva e interativa usando a voz, assim como criar fluxos conversacionais [ale17].

Assistente Virtual de chamadas recebidas – O segundo sistema desenvolvido tira partido de uma aplicação Android e permite dar resposta imediata às chamadas de voz recebidas pelo seu utilizador. Com este sistema resolveu-se o problema dos utilizadores que estão ocupados para responder a chamadas que recebem. Além disso, este sistema evita o uso de tecnologias como o *voice mail* e a "voz para SMS" que, como já se conseguiu concluir, são muitas vezes ignoradas pelos utilizadores de *smartphones* [Día05]. O sistema deteta uma tentativa de chamada de voz nativa e, automaticamente, inicia uma sessão de comunicação com a pessoa que tentou contactar o utilizador da aplicação. Para estabelecer a sessão de comunicação elaborou-se um assistente virtual (*bot*), que é capaz de perceber o motivo da chamada e oferecer, de forma interativa, opções ao interlocutor. As opções oferecidas são, por exemplo, o agendamento de um novo contacto para mais tarde. Permitir ao utilizador final tomar certas decisões durante a conversação é outra das funcionalidades que o assistente virtual é capaz de executar. No processo descrito, foram utilizadas tecnologias já existentes de reconhecimento de fala, assim como de conversão de texto em voz. Existem ainda serviços que disponibilizam funcionalidades avançadas de *deep learning* para reconhecimento automático de fala (*automatic speech recognition, ASR*), para a conversão de voz

em texto e compreensão de linguagem natural (*natural language understanding, NLU*), para o reconhecimento da intenção do texto, que poderão permitir a criação de aplicações com experiências do utilizador altamente enriquecedoras e interações com conversas que pareçam reais [ama17a].

1.3 Estrutura

Esta dissertação é composta por seis capítulos adicionais. No Capítulo 2, é apresentada uma revisão sumária das investigações que se têm feito no domínio do processamento de linguagem natural. São apresentadas algumas ferramentas relevantes de *NLP* e sistemas de audioconferência, assim como tecnologias que se poderiam ter utilizado, ou se utilizaram, nesta dissertação. No Capítulo 3, apresenta-se a *Wit Bot Platform*, uma plataforma de desenvolvimento de *bots* criada pela empresa *Wit-Software* e, seguidamente, nos capítulos 4 e 5, expõem-se os dois sistemas desenvolvidos para responder aos objetivos propostos para esta dissertação: **O Assistente Virtual para gestão de audioconferências telefónicas** que, através de comunicação por voz permite agendar, listar e ligar a audioconferências telefónicas e o **Assistente Virtual de chamadas recebidas** um sistema que, tirando partido de um protótipo de uma aplicação Android e em conjunto com um bot, permite dar resposta imediata às chamadas recebidas pelos seus utilizadores. No Capítulo 6 são validados os resultados obtidos após a implementação dos dois sistemas e no Capítulo 7 é apresentado o fecho deste trabalho, isto é, se respondeu aos objetivos que foram propostos e o que pode ser feito futuramente.

Capítulo 2

Revisão Bibliográfica

Neste capítulo é feita uma revisão bibliográfica do que tem sido feito de importante nos últimos anos no domínio do processamento de linguagem natural, do reconhecimento de voz e das infraestruturas de comunicação. Também são exibidas ferramentas relevantes que foram ou poderiam ter sido utilizadas para o desenvolvimento deste projeto de dissertação. Por último, são apresentadas tecnologias existentes nos dias de hoje que permitem responder parcialmente aos objetivos desta dissertação.

2.1 Processamento de Linguagem Natural (*NLP*)

O interesse pelo processamento de linguagem natural (*NLP*) começou em 1950 quando Alan Turing publicou o seu artigo intitulado "*Computing Machinery and Intelligence*", do qual surgiu o chamado *Turing Test*. Turing afirmou basicamente que um computador poderia ser considerado inteligente se pudesse continuar uma conversa com um ser humano sem que o humano percebesse que estava a falar com uma máquina [KY15].

Segundo [Cho05], o processamento de linguagem natural é uma área de pesquisa e aplicação que explora como os computadores podem ser usados para entender e manipular texto em linguagem natural ou discurso humano para fazer coisas úteis. Os investigadores em *NLP* procuram reunir conhecimentos sobre como os seres humanos entendem e usam a linguagem, de modo a permitir o desenvolvimento de ferramentas e técnicas adequadas para que os sistemas computacionais compreendam e manipulem linguagem natural para executar as tarefas desejadas. Os fundamentos de *NLP* estão presentes em várias disciplinas, nomeadamente, a informática e as ciências da informação, a linguística, a matemática, a engenharia elétrica e eletrónica, a inteligência artificial, a robótica e a psicologia.

Mais recentemente, [CW14] sumariou que *NLP* pode ser definido como uma gama de técnicas computacionais para a análise automática e representação da linguagem humana.

As aplicações de *NLP* incluem um grande número de campos de estudo, tais como tradução automática, processamento de texto em linguagem natural e sumarização, interfaces com o utilizador, recuperação de informações multilinguísticas e em vários idiomas, reconhecimento de fala, inteligência artificial e sistemas periciais [Cho05].

2.1.1 Técnicas para processamento de texto

Em [CW14] são evidenciadas algumas limitações do que foi feito na área da *NLP* até então. Segundo os autores, a pesquisa em *NLP* evoluiu desde a era dos cartões perfurados e do processamento em lote (na qual a análise de uma frase podia demorar até 7 minutos) até à era do Google (em que milhões de páginas podem ser processadas em menos de um segundo). No entanto, quando se trata de interpretar frases e extrair informações significativas, as suas capacidades são reconhecidamente muito limitadas. O processamento de linguagem natural, de fato, requer capacidades simbólicas de alto nível [Dye94].

Várias técnicas e tarefas de *NLP* foram surgindo, umas mais populares que outras, sendo aqui descritas aquelas que tiveram mais importância para o desenvolvimento desta área.

Uma das estratégias de representação de *NLP* é a lógica de primeira ordem (*FOL*), um sistema dedutivo que consiste em axiomas e regras de inferência e que pode ser usado para formalizar predicados e quantificações relacionalmente ricos [Bar77]. A *FOL* suporta expressões sintáticas, semânticas e, até certo ponto, pragmáticas.

Segundo [SLC17] são necessárias várias etapas de pré-processamento para estruturar o texto e extrair recursos, como a *tokenization*, *word segmentation*, *Part of Speech (POS) tagging* e *parsing*. Seguidamente é fornecida uma breve visão geral dessas técnicas, assim como de outras duas técnicas: *Semantic Role Labeling(SRL)* e *Named Entity Recognition (NER)*.

Uma das tarefas exploradas numa vasta gama de aplicações *NLP* é o *part-of-speech (POS) tagging*. Esta tarefa consiste na classificação/rotulamento de palavras presentes numa frase de linguagem natural com *tags POS* [LLD⁺17]. Estas etiquetas correspondem a categorias como substantivos, verbos, advérbios, pronomes, conjunções e as suas subcategorias. Por exemplo, a palavra “assistente” teria associada a *POS tag* substantivo comum singular.

A *tokenization* é, também, uma técnica fundamental para a maioria das tarefas de *NLP*. Esta divide uma frase ou documento em *tokens* que podem ser palavras. Para o inglês, é trivial dividir palavras pelos espaços, mas algum conhecimento adicional deve ser levado em consideração. Na *tokenization*, algumas palavras, como os determinantes "o" e "a", são removidas, pois estas palavras fornecem pouca informação útil [Mad07]. Para chinês, japonês ou outros idiomas que não têm marcadores de divisão de palavras explícitos, a *tokenization* não é tão trivial quanto o inglês e é necessária uma *word segmentation*.

Na técnica de *parsing*, um *parser* constrói a árvore de análise para uma frase fornecida. Alguns *parsers* assumem a existência de um conjunto de regras gramaticais para executar a análise, mas os *parsers* recentes são inteligentes o suficiente para deduzir as árvores de análise diretamente a partir dos dados usando modelos estatísticos complexos [Mad07]. A maioria dos *parsers* também

trabalham num ambiente supervisionado e exigem que as palavras sejam classificadas com *tags POS* antes de se analisar a frase. A análise estatística é uma área de pesquisa ativa no *NLP*.

Por sua vez, a *Semantic Role Labeling (SRL)* procura dar um papel semântico a um constituinte sintático de uma frase. No formalismo *PropBank* [PGK05] atribui-se ARG0-5 às palavras que são argumentos de um predicado numa frase. Apresenta-se a seguinte frase como exemplo: "[O João] ARG0 [comeu] REL [a maçã] ARG1", onde "comeu" é o predicado. Os argumentos precisos dependem da estrutura de um verbo e se houver vários verbos numa frase, algumas palavras podem ter várias *tags*. Além das *tags* ARG0-5, há 13 *tags* de modificadores como ARGM-LOC (localização) e ARGM-TMP (tempo) que funcionam de forma semelhante para todos os verbos.

É também importante não esquecer a tarefa *Named Entity Recognition (NER)*. Esta fornece recursos-chave que ajudam em tarefas mais elaboradas de gestão de documentos e extração de informações [MDC07]. A tarefa do *NER* consiste em encontrar todos os substantivos próprios num texto e classificá-los entre várias categorias de interesse ou numa categoria padrão chamada *default* [MDC07]. Por exemplo, na frase "A Intel lançará um novo produto", a palavra Intel estará associada à categoria "Organização".

2.2 Ferramentas de *NLP*

O processamento de linguagem natural é fundamental para a construção de um *bot*, isto é, uma aplicação de software concebida para simular ações humanas. Como resultado, tem havido uma correlação direta entre a evolução de plataformas para desenvolvimento de *bots* e das plataformas para processamento de linguagem natural. Após uma análise de inúmeras plataformas existentes, foram selecionadas três. Seguidamente é apresentado o método de funcionamento de cada uma delas.

2.2.1 The Stanford CoreNLP Natural Language Processing Toolkit

Stanford CoreNLP [Cor17] é uma estrutura em cadeia de anotações Java (ou pelo menos baseada em *JVM*), que fornece a maior parte das técnicas de *NLP*, desde a *tokenization* e *part-of-speech (POS) tagger* até ao *named entity recognizer (NER)* [MSB⁺14]. O seu objetivo consiste em tornar fácil aplicar ferramentas de análise linguística a um texto, sendo altamente flexível e extensível. Com apenas duas linhas de código, o *Stanford CoreNLP* possibilita que uma ferramenta de *pipeline* corra sobre um pedaço de texto [Cor17]. Este *toolkit* fornece um conjunto de ferramentas de análise de linguagem natural. Permite reconhecer a maneira como as palavras são elaboradas, as suas posições numa frase, verificar se são nomes de empresas, pessoas, entre outros. Para além de tudo isto, possibilita normalizar datas, horas e quantidades numéricas, marcar a estrutura de frases em termos de dependências de palavras, indicar que substantivos correspondem às mesmas entidades e extrair relações particulares ou de classe aberta entre menções de entidade [Cor17]. Acrescente-se que as análises do *CoreNLP* fornecem os blocos de construção fundamentais para aplicações de compreensão de texto de alto nível e de domínio específico. No entanto, o *Stanford CoreNLP* é muitas vezes criticado por ser lento e usar uma grande quantidade de memória. Tal

facto deve-se sobretudo aos anotadores que são escolhidos. Na maior parte dos casos, as pessoas correm o *Stanford CoreNLP* apenas com os anotadores por defeito. Claro que os anotadores mais eficientes para um grande número de casos nem sempre são os que produzem os melhores resultados em questões de tempo e memória [sta16].

2.2.2 *Amazon Lex*

O *Amazon Lex* [ama17a] é um serviço para a criação de interfaces conversacionais com voz e texto. O reconhecimento de fala e o entendimento da linguagem natural são dois dos problemas mais difíceis de solucionar na ciência da computação, exigindo que algoritmos avançados de *deep learning* sejam treinados usando quantidades enormes de dados e infraestruturas. Através da análise do conteúdo detalhado em [ama17d], percebe-se que, além do reconhecimento automático de fala e entendimento da linguagem, através das falas que o utilizador introduz este serviço é também capaz de aprender as diferentes maneiras com que uma pessoa pode exprimir as suas intenções.

Para perceber como funciona o serviço *Amazon Lex* é importante conhecer o significado de conceitos como *Bot*, *Intent*, *Slot* e *Slot type* [How17]. No âmbito do *Amazon Lex* o *Bot* é responsável por executar tarefas de forma automática, como agendar uma reunião ou encomendar uma pizza. Para tal, este *Bot* usa recursos de reconhecimento automático de fala (*ASR*) e de entendimento de linguagem natural (*NLU*) [How17]. Uma intenção, *Intent*, representa uma ação que se pretende realizar e deve ter um nome descritivo. A uma intenção estão associadas frases de ativação, isto é, frases que o utilizador deve dizer, não necessariamente palavra a palavra, para invocar essa intenção. Um exemplo de uma dessas frases poderia ser a frase "Posso agendar uma reunião para, por favor". Por sua vez, os *Slots* representam os parâmetros de uma intenção. Seguidamente, em tempo de execução, o *Amazon Lex* solicita ao utilizador valores específicos para estes *Slots*. Por último, os *Slot types* são os tipos de *slot*, podendo ser personalizados ou incorporados. O *Amazon Lex* contém *Slot types* internos, por exemplo, a data, a hora ou certas localizações. No caso do agendamento de uma reunião os *Slot types* poderiam ser o local da reunião, a data e tipo de reunião, sendo que este último seria um tipo de *slot* personalizado.

Para criar um *bot Amazon Lex* com mínima informação requerida inicialmente é necessário atribuir um nome à nossa aplicação, selecionar uma voz de *output* e definir um tempo de duração máxima de sessão [Lex17]. De seguida criam-se os *Slot types* (data da reunião, tipo de reunião, local). A cada *Slot type* são associados valores possíveis (por exemplo, Coimbra e Porto para o *Slot type* local). Criam-se também as *Intents* escolhendo um nome sugestivo ("Agendar Reunião"). Para esta *Intent* são adicionados os *inputs* esperados para ativar a *Intent*. Estes *inputs* podem conter *slots*, representados entre chavetas no seguinte exemplo: "Quero agendar uma reunião {data} nos escritórios do {local}." Os *slots* também devem ser adicionados posteriormente, e estão associados a *Slot type* já criados. Realizados todos estes passos, é necessário testar a aplicação e está concluída uma simples aplicação *bot*.

Refira-se que esta ferramenta permite uma fácil integração com o serviço *AWS Lambda*. O *AWS Lambda* permite a execução de código sem a necessidade de gerir servidores. É assim possível executar código para praticamente qualquer tipo de aplicativo ou serviço de *back-end*, tudo sem precisar de administração. Basta carregar o código e o *Lambda* toma conta de tudo o que for necessário para executar e escalar o código com alta disponibilidade [amal17c].

2.2.3 API.AI

API.AI é uma plataforma de compreensão de linguagem natural que permite projetar e integrar interfaces de conversação inteligentes e sofisticadas em aplicações móveis, Web, dispositivos e bots [wel17]. Nesta plataforma as aplicações são organizadas em torno de *agentes*. Já existem diversas intenções pré treinadas, que abrangem tarefas comuns, como autenticação, compras, reserva, etc.

Tal como o *Amazon Lex*, a *API.AI* é construída sobre uma série de conceitos: *Agents*, *Entities*, *Intents*, *Actions* e *Contexts*. Os *Agents* podem ser definidos como módulos de *NLP* para aplicações. As *Entities* representam conceitos como uma forma de mapear frases em linguagem natural para frases canónicas que capturam o seu significado. *Intents* correspondem ao mapeamento que é feito entre o que o utilizador diz e a acção que o software deve executar. *Actions* representam as etapas que a aplicação executará quando certos *Intents* são acionados pelas entradas do utilizador. Por último, *Contexts* são sequências de caracteres que representam o contexto atual da expressão do utilizador. Os *Contexts* são de uma importância extrema uma vez que permitem diferenciar frases que podem ser vagas e ter significados diferentes em diferentes contextos.

Para construir uma aplicação usando a *API.AI*, primeiro começa-se por criar um agente (*Agent*), que representará a interface comunicacional da aplicação ou *bot*. A esse agente deve ser atribuído um nome, por exemplo "AssistentePessoal", e uma linguagem. Criado o agente deverão ser-lhe atribuídas algumas *Entities*. As *Entities* são muitas vezes específicas do domínio em questão, por exemplo, para o assistente pessoal que permite agendar reuniões uma *Entity* poderia ser o tipo de reunião, que poderia tomar o valores de reunião de administração ou reunião de equipa. Seguidamente são elaborados *Intents*, ou seja, as frases que podem ser ditas para executar certa ação ("Gostaria de agendar uma reunião de equipa"). Por fim, o agente deve ser treinado, para ver se entende o que se pretende dele. Se as frases ditas pelos utilizadores não forem percebidas, mas o conteúdo dito corresponda a outra forma de invocar uma dada intenção, adiciona-se essa frase de imediato às *Intents*.

2.2.4 Comparação de ferramentas

Após a análise do funcionamento das três ferramentas anteriores, será feita agora uma análise crítica com as mais e menos valias de cada uma das ferramentas. Nesse sentido, apresenta-se a Tabela 2.1 que sintetiza os prós e contras de cada ferramenta.

Tabela 2.1: Comparação de ferramentas [Lex17, Rod17, try17]

Ferramenta	Prós	Contras
<i>Stanford CoreNLP</i>	Baseada em <i>JVM</i> ; Possibilita uma grande customização ; É um serviço gratuito.	Ausência de interface de voz; Sem integração com plataformas de chat.
<i>Amazon Lex</i>	Interface de voz; Integração com o serviço <i>AWS Lambda</i> ; Grande comunidade de programadores e parceiros por parte da <i>AWS</i> .	Pouca integração com plataformas de chat; O serviço é pago consoante o número de pedidos; Última das plataformas a entrar no mercado.
<i>API.AI</i>	Interface de voz; Forma poderosa de modelar fluxos grandes e complexos usando <i>Intents</i> e <i>Contexts</i> ; Uma boa parte da lógica pode ser resolvida pelo <i>chatbot</i> , o que diminui a codificação do lado do servidor; Integração com um clique em várias plataformas de chat.	É impossível bloquear a correspondência de uma <i>intent</i> se um <i>context</i> estiver presente; A fase de treino ainda está em versão beta.

Após a análise da Tabela 2.1, percebe-se que para este projeto de dissertação as duas ferramentas mais poderosas são o *Amazon Lex* e o *API.AI*. Isto porque fornecem, desde já, mecanismos de *speech recognition* e de *text-to-voice*. O *Stanford CoreNLP*, por sua vez, permite uma grande personalização, por outras palavras, adaptar o processamento dos conteúdos aos *inputs* esperados e ao tema em questão. Para além disso, o *Stanford CoreNLP* é uma ferramenta *open source*. O *API.AI* tem como mais-valia a sua fácil integração com várias plataformas de chat, enquanto o que o *Amazon Lex* confia na *AWS Lambda serverless computing stack* para implementar “*fulfillments*”, que são ações que precisam de ser acionadas com base numa intenção específica. Atualmente, a *AWS Lambda* é a mais usada e tecnologicamente mais avançada plataforma de computação sem servidor no mercado, o que traz algumas vantagens indiretas para o serviço *Amazon Lex* [Rod17].

2.3 Tecnologias e protocolos de comunicação

Hoje em dia existe uma forma de executar chamadas de voz, nomeadamente através do uso de tecnologias que transmitem dados usando *IP*. O *VoIP* é essa nova forma de comunicação. É uma tecnologia que permite a realização de chamadas telefónicas através de uma rede *IP*. Por outro lado, antes de tudo é necessário estabelecer essa chamada telefónica; o *SIP* é o protocolo que permite a

realização dessa tarefa e outras necessárias no processo de comunicação [SC14]. Nas subsecções seguintes será apresentada a tecnologia *VoIP*, o protocolo *SIP*, assim como o sistema *ivr*, um sistema capaz de responder ou interagir com o interlocutor de uma chamada através de áudio. Este áudio pode ser pré-gravado ou dinâmico. Acrescente-se que também se apresenta de seguida a plataforma Twilio, uma plataforma *cloud* de comunicações que foi utilizada no desenvolvimento deste projeto de dissertação.

2.3.1 *VoIP*

Atualmente, uma das tecnologias mais dominantes da comunicação é a tecnologia *VoIP* (*Voice Over Internet Protocol*). Esta tecnologia permite a realização de telefonemas através da Internet, enviando pacotes através da rede de comutação por pacotes [JV15]. Assim sendo, a tecnologia *VoIP*, ao contrário da *PSTN* (*Public Switched Telephone Network*), não necessita de uma linha dedicada para o serviço de telecomunicação. Na Tabela 2.2 é feita uma comparação entre as características principais destas duas tecnologias.

Tabela 2.2: Diferenças entre *VoIP* e *PSTN*

<i>VoIP</i>	<i>PSTN</i>
<p>Todos os canais usam uma conexão de internet;</p> <p>A compressão pode resultar em 10Kbps;</p> <p>Recursos como chamada em espera e identificador de chamadas são incluídas como serviço;</p> <p>Longas distâncias são geralmente incluídas nos preços mensais;</p> <p>As atualizações normalmente só requerem uma atualização de software.</p>	<p>Linhas dedicadas;</p> <p>Cada linha tem uma largura de 64Kbps para cada lado;</p> <p>As funcionalidades de chamada em espera são pagas à parte;</p> <p>Para ser atualizado é necessário mudar o equipamento e a linha de comunicação;</p> <p>Longas distância são normalmente pagas ao minuto.</p>

Atualmente o *VoIP* junta o melhor dos dois mundos. Permite falar com qualidade, fiabilidade e estabilidade, através do uso da Internet, e a um preço bastante mais baixo que as comunicações de voz tradicionais, sejam elas através de telefone fixo ou telemóvel.

Existem cada vez mais pacotes de *software* que possuem esta tecnologia, como por exemplo o *Skype* e o *WhatsApp*.

O *WhatsApp* é um software que permite enviar e receber todo o tipo de ficheiros: fotografias, vídeos, documentos, localizações e também estabelecer chamadas de voz. Como as chamadas de voz pelo *WhatsApp* utilizam o protocolo *VoIP*, basta ter uma ligação à Internet para se comunicar. Uma outra funcionalidade importante deste software é a sua capacidade de enviar facilmente mensagens de voz para qualquer contacto que também possua o software [wha17].

Por sua vez, O *Skype* também oferece o serviço *VoIP*, permitindo aos seus utilizadores comunicarem bidirecionalmente entre si, oferecendo ainda suporte de conferências. O *Skype* é o primeiro cliente *VoIP* baseado numa arquitetura *peer-to-peer*. Na arquitetura redes de computadores *peer-to-peer* cada um dos pontos ou nós da rede funciona quer como cliente, quer como servidor, permitindo a partilha de serviços e dados sem a necessidade de um servidor central. [BS04].

Ambos os programas se apresentam como uma alternativa às chamadas de voz tradicionais.

2.3.2 Protocolo de iniciação de sessão (SIP)

O *SIP* (*Session Initiation Protocol*) é um protocolo de camada de aplicação que permite criar, modificar e encerrar sessões com um ou mais participantes. Estas sessões podem ser chamadas telefónicas pela Internet ou conferências multimédia [RSC⁺02]. O *SIP* utiliza elementos chamados servidores *proxy* para ajudar a encaminhar solicitações para o atual local do utilizador, autenticar e autorizar utilizadores para o serviço, implementar políticas de roteamento de chamadas de provedor e fornecer recursos aos utilizadores. O *SIP* também fornece uma função de registo que permite aos utilizadores enviar o seu local atual para ser usado por servidores *proxy*. O uso do *SIP* para estabelecer sessões de voz, vídeo e dados coloca a telefonia como apenas outra aplicação na Internet, usando endereçamento, tipos de dados, software, protocolos e segurança semelhantes. Assim sendo, a utilização de redes separadas para voz não é mais necessária. A integração completa de voz com todos os outros serviços e aplicações da Internet provavelmente forneceu a maior oportunidade para inovação [SJ12]. A Figura 2.1 mostra os diferentes elementos de uma rede de comunicação *SIP*.

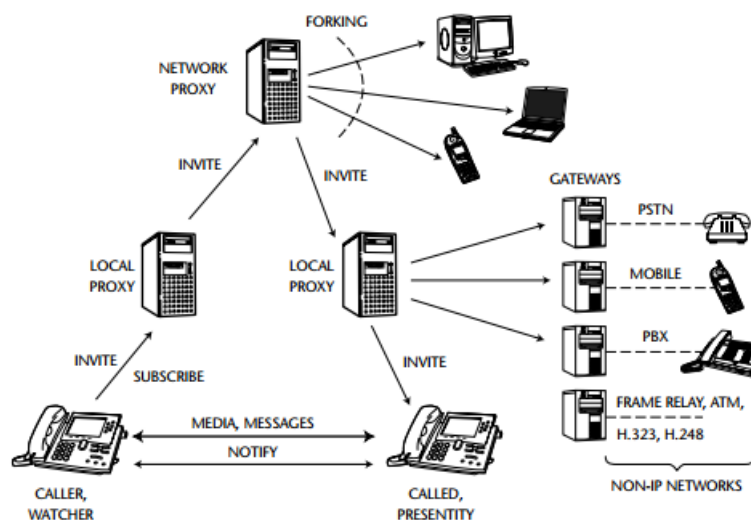


Figura 2.1: Rede de comunicação *SIP* [SJ08]

A semelhança entre *SIP* e *HTTP* facilita a criação de serviços por uma comunidade muito grande de desenvolvedores de software que já estão familiarizados com o desenvolvimento de

websites. O SIP pode ser interpretado como o protocolo de autenticação para serviços de VoIP [JV15]. Hoje em dia já existem ferramentas que funcionam como plataforma *cloud* de comunicações. Uma dessas plataformas é a plataforma Twilio que possibilita aos programadores executar e receber chamadas telefônicas usando *web service APIs*. A Twilio possibilita, por exemplo, a construção de sistemas IVR usando interfaces SIP [twi17a].

2.3.3 Sistema Interativo de Resposta por Voz (IVR)

Um sistema IVR é uma tecnologia que permite que um computador consiga interagir com os seres humanos através do uso de tons de voz e entradas via teclado. Os sistemas de voz interativa podem ser implementados com reconhecimento de voz e *prompts* de voz gerados usando *VoiceXML*. Apresenta-se na Figura 2.2 um exemplo de fluxo de chamadas para um servidor IVR [SJ08].

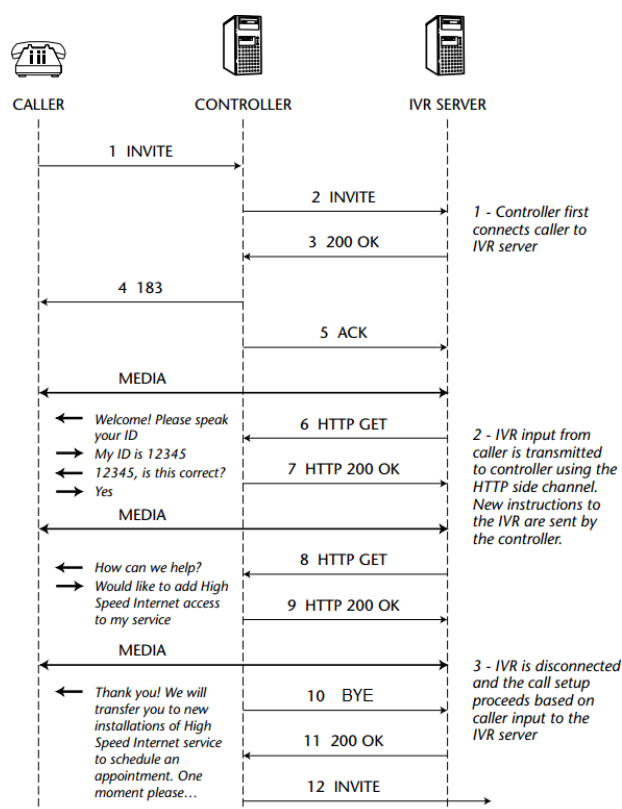


Figura 2.2: Fluxo de chamadas de sistema IVR [SJ08](adaptado)

O serviço é inicializado com uma troca IVR para determinar o objetivo do chamador. A mensagem *INVITE* inicial do chamador tem o *URI* de solicitação a apontar para o controlador e para este serviço específico. De seguida o IVR irá gerar um *prompt* de voz para o chamador, ao longo da linha, com a frase "Bem-vindo ao nosso serviço <nome>! Por favor, indique o seu ID ". A resposta do chamador é transformada de fala para texto e retornada na mensagem 6 da figura, *HTTP GET*, para o controlador. O próximo *script VoiceXML* é enviado pelo controlador no *HTTP 200 OK*, que corresponde à mensagem 7 da figura, para solicitar mais informações ao chamador

sobre sua solicitação. Depois do processo *IVR* chegar ao fim, a última mensagem *HTTP 200 OK*, mensagem 9, carrega um *script VoiceXML* vazio. A chamada para o *IVR* é terminada com um *BYE*, mensagem 10, e a chamada é encaminhada para outro destino com o *INVITE* na mensagem 12. Todo este processo foi baseado em [SJ08].

Os sistemas interativos de resposta por voz baseados em *VoiceXML* podem suportar diversos recursos e serviços de voz tais como, conversão de texto em voz, reconhecimento de discurso e gravação de discursos introduzidos pelos utilizadores.

2.4 Sistemas de audioconferência

Seria expetável que as pessoas do século passado não fizessem ideia do que é uma audioconferência. Mas, na verdade, a primeira audioconferência foi realizada a 25 de Janeiro de 1915 [Law15]. Relatórios da altura indicam que demorou cerca de dez minutos para ligar a chamada, uma vez que foi preciso estabelecer conexões manuais em cada cidade ao longo da rota da chamada. Hoje em dia, com os avanços da tecnologia de audioconferência, a qualidades das chamadas melhorou drasticamente, assim como o processo de as efetuar. Seguidamente descrevem-se algumas das tecnologias que permitem a realização de audioconferências mais usadas atualmente.

2.4.1 WebEx

A *WebEx* fornece sistemas colaborativos de reuniões online, audioconferências *web* e videoconferências. A *WebEx* possui uma plataforma *online* que permite o agendamento antecipado de conferências ou até mesmo a criação de um endereço *web* fixo para que, a qualquer momento, uma pessoa se ligue a esse endereço e entre em conferência com quem se encontra *online* nesse mesmo endereço. Uma das grandes funcionalidades deste sistema é o facto de, embora seja maioritariamente um serviço de *VoIP*, possibilitar a ligação a um número da rede *PSTN*. Para se agendar uma reunião através deste serviço o utilizador deve inicialmente validar os seus dados na plataforma *web*, seguidamente seleccionar a opção "agendar um reunião rápida", seleccionar o tipo de reunião, introduzir a data, o horário e a duração da mesma e por fim introduzir o endereço de correio eletrónico dos convidados. Seguidamente, os convidados apenas têm de aceder ao seu endereço de correio eletrónico e entrar na conferência introduzindo os seus dados pessoais[int17].

2.4.2 Conference Calling by Vastconference

Esta plataforma permite a criação rápida audioconferências, bastando para isso aceder ao painel administrativo da página *web* da plataforma, e adicionar possíveis convidados. Cada convidado tem associado um número de ligação único. Por outro lado, e embora seja este um serviço *VoIP*, ele permite a ligação a chamadas para a rede *PSTN*. Além disso, esta plataforma permite a gestão de audioconferências através dos dígitos do telephone em ligação. Por outras palavras, o administrador da audioconferência pode, por exemplo, silenciar todos os utilizadores em linha primindo o dígito 1 seguido de "*", ou, bloquear a chamada primindo 2 seguido de "*" [tel17].

2.4.3 Infinite Conferencing

A plataforma *Infinite Conferencing* oferece uma vasta gama de serviços áudio, *web* e *streaming*. Este serviço, por defeito, opta por distinguir conferências *web* de audioconferências telefónicas. Além disso, baseia-se na utilização de códigos para entrada na conferência e PINs posteriores de acesso à mesma [Con17].

2.5 Conclusões

Embora a pesquisa em *NLP* tenha feito grandes avanços na produção de comportamentos artificialmente inteligentes, como o Google, o Watson da IBM e a Siri da Apple, estas *frameworks/APIs NLP* ainda não são perfeitas [CH12].

Ainda assim, a maioria das *APIs* são boas para começar rapidamente a criar pequenas aplicações conversacionais. Mas nem tudo são vantagens. Ao usar uma *API* externa haverá sempre uma dependência nas decisões de outrem. Isto pode ser uma limitação à medida que o projeto se desenvolve. Existem vários problemas que podem prejudicar a experiência do utilizador. Entre estes problemas destaca-se a falta de contexto, isto é, muitas vezes o significado de uma frase depende do seu contexto e nem sempre este é identificado por estas ferramentas. Outra limitação relaciona-se com a capacidade de gestão de falhas, que nem sempre é fornecida por estas *API*, quebrando assim a experiência positiva do utilizador. Além disso, o envio de informações sensíveis para uma terceira parte deixa os utilizadores desconfortáveis.

Por outro lado, o objetivo desta dissertação consiste na construção de provas de conceito, pelo que o uso de uma *API* é imperial. As *API* de linguagem natural fornecem uma boa maneira de começar com interações de pergunta e resposta. Uma das vantagens do uso destas ferramentas prende-se com a sua capacidade de construir plataformas de conversação de forma rápida e simples. Das *API* analisadas, a *API.AI* e o *Amazon Lex* são as mais apropriadas para o desenvolvimento deste projeto pois já oferecem tecnologias de conversão de voz em texto e capacidades de fala. O *Amazon Lex* apresenta-se como o serviço de reconhecimento de fala e compreensão de linguagem natural com maior potencialidade. Tal deve-se à sua simplicidade entre as ferramentas analisadas e a sua integração com o *AWS Lambda*. Acrescente-se que este serviço já está exposto a mais de 3 milhões de utilizadores, através do *Amazon Alexa* [Bor16], o que faz evoluir a sua aptidão de dia para dia através de técnicas de *machine learning*. Uma das vantagens da *API.AI* é a sua fácil integração com plataformas de chat, mas para este projeto de dissertação tal não é necessário.

Já existem hoje softwares alternativos, que comunicam através da utilização da tecnologia *VoIP*. Contudo, pretende-se desenvolver uma solução para o caso das chamadas telefónicas que usam a *PSTN*, o que leva à inviabilidade destas alternativas. No mesmo sentido, existem plataformas, como a plataforma *Twilio*, que possibilita aos programadores executar e receber chamadas telefónicas usando *web service API*. No caso do assistente virtual para gerir audioconferências telefónicas, as plataformas existentes já se mostraram eficazes e permitem inclusive a realização de

Revisão Bibliográfica

videoconferências. Contudo, estas plataformas ainda não permitem uma gestão das conferências através de controlos de voz e muitas empresas utilizam sistemas de audioconferências telefónicas devido à sua elevada qualidade de serviço.

Capítulo 3

Wit Bot Platform

Como foi referido no Capítulo 1 deste documento, esta dissertação está a ser desenvolvida no seguimento de uma proposta da empresa *Wit-Software*. Na verdade, os sistemas que estão a ser desenvolvidos foram propostos por esta mesma empresa, sendo que todo o seu desenvolvimento teve um acompanhamento constante. Para além disso, ambos os sistemas desenvolvidos estão integrados numa plataforma própria da *Wit-Software*, a *Wit Bot Platform*. Esta plataforma tem como objetivo simplificar a maneira como se constroem *bots*, nomeadamente nas suas integrações com *APIs* externas, plataformas *OTT (Over-the-top)*, serviços de processamento de linguagem natural e gestão de fluxos conversacionais. A Figura 3.1 representa uma visão generalizada da arquitetura desta plataforma, dividida em quatro módulos principais: *Channel Gateway*, *Bot Engine*, *Bot Builder* e *Integration*. No desenvolvimento deste projeto de dissertação não foi utilizado o módulo *Bot Builder*.

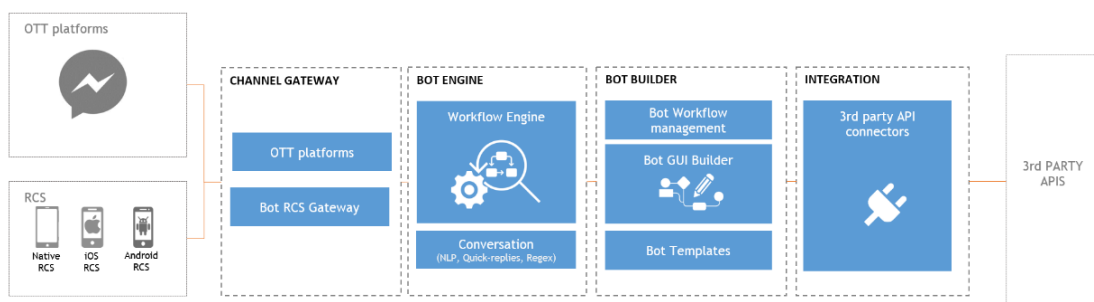


Figura 3.1: Arquitetura da *Wit Bot Platform*

3.1 *Channel Gateway*

Neste módulo evidencia-se a capacidade que a *Wit Bot Platform* tem de se integrar com diferentes canais de comunicação, como é o caso do *Facebook Messenger*, *Skype* e do *Slack*, entre

outros canais que estão constantemente a ser adicionados a esta plataforma. Além destes canais, esta plataforma já tem também canais de comunicação utilizando protocolos *RCS*, como por exemplo o serviço *Message+* da Vodafone. Isto permite construir *bots*/assistentes virtuais que posteriormente poderão ser publicados em diferentes plataformas, simples e eficazmente.

3.2 Bot Engine

É neste módulo que toda a plataforma é gerida, sendo o maior e principal módulo da mesma. Por sua vez este módulo encontra-se dividido em sub módulos mais pequenos que reunidos permitem a criação de um *bot* com capacidades ímpares.

3.2.1 NLP Engine

Esta plataforma tem integrações com várias ferramentas de processamento de linguagem natural. Existem, entre outros, canais de comunicação com a *WIT.AI* ou o *Amazon Lex*. Esta integração permite tornar a plataforma super poderosa pois agrupa mais do um serviço de processamento de linguagem natural. São estes serviços que, como já foram mencionados no Capítulo 2, possibilitam a análise do texto que é falado ou escrito pelos utilizadores, fazendo com que se extraia o conteúdo presente numa frase. Cada serviço de *NLP* procura detetar a intenção da fala do utilizador. Se o sistema conseguir detetar a intenção, seguidamente inicia um fluxo conversacional que foi criado para começar quando essa intenção é invocada.

3.2.2 AIML Engine

Quando o sistema não consegue perceber o que foi dito pelo utilizador através dos serviços de processamento de linguagem natural, isto é, não encontrou a intenção da frase dita pelo utilizador, este irá procurar num conjunto de ficheiros *AIML* se existe alguma resposta prevista para aquela frase. Este mecanismo não só acrescenta vocabulário e “inteligência” ao *bot*, como ainda permite a construção de *bots* de pergunta e resposta rápida, de uma forma simples e fácil. No mesmo sentido, a *Wit Bot Platform* já possui um conjunto enorme de padrões de frases do utilizador e respostas possíveis do *bot*. Estas respostas abrangem vários domínios e oferecem capacidades aos *bots* que surpreendem o utilizador. Por exemplo, imagine-se que se está a construir um simples *bot* de reserva de salas. À priori, apenas serão criadas intenções para apurar que é de facto uma reserva de uma sala e, seguidamente, detetar o número da sala e data da reserva. O próprio utilizador não esperará muito mais daquele *bot* que possibilita unicamente reservar salas. Mas, imagine-se agora, que o utilizador decide perguntar ao *bot* se este gosta de cerveja. Um contexto em nada relacionado com um sistema de reserva de salas. Assim, os sistemas de processamento de linguagem natural não detetarão a intenção do utilizador, pois não estavam preparados para tal, mas, ao procurar no conjunto de ficheiros *AIML* de possíveis padrões de perguntas e respostas, o *bot* encontrará uma resposta à questão do utilizador. Deste modo, percebe-se o poder extra de construir um *bot* associado a esta plataforma. Acrescente-se que estes ficheiros também incorporam frases que

podem ativar intenções, isto é, ao invés de o *bot* dar uma simples resposta, irá começar um fluxo conversacional previamente criado.

3.2.3 BPMN Engine

Este sub módulo representa os fluxos conversacionais que os *bots* podem possuir. Os fluxos são criados através da utilização do *Bizagi Modeler*. O *Bizagi Modeler* é um *software* de modelação de processos de negócio que permite e facilita a criação de diagramas de fluxo e *workflows* [cez17]. Por sua vez, este software permite e incorporação de parâmetros e propriedades específicas para a construção de fluxos conversacionais para a *Wit Bot Platform*. Expõe-se na Figura 3.2 um exemplo de um fluxo conversacional construído através do *Bizagi Modeler*.

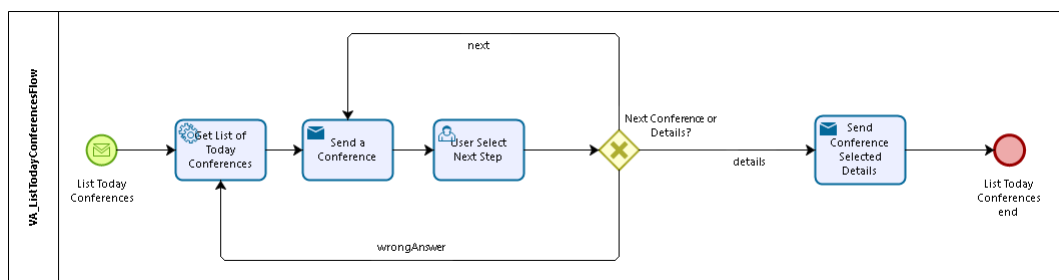


Figura 3.2: Exemplo de fluxo conversacional implementado com o *Bizagi Modeler*

Start Event

É neste nó que se define o início do fluxo conversacional. Por exemplo, quando através do serviço de processamento de linguagem natural o bot percebe a intenção da frase que lhe foi enviada, ele verifica qual dos nós iniciais corresponde a essa intenção e seguidamente inicia o fluxo associado.

Service Task

É neste nó onde se encontra a lógica do fluxo. Na realidade, é neste tipo tarefa que são processadas informações e se comunica com APIs externas ou sistemas de base de dados.

User Task

Esta tarefa é responsável por receber *inputs* por parte do utilizador.

Exclusive Gateways

Neste nó são tomadas diferentes direções no fluxo com base em informações anteriores. Esta informação pode ser, por exemplo, um *input* do utilizador ou resultados do processamento de uma *service task*.

Message task

Esta tarefa permite enviar informação ao utilizador. Esta informação pode, por sua vez, ser configurada com base no serviço *OTT* a ser utilizado.

End event

É neste nó que se define o término do fluxo.

3.3 *Integration*

A *Wit Bot Platform* já tem incorporadas integrações com serviços externos e, através deste módulo, a incorporação de outro qualquer serviço é trivial. Neste momento existem integrações com, por exemplo, a Broadsoft UC, o Google Calendar, o Asterisk ou a PIPL. Além disso, esta plataforma já tem incorporadas bases de dados para gerir a informação.

3.4 *Conclusões*

Neste capítulo ficaram patentes as diversas vantagens do desenvolvimento de *bots* integrados na *Wit Bot Platform*. Para além dos diversos canais de comunicação que esta plataforma já tem integrados, ela funciona como um cérebro digital, percebendo muito do que é dito pelos utilizadores e respondendo com base em mais de dez mil enunciados possíveis de que dispõe. Isto permite oferecer aos *bots* desenvolvidos, para além das capacidades principais e específicas do *bot*, um perfil, emoções e até reações humorísticas. Além disso, integra vários serviços de processamento de linguagem natural, permitindo uma elevada taxa de confiança na interpretação dos enunciados do utilizador, assim como facilita a manutenção de contexto conversacional através dos fluxos que são criados.

Capítulo 4

Assistente Virtual para gestão de conferências telefônicas

Neste capítulo expõem-se os fatores que levaram à elaboração do assistente virtual para gestão de audioconferências telefônicas. São demonstradas as limitações que existem atualmente no processo comum de agendamento de audioconferências telefônicas, assim como, a solução desenvolvida. No mesmo sentido é apresentada a arquitetura do protótipo desenvolvido, as ferramentas utilizadas e os casos de uso funcionais. Por outro lado, e como este produto, embora tenha uma componente singular, foi desenvolvido partindo de alguns componentes já desenvolvidas na *Wit-Software*, é demonstrado o estado do sistema inicial e o que foi criado de novo.

4.1 Inovação empresarial e surgimento de novas tecnologias

Atualmente, muitas das médias e grandes empresas têm escritórios em diferentes locais do globo. Na verdade, na maioria das vezes, a sede da empresa fica num local diferente dos escritórios de produção. Acrescente-se que, as empresas, cada vez mais, optam pela terceirização dos seus serviços, para além de permitirem aos seus funcionários trabalharem remotamente a partir de suas casas. A realização de conferências tornou-se assim numa prática quase diária [Wat17a]. Será então possível melhorar o processo de gestão de audioconferências telefônicas?

Como foi demonstrado anteriormente, já existem tecnologias capazes de realizar conferências através de telefonia IP. Contudo, no que diz respeito à rede pública de telefonia (PSTN) global, o processo de ligação e agendamento continua a ser obsoleto. A maioria dos sistemas funcionam através da utilização de números e palavras passe de ligação.

Por outro lado, começam a surgir, cada vez mais frequentemente, sistemas que respondem a comandos de voz, reproduzem música ou, por exemplo, conseguem através da voz controlar diferentes dispositivos. Dois desses sistemas são o *Amazon Alexa*, controlada através de dispositivos *Amazon* compatíveis, como o *Amazon Echo*, *Echo Dot* ou, mais recentemente, o *Echo Show*, e o

Google Home. O *Amazon Echo* e *Echo Dot* são colunas *wireless* que tiram partido das capacidades do serviço de voz, *Amazon Alexa*, e possibilitam ao seus utilizadores ativar alarmes, tocar músicas, ler notícias, controlar luzes, realizar compras *online*, entre outras tarefas. Mas mais importante, tudo isto é feito através de comandos de voz. As respostas devolvidas pelos dispositivos são maioritariamente respostas de áudio, sendo que também é possível acrescentar conteúdo visual através de uma aplicação própria, *Amazon Alexa App*. O *Amazon Echo Show* tem como principal mais-valia o facto de possuir um pequeno ecrã que permite, desde logo, apresentar conteúdo audiovisual, e ainda tem a capacidade de *touchscreen* [Vin17]. Acrescente-se que a *Alexa* possui o *Alexa Skills Kit*, uma coleção de *APIs*, ferramentas e documentação que permite desenvolver *bots* que acrescentem capacidades de voz às já existentes [ale17]. Por sua vez, o *Google Home* contempla a maioria das capacidades apresentadas para o *Amazon Echo*. É também ele uma coluna com capacidade de interpretar e responder a comandos de voz e partilha das funcionalidades e capacidades do *Google Virtual Assistant*. Contudo, o seu maior potencial é a sua habilidade de responder a questões de forma mais eficiente que os seus concorrentes diretos [Gib17].

4.2 Sistema de agendamento de audioconferências telefônicas através de um assistente virtual controlado por voz

Atendendo ao que foi referido, em conjunto com a empresa *Wit-Software*, pensou-se numa forma de melhorar o processo de gestão de audioconferências. Isto porque, também esta empresa, tem escritórios em diferentes locais do globo e, simultaneamente, trabalha como prestadora de serviços para outras empresas. Desenvolveu-se um assistente virtual que, através de comunicação por voz permite agendar, listar e ligar a audioconferências telefônicas, entre outras funcionalidades. Este assistente comunica com um servidor de audioconferências telefônicas e permite aos utilizadores deixarem de lado o processo de inserção de códigos e de utilização de aplicações manuais para reunirem com a sua equipa.

O que foi dito na secção anterior permite admitir a utilização de uma das ferramentas apresentadas para ajudar a dar resposta à questão do melhoramento dos atuais sistemas de gestão de audioconferências. Assim sendo, e tirando partido da *Amazon Alexa*, em conjunto com o dispositivo *Echo Dot*, desenvolveu-se um sistema de agendamento, listagem e ligação de audioconferências telefônicas. Refira-se que à data de início de desenvolvimento desta dissertação, ainda não havia sido lançado o dispositivo *Echo Show*, que poderia ter sido uma enorme ajuda para tornar a aplicação desenvolvida mais eficaz. O sistema desenvolvido tirou partido de um *bot*/assistente virtual, já desenvolvido dentro da empresa *Wit-Software*. O *bot* existente procurava melhorar o serviço de agendamento de audioconferências telefônicas utilizando o *Facebook Messenger*. Manteve-se a compatibilidade com esse mesmo *bot* procurando, contudo, adicionar as capacidades de voz às funcionalidades já existentes, assim como desenvolver novas funcionalidades. Criou-se uma *Alexa Skill* que, através de comunicação com o assistente virtual/*bot*, permite gerir as audioconferências. Na verdade, o utilizador pode solicitar à *Amazon Alexa*, tirando partido do *Echo Dot*, para, por

exemplo, agendar uma audioconferência. Seguidamente o enunciado é analisado pelas ferramentas de compreensão de fala da *Amazon Alexa*, e segue-se um processo de obtenção de informação necessárias ao agendamento da audioconferência. No caso de uso referido, por exemplo, é pedido ao utilizador para indicar a data, o título e convidados.

4.3 Implementação

No sentido de se desenvolver a solução referida anteriormente, foi desenvolvido um servidor que comunica com diferentes *APIs*, assim como está integrado com a *Wit Bot Platform*, explicada no Capítulo 3. Nesta secção apresenta-se o protótipo desenvolvido, incluindo a sua arquitetura, as ferramentas utilizadas e os casos de uso funcionais.

4.3.1 Ferramentas e tecnologias utilizadas

Para o desenvolvimento deste sistema foram usadas diversas tecnologias que surgiram recentemente. As ferramentas mais importantes para o desenvolvimento deste produto, e sem elas não teria sido possível realizar este sistema, foram a coluna *Echo Dot* e o serviço *Amazon Alexa*. Este dispositivo contou com um aumento significativo no número de vendas no mês de Dezembro de 2016, mais concretamente onze milhões em número de unidades, motivo por se optar pela escolha deste dispositivo em detrimento de outros, como o referido *Google Home* [Gon17]. Outra das ferramentas utilizadas foi a plataforma *Asterisk*, uma plataforma *open-source* para construção de aplicações de comunicações. Através do uso do *Asterisk* consegue-se transformar um simples computador num servidor de comunicações [ast17]. Refira-se que o servidor *Asterisk* utilizado já havido sido desenvolvido e já possuía uma *API* desenvolvida. O servidor foi desenvolvido em *NodeJS*, que permite a criação de servidores *web* utilizando *JavaScript* e uma coleção de módulos com capacidades diversas que estão em constante aumento. Como o sistema está inserido na *Wit Bot Platform*, e esta plataforma utiliza *MongoDB* para guardar informação, manteve-se esta base de dados que usa um modelo orientado a documentos.

4.3.2 Arquitetura

Esta secção apresenta uma vista de alto nível da arquitetura do sistema desenvolvido. Esta vista possibilita perceber como as principais ferramentas e tecnologias usadas se relacionam entre si. Assim, apresenta-se na Figura 4.1 a vista de alto nível da arquitetura do sistema. Esta arquitetura será descrita mais aprofundadamente nas próximas secções do documento.

Assistente Virtual para gestão de conferências telefónicas



Figura 4.1: Arquitetura global do sistema

4.3.3 Amazon Alexa

Para o desenvolvimento do sistema, o primeiro passo passou pela criação de uma *Amazon Alexa Skill*, que é capaz de receber informação enviada por voz, através do *Amazon Echo Dot*, e perceber que intenção está associada à fala do utilizador. Para isso, é necessário criar um modelo de interação, com uma lista de todas as intenções que se pretendem ter associadas à *Amazon Alexa Skill*. Cada intenção terá associada uma ou várias frases/enunciados de ativação e pode ter, ou não, associados *slots*. Os *slots* são como variáveis que têm a eles um tipo associado, *slot type*. Por exemplo, supondo que se pretende criar uma intenção para criar um lembrete para um dia semanal à escolha, o dia semanal poderá variar. Assim, associa-se um *slot* à intenção criada, neste caso no tipo *AMAZON.DATE*. Refira-se que a *Amazon* disponibiliza diversos tipos de *slot* que ajudam a construir sistemas robustos, mas também é possível criar tipos de *slot* personalizáveis. Para maior compreensão do que se pretende neste modelo de interação, isto é, a lista de intenções, apresenta-se na figura 4.2 uma das intenções criadas para o sistema desenvolvido. Acrescente-se que, todo o modelo de interação desenvolvido está presente no Anexo A.

```
1  {
2    "name": "SCHEDULE_CONFERENCE_WITH_DAY_HOUR",
3  "samples": [
4    "schedule a conference for {weekday} at {hour} hours"
5  ],
6  "slots": [
7    {
8      "name": "weekday",
9      "type": "WEEK_DAYS",
10     "samples": []
11   },
12   {
13     "name": "hour",
14     "type": "AMAZON.NUMBER",
15     "samples": []
16   }
17 ]
18 },
```

Figura 4.2: Exemplo de intenção para agendar uma audioconferência com dia semanal e hora

Posteriormente à criação do modelo de interação, definiu-se o tipo de *endpoint* da *Amazon Alexa Skill*, neste caso é um *endpoint* do tipo *HTTPS*, e associou-se o respetivo endereço deste *endpoint*. O endereço é o endereço do servidor já referido na arquitetura, criado em *NodeJS*. Refira-se que é necessário associar um nome de invocação à *Amazon Alexa Skill*, que será o nome que utilizador usará para invocar a *skill*. Por exemplo, se o nome de invocação for "James", o utilizador quando está a comunicar através do dispositivo *Amazon Echo Dot*, para entrar no contexto desta *skill* deve dizer: "Alexa, ask James...".

4.3.4 Servidor/ Wit Bot Platform

A componente servidor está incorporada na *Wit Bot Platform*. Neste sentido, o primeiro passo no desenvolvimento do servidor passou pela criação de um *endpoint* para comunicar com a *Amazon Alexa*. É neste *endpoint* que se deteta quando chega um pedido *HTTPS* proveniente da *Amazon Alexa*. Por sua vez, existe na plataforma um *endpoint* destinado à construção de mensagens para serem retornadas aos serviços *OTT*. Assim sendo, criou-se um construtor de mensagens destinado a ser utilizado pela *Amazon Alexa*. A plataforma permite o acesso às informações de todos os funcionários da *Wit-Software* através do protocolo *LDAP*. No mesmo sentido, para o *bot* existente de agendamento de audioconferências através do *Facebook Messenger*, já existia um módulo para associar a conta do *Facebook* do utilizador ao sistema, pelo que se manteve o mesmo sistema para a *Amazon Alexa*. No caso da utilização do assistente de voz desenvolvido, o pedido registo é feito através do envio de um pedido por voz enviado pela *Amazon Alexa* acompanhado de um *Amazon Alexa Card* enviado para a *Amazon Alexa App*, que se pode visualizar na Figura 4.3.

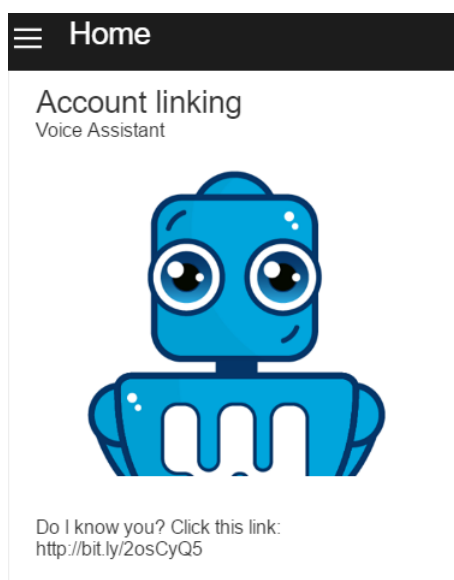


Figura 4.3: *Amazon Alexa Card* para associar conta *Wit-Software* e *Amazon*

O endereço enviado redireciona o utilizador para uma página, onde ente introduz as suas credenciais de acesso na rede *Wit-Software*. Este processo cria também, desde logo, uma conta *Aste-*

risk que fica associada à conta *Amazon* e *Wit-Software*. Esta associação é solicitada ao utilizador na primeira vez em que este último interage com o sistema. Apresenta-se na Figura 4.4 o processo de associação de contas.

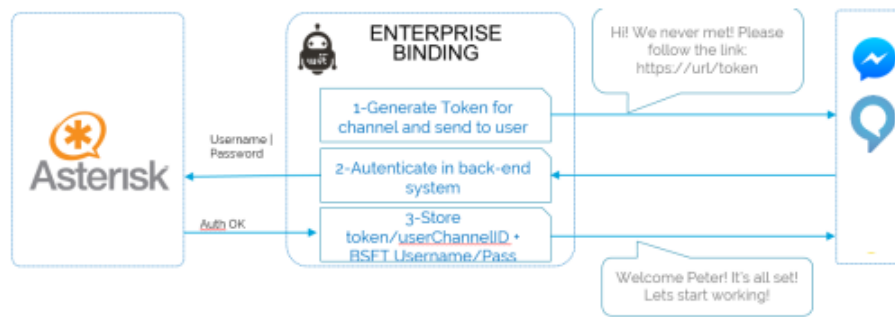


Figura 4.4: Processo de associação de contas

De seguida, foram criados fluxos com o *Bizagi Modeler* para os diferentes casos de uso do sistema. Alguns fluxos foram possíveis manter, sendo que foi alterada a parte de comunicação, neste caso através da *Amazon Alexa*, corrigidas algumas falhas nas *service tasks* e adicionadas várias exceções para lidar com *inputs* errados do utilizador. Para uma maior clareza do que foi desenvolvido, apresentam-se no Anexo B todos os fluxos do sistema que foram associados ao assistente virtual desenvolvido, distinguindo-se aqueles que foram alterados ou criados de raiz. Tal como ficou demonstrado na arquitetura do sistema, existe ainda comunicação com um servidor *Asterisk* que permite a criação das audioconferências. Esta comunicação funciona através de pedidos *HTTP* para o servidor em questão. Apresenta-se seguidamente um exemplo de um caso de uso simples da aplicação desenvolvida para que se perceba todo o processo da informação. Imagine-se que o utilizador pede à *Amazon Alexa* para se ligar à conferência número 50, ou seja, "Alexa, ask James to call me to the conference number fifty". De seguida, o serviço *Amazon Alexa* deteta uma intenção presente no enunciado do utilizador, neste caso a intenção *CH_DIALMEIN*. O serviço *Amazon Alexa* envia um pedido *HTTPS POST* para o servidor com toda a informação recolhida. O mais importante neste caso de uso é o nome da intenção e o *slot*, que neste caso, é o número da audioconferência. De seguida, através do *endpoint* criado, o servidor trata da receção da mensagem e procura nos ficheiros *AIML* por um enunciado que será o *Start Flow* da intenção detetada. O fluxo é então iniciado e, através de uma *Service Task*, é feito um pedido ao *HTTP POST* ao servidor *Asterisk* para estabelecer uma ligação entre a audioconferência número 50 e o número de telefone associado ao utilizador. O utilizador irá então receber uma chamada no seu telemóvel e, a partir daí, está ligado à audioconferência. Pressupõe-se que o utilizador já se associou à plataforma através da sua conta de funcionário *Wit*. Isto porque, para averiguar o número do telefone para se ligar, é necessário saber quem é o utilizador, para se apurar o seu número de telefone introduzido na *intranet Wit*. Posteriormente, o fluxo continua, sendo enviada para o utilizador uma mensagem através de uma *Send Task* e utilizando o construtor de mensagens para a *Amazon Alexa* e o *endpoint* criado. Após este passo, o fluxo termina, atingiu-se o *End event*.

4.3.5 Casos de uso funcionais

Nesta secção são apresentados os casos de uso que se encontram funcionais, isto é, que enunciados podem ser fornecidos ao assistente virtual/bot e que este tem capacidade de dar resposta. Juntamente com os enunciados, e para o alguns casos em que a resposta a estes é acompanhada por um *Amazon Alexa Card* na *Amazon Alexa App*, são apresentados os cartões retribuídos após o processamento dos fluxos conversacionais associados.

Dial-In

- "Alexa, ask James to call me to the conference number 125"
Permite ligar um utilizador, através do seu telefone particular, a uma audioconferência.
- "Alexa, ask James to call Tiago to this conference"
Permite ligar um utilizador registado na *intranet Wit* à atual audioconferência que está ativa.

Participants/Guests

- "Alexa, ask James about guests for this conference"
Aqui são listados os convidados para a audioconferencia em que o utilizador está ligado. Seguidamente o utilizador pode iterar sobre os diferentes convidados, sendo que, é apresentado ainda, na *Amazon Alexa App* a vista apresentada na Figura 4.5

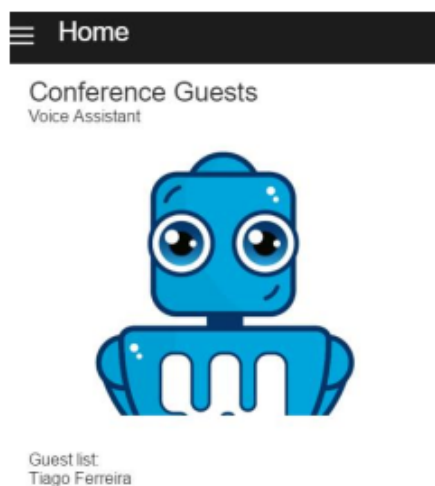


Figura 4.5: Vista de lista de convidados na *Amazon Alexa App*

- "Alexa, ask James to mute Tiago"
Com este enunciado o utilizador poderá silenciar um dos participantes da chamada onde se encontra ligado.

Assistente Virtual para gestão de conferências telefônicas

- "Alexa, ask John for conference participants"
É retribuída com esta questão a lista de participantes da chamada ativa do utilizador.

Schedule

- "Alexa, ask James to schedule a conference for Friday at ten"
- "Alexa, ask James to schedule a conference for Friday"
- "Alexa, ask James to schedule a conference"
Com estes três *inputs* o utilizador poderá agendar uma audioconferência para sexta às dez horas da tarde, ou se este não disser o dia nem a hora, o *bot*/assistente virtual, apurará esses dados. Seguidamente o fluxo continuará, e será ainda apurado o título da audioconferência, assim com possíveis convidados. Após agendada a audioconferência será apresentado o cartão representado na Figura 4.6 na *Amazon Alexa App*.

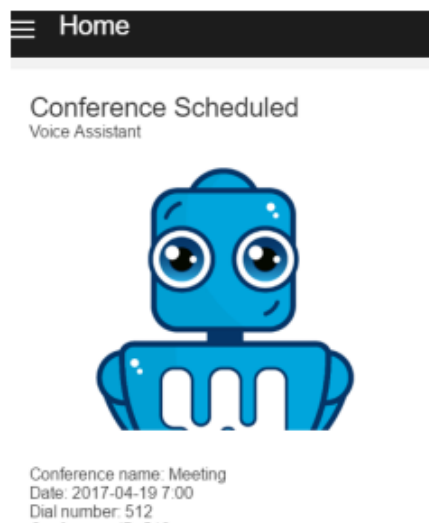


Figura 4.6: Vista de conferencia agendada na *Amazon Alexa App*

Who Is

- "Alexa, ask James who is Tiago Ferreira"
Permite devolver informações sobre uma pessoa registada na rede *intranet* da *Wit-Software*. Se existir mais do que uma pessoa na rede com o mesmo nome, é possível iterar sobre as diferentes pessoas e mostrar detalhe de uma pessoa em particular, assim como se apresenta na Figura 4.7.

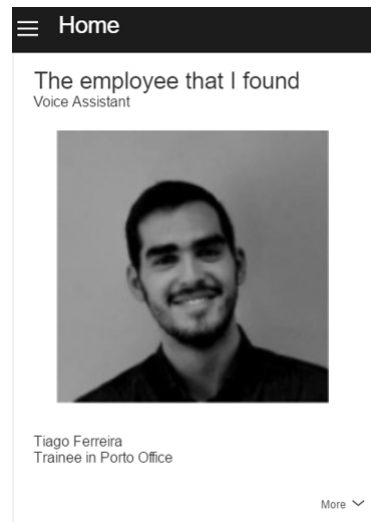


Figura 4.7: Vista de resposta ao enunciado "Who is Tiago Ferreira" na *Amazon Alexa App*

Other

- "Alexa, ask James for next conferences"
Devolve a próxima conferência agendada para o utilizador em questão.
- "Alexa, ask James what are my conferences for today"
Devolve uma lista de todas as conferências que o utilizador tem para o dia em questão. Estas podem ser iteradas para, seguidamente, se devolver os dados de uma conferência específica.

4.4 Conclusões

Este capítulo mostrou o como foi desenvolvido o assistente virtual que auxilia no agendamento de conferências, assim como, os casos de uso que estão funcionais em relação a este mesmo assistente. Infelizmente, à data de início de realização desta dissertação, ainda não havia sido lançado o dispositivo *Amazon Echo Show*. Este com certeza teria sido uma mais-valia para tornar os casos de uso mais interessantes, especialmente nos processos conversacionais em que é necessário iterar sobre funcionários ou conferências. Refira-se que a *Amazon* anunciou que em breve irá permitir enviar para a *Amazon Alexa* notificações *Push*. Por outras palavras, poderá ser enviada informação diretamente de servidores para a *Amazon Alexa*, para que esta, posteriormente, interaja diretamente com o utilizador sem que este tenha de perguntar alguma coisa. Esta funcionalidade poderá ser importante para, por exemplo, lembrar o utilizador que existe uma audioconferência prestes a começar. Seguidamente o utilizador poderia solicitar uma chamada para entrar na audioconferência.

Assistente Virtual para gestão de conferências telefónicas

Capítulo 5

Assistente virtual de chamadas recebidas

Nesta secção é feita uma análise, de forma mais detalhada, ao motivo que levou à elaboração do assistente virtual de chamadas recebidas, assim como, são expostas as questões que se levantam à sua utilização. É ainda neste capítulo que é exposta a abordagem adotada com vista à resolução do problema, apresentada a arquitetura deste assistente virtual, descritas as ferramentas e tecnologias utilizadas e expostos os casos de utilização.

5.1 Atendimento automático de chamadas

A chamada telefónica continua a ser a maneira mais rápida e fácil de contactar uma pessoa e, simultaneamente, a que oferece uma experiência mais pessoal. Não obstante a existência de outros meios de comunicação, como canais de *email* ou *chat*, as pessoas continuam a preferir uma chamada telefónica direta para a pessoa que querem contactar. Contudo, hoje em dia, a agenda de um empresário é preenchida de reuniões e outras atividades que o levam a estar indisponível para responder a todas as chamadas que recebe [AB01]. Posteriormente, essas chamadas caem no esquecimento ou já não são retornadas em tempo útil.

As chamadas mal sucedidas são um sério obstáculo para a produtividade e uma fonte de frustração, uma vez que ambos os interlocutores perdem tempo [SJ08]. Além disso, o horário da chamada telefónica pode não ser apropriado ou não alcançar o recetor num local adequado [SJ08].

Acrescente-se que, muitas vezes, as chamadas recebidas são apenas para marcação de reuniões ou encontros, confirmar assuntos já discutidos entre os intervenientes ou solicitar uma decisão para determinada tarefa.

Existem soluções que permitem abordar o problema apresentado anteriormente. Uma dessas soluções é a tecnologia de *Voz para SMS*, disponibilizada apenas por algumas operadoras de telecomunicações. Esta tecnologia permite a uma pessoa que está a tentar contactar um subscritor do serviço *Voz para SMS* deixar uma mensagem de voz, quando este último se encontra indisponível. Esta mensagem é posteriormente convertida em texto e enviada sob a forma de uma mensagem

SMS para o subscritor permitindo um acesso rápido ao respetivo conteúdo através da leitura da mensagem escrita [vod14]. Outra solução existente é o serviço de *voicemail*. Este serviço permite aos seus utilizadores trocar mensagens de voz, isto é, possibilita ao chamador enviar uma gravação de áudio para um destinatário que não respondeu à chamada [SJ08].

Contudo, os utilizadores continuam a evitar o uso destas tecnologias [Día05, SJ08], apontando a perda de tempo como fator determinante. Realizou-se ainda um inquérito quem contou uma amostra de 104 indivíduos, para avaliar se existe de facto um problema no mercado atual. O inquérito segue no Anexo C juntamente com as respostas ao mesmo. A amostra era constituída maioritariamente (85,6%), por pessoas entre os 18 e os 25 anos, sendo que 13,5% tinha idade compreendida entre os 26 e os 40 anos e, por último, 1%, tinha idade entre os 41 e os 60 anos. Neste inquérito apurou-se que 88,5% dos participantes já ouviu falar nas tecnologias de correio de voz (*voicemail*) e *Voz para SMS*. Sendo que destes 88,5%, 71,2% já utilizaram correio de voz e 27,9% *Voz para SMS*. Ambos os gráficos dos resultados podem ser consultados na Figura 5.1.

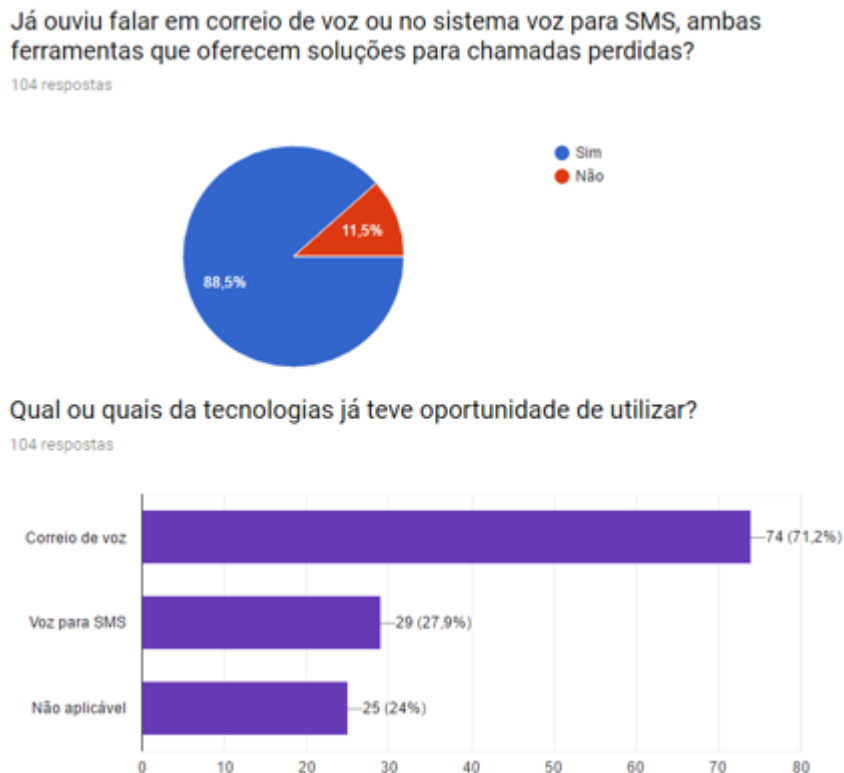


Figura 5.1: Resultados de utilização e conhecimento das tecnologias de correio de voz e voz para SMS por parte da amostra

Por outro lado, concluiu-se também que, 57,8% dos inquiridos que já usaram uma destas tecnologias, encontram-se satisfeitos com a sua qualidade de serviço, e 3,6%, muito satisfeitos. Já 27,7% afirmaram que se encontram satisfeitos e 10,8% nada satisfeitos. Estes dados permitem dizer que, embora a maioria dos inquiridos que conhecem as tecnologias que permitem dar resposta

a chamadas nativas perdidas, se encontrem satisfeitos, existe ainda uma grande percentagem de indivíduos pouco ou nada satisfeitos. O gráfico interpretativo desta questão encontra-se na Figura 5.2.

Em que medida ficou satisfeito com a sua qualidade de serviço?

83 respostas

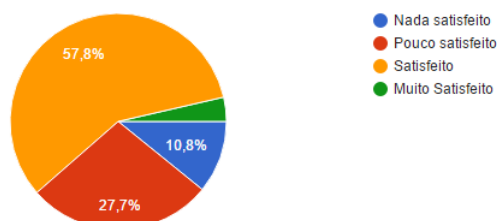


Figura 5.2: Resultados de satisfação com a qualidade do serviço das por parte da amostra

Um dos dados que importa também fazer referência advém do facto de, embora existam cada vez mais tecnologias de comunicação VoIP, como já foi referido, a grande maioria dos inquiridos, 87,5%, continua a executar chamadas de voz nativa na altura de contactar alguém. A Figura 5.3 apresenta esses resultados e não deixa dúvidas em relação a este assunto.

Na altura de contactar alguém, opta por realizar uma chamada nativa, isto é, uma chamada regular através da rede pública de telefonia comutada, ou liga através de aplicações VoIP, como o Skype ou o WhatsApp?

104 respostas

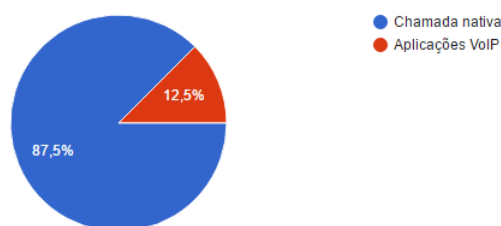


Figura 5.3: Resultados de serviço preferencial de chamadas de voz

Surgem assim questões às quais se pretende dar resposta. Uma dessas questões é como conseguir um sistema autónomo, que consiga dar resposta a uma chamada sem intervenção do recetor da chamada ou alguém no seu papel. De facto, este sistema terá de tomar uma decisão, não sabendo à priori qual é o tema pelo qual o utilizador recebeu o contacto. Isto coloca ainda outras questões pertinentes, por exemplo, até que ponto esse sistema será autónomo? Qual será o limite da sua autonomia? De facto, o utilizador ficará dependente deste sistema. Por outro lado, a aplicação terá de ser intuitiva, agradável e, sobretudo, eficaz. Ninguém irá usar um sistema que não consiga perceber mais de metade dos motivos dos contactos recebidos. É preciso garantir que o produto a

desenvolver seja superior às tecnologias já existentes para o problema, ou seja, superior às tecnologias que procuram resolver o problema da indisponibilidade dos utilizadores para responder às chamadas recebidas.

5.2 Aplicação móvel e assistente virtual

É apresentada nesta secção a solução concebida para o problema apresentado. Na verdade, desenvolveu-se um sistema que, tirando partido de um protótipo de uma aplicação *Android* e em conjunto com um *bot*, permite dar resposta imediata às chamadas perdidas pelos seus utilizadores. O *bot* funciona como um assistente virtual, com capacidade de comunicar com o interlocutor através de fala e usando as mais recentes tecnologias de linguagem natural e reconhecimento de voz. A aplicação tem dois modos de utilização: automático e ocupado. No modo automático a chamada é atendida pelo *bot* imediatamente após a receção da mesma. No modo ocupado, o utilizador terá de rejeitar a chamada para que esta seja atendida pelo *bot*. Após atender a chamada, o assistente virtual/*bot* procura imediatamente perceber qual foi o motivo para o contacto. Seguidamente, o assistente irá combinar com o interlocutor uma data para um possível contacto posterior por parte do utilizador da aplicação. Por sua vez, o utilizador, acedendo à sua aplicação, poderá, além de configurar quando deverá ser o assistente pessoal a atender a chamada, consultar as últimas chamadas atendidas pelo assistente virtual e adicionar lembretes automaticamente com a data e hora acordadas entre o assistente e o interlocutor. Nesta consulta, ele poderá também verificar todo o fluxo conversacional trocado entre o seu assistente e o interlocutor, assim como reproduzir a gravação do que foi dito em cada iteração. Isto permite, por exemplo, ao utilizador verificar se é um assunto que exige uma resposta urgente ou se a chamada pode ser retribuída só quando este estiver disponível.

5.3 Implementação

O sistema desenvolvido foi implementado tirando partido de diversas ferramentas externas, ferramentas essas que serão enumeradas nas subsecções seguintes. Serão também apresentados os diferentes componentes do sistema, as interações entre esses componentes e os casos de uso.

5.3.1 Ferramentas e tecnologias adotadas

Como foi possível verificar com a análise realizada na revisão bibliográfica, existem diversas ferramentas que poderiam ajudar no desenvolvimento da solução, isto é, no desenvolvimento do *bot*/assistente virtual e da aplicação móvel *Android*. Optou-se por seleccionar o *Amazon Lex* como a ferramenta que irá ser responsável pela análise das frases ditas pelo interlocutor durante a chamada. Esta decisão foi tomada com base nas mais-valias já demonstradas desta ferramenta, nomeadamente o seu serviço de compreensão de linguagem natural (*NLU*), que permitirá ao interlocutor ter conversas com o assistente que pareçam reais. No mesmo sentido, foram utilizadas

funções *Lambda* para executar determinadas ações após terminar o fluxo conversacional com os serviços do *Amazon lex*. Esta ferramenta permite executar funções de código sem necessidade de gerir ou possuir servidores [ama17b]. Assim, permite comunicar com os *endpoints* do servidor principal do sistema de forma simples. Outra das ferramentas utilizadas, neste caso para transcrição da voz do interlocutor, foi o *Google Cloud Speech API*. Esta plataforma possui recursos capazes de converter áudio em texto através de modelos de redes neuronais.

“O *Google Cloud Speech API* possui uma grande precisão em transcrições de voz para texto em tempo real. Os seus grandes rácios de precisão significam que se pode ajudar os vendedores a reunir o maior número de informação possível a partir das interações telefónicas com os seus clientes e aumentar as vendas.” — Gary Graves, CTO and Co-Founder, InterActiveTel [goo17]

Através desta afirmação de Gary Gaves consegue-se perceber o grande potencial desta ferramenta, que ajudará a diminuir possíveis falhas de perceção do discurso do interlocutor durante o processo conversacional. O *Twilio* foi outra das tecnologias indispensáveis à construção da solução proposta. No fundo, o *Twilio* permitiu a receção de chamadas através do uso de *web services*. Através do *Twilio* foi possível enviar pedidos para o servidor com informações relativas a uma determinada chamada, assim como, enviar informações de volta para essa mesma chamada. No desenvolvimento do servidor principal, que trata de receber e enviar pedidos, gerir todo o processo de comunicação entre as diferentes *APIs* e comunicar com a aplicação móvel, foi usada a ferramenta *Node.js*. *Node.js* é uma ferramenta *open-source* que permite executar código *JavaScript* do lado do servidor. Através da utilização de *Node.js* é possível criar aplicação com uma grande escalabilidade. O seu principal objetivo é ajudar programadores na criação de aplicações de alta escalabilidade, capazes de manipular milhares de conexões em simultâneo numa única máquina [McL11].

5.3.2 Arquitetura

Nesta secção é apresentada a arquitetura de alto nível da solução proposta, isto é, uma visão global de como todas as ferramentas comunicam entre si, com o servidor principal e com a aplicação móvel *Android*. Os componentes trocam dados com pedidos REST, que assentam sobre o protocolo *HTTPS* e *WebSocket*. A Figura 5.4 mostra a arquitetura aqui descrita. A arquitetura é descrita mais pormenorizadamente nas subsecções seguintes.

Assistente virtual de chamadas recebidas

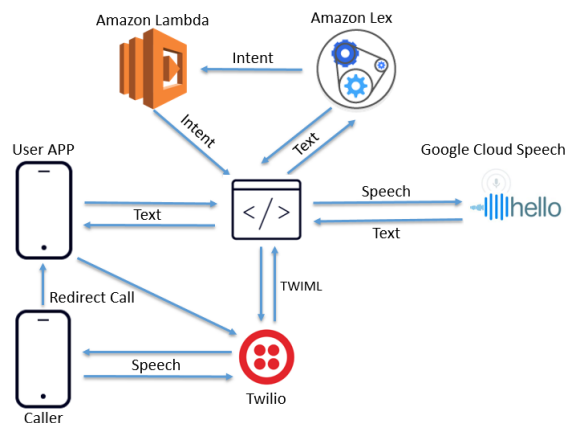


Figura 5.4: Arquitetura de alto nível

5.3.3 Servidor

O primeiro passo para a construção desta aplicação passou pelo redirecionamento das chamadas recebidas pelo utilizador. Este redirecionamento foi feito para a plataforma *cloud* de comunicações *Twilio*. De seguida, o *Twilio* envia um pedido (*POST*) para o servidor principal. É neste servidor que todo o processo é gerido. Inicialmente é retornada ao *Twilio* uma mensagem inicial ("Mensagem de boas-vindas"). Por sua vez, o *Twilio* retransmite essa mensagem para o interlocutor, isto é, a pessoa que tentava contactar o utilizador. Refira-se que o *Twilio* possui uma lista de instruções num formato próprio chamado *Twiml* (*Twilio Markup Language*). As instruções *Twiml* são usadas para dizer ao *Twilio* o que fazer quando se recebe, ou durante, uma chamada telefónica. Quando se recebe ou se executa uma chamada para um dos números associados a uma conta *Twilio*, o *Twilio* irá captar o *URL* associado a esse número e enviar um pedido. Seguidamente o *Twilio* interpreta esse pedido e determina o que fazer, por exemplo gravar áudio ou reproduzir uma mensagem [twi17b]. Apresenta-se na Figura 5.5 um exemplo de uma mensagem no formato *Twiml* para reproduzir uma fala a pedir ao utilizador para deixar uma mensagem de voz e, seguidamente, gravar a voz do utilizador.

```
1 <?xml version="1.0" encoding="UTF-8"?>
2 <Response>
3   <Say voice="woman">Please leave a message after the tone.</Say>
4   <Record maxLength="20" />
5 </Response>
```

Figura 5.5: Exemplo de mensagem no formato *Twiml*

Atendendo ao que foi dito, seguidamente, após o envio da mensagem de inicial através de operação *Twiml* `<Say>`, será executada uma operação `<Record>`, operação que ficará à escuta durante um tempo previamente programado. Esta gravação será seguidamente enviada para o *Google Cloud Speech API* para ser convertida para texto. Esta plataforma recomenda o envio dos

ficheiros de áudio num formato sem perdas, por exemplo *LINEAR16*. Contudo, o *Twilio* gera um ficheiro de áudio no formato WAV binário. Além disso, o *Google Cloud Speech API* recomenda um *sampling rate* de 16000Hz ao invés dos limitados 8000Hz usados normalmente em chamadas telefónicas. Assim, as transcrições geradas não são tão eficientes como se pretendia. Após este passo, o texto resultante é enviado para o *Amazon Lex* para reconhecimento da intenção do texto e este devolve uma resposta apropriada, através de uma função *Lambda*, para obtenção das informações necessárias.

Este fluxo continuará até que alguma ação seja tomada, por exemplo, a promessa de um contacto posterior com as informações reunidas. Estas informações são posteriormente enviadas para o *smartphone* do utilizador.

5.3.4 Amazon Lex

No *Amazon Lex* foi criado um *bot* e, até ao momento de escrita desta dissertação, associada uma intenção. Esta intenção procura obter o nível de urgência da chamada e o dia e hora em que o interlocutor pretende ser contactado de volta por parte do utilizador do sistema desenvolvido. Assim sendo, existem 2 *slots*; o processo de construção já foi referido no Capítulo 2. Um dos *slots* é do tipo *AMAZON.DATE* e outro é do tipo *AMAZON.LITERAL*.

5.3.5 Aplicação móvel

Tal como foi referido no Capítulo 1, um dos objetivos seria a elaboração de um protótipo de uma aplicação *Android*, que conseguisse ser por um lado atrativa para o utilizador e, por outro, conseguisse mostrar ao utilizador toda a informação que é necessário apresentar, isto é, toda a informação reunida pela conversação entre o *bot* e o interlocutor.

A aplicação é constituída por três atividades. A atividade principal é responsável por permitir ao utilizador selecionar o modo ativo da aplicação. Por outras palavras, o utilizador deve indicar se pretende que o assistente pessoal responda a todas as suas chamadas, se quer que este apenas responda a chamadas quando este último recusa a chamada, ou, por último, se pretende desativar qualquer um dos dois últimos modos. É apresentada na Figura 5.6 uma visualização desta atividade da aplicação. Refira-se que todo o fluxo da aplicação pretende dar a entender ao utilizador que está a interagir com um assistente virtual. No mesmo sentido, procurou-se dar personalidade a este assistente virtual. Atendendo à aposta da empresa *Wit Software* no mercado Japonês e, consequentemente, às tendências deste mercado, optou-se por selecionar a *vocaloid Hatsune Miku* como “corpo” do assistente virtual. *Hatsune Miku* [voc12] é uma *vocaloid* dobrada por um sintetizador de voz que já deu vários concertos em palco como uma projeção 3D. Como uma “ídolo virtual”, *Hatsune Miku* realizou o seu primeiro concerto “ao vivo” em 22 de agosto de 2009, sendo considerada a mais popular e bem conhecida *vocaloid*. Acrescente-se que *Miku*, como também é conhecida, foi a primeira *vocaloid* a ser considerada um ídolo pop.

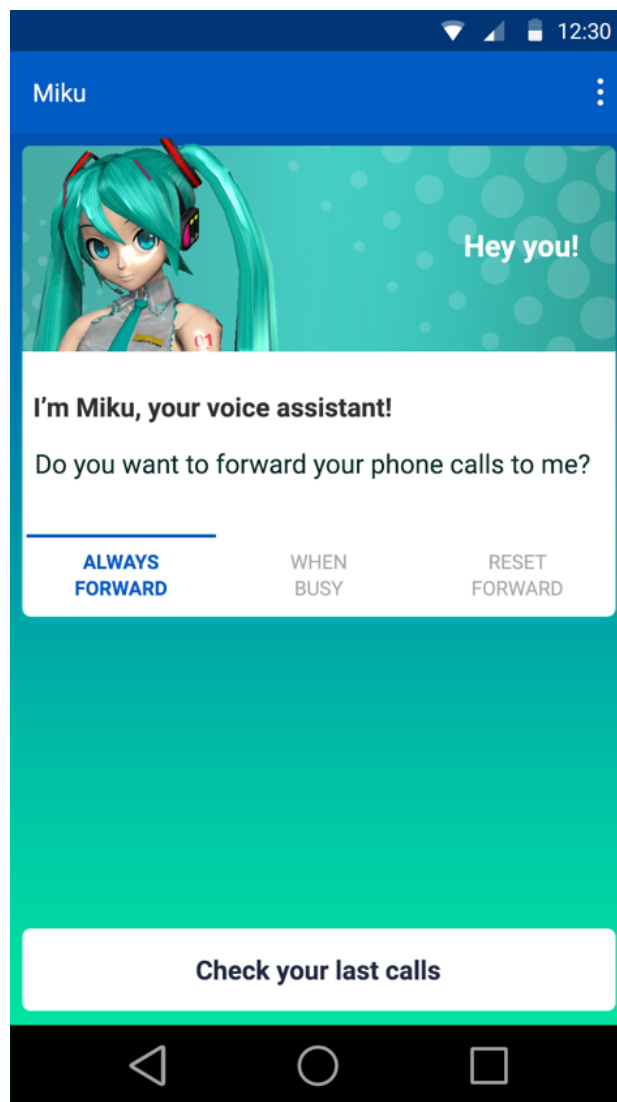


Figura 5.6: Visualização do menu principal da aplicação

A segunda atividade, representada na Figura 5.7, representa o conjunto de chamadas perdidas pelo utilizador. Nesta, apresenta-se uma fotografia da pessoa que o contactou, o nome dessa mesma pessoa, e as horas preferenciais indicadas por essa pessoa para um contacto posterior. A fotografia e o nome da pessoa é conseguida com base na lista de contactos do utilizador. Na realidade, é enviado pelo servidor o número associado ao contacto e, no lado da aplicação, é verificado o nome correspondente a esse contacto e a imagem associada. Se pessoa que tenta contactar o utilizador não está presente na sua lista de contactos, apresenta-se o número de telefone e uma imagem de utilizador anónimo. Toda a referida informação é apresentada sob a forma de um cartão. São ainda apresentadas ao utilizador as possibilidades de ligar de volta à pessoa que o contactou, adicionar um lembrete à sua agenda com base na disponibilidade dessa pessoa, ou verificar todo o fluxo conversacional trocado entre esta última e o assistente virtual / bot.

Assistente virtual de chamadas recebidas

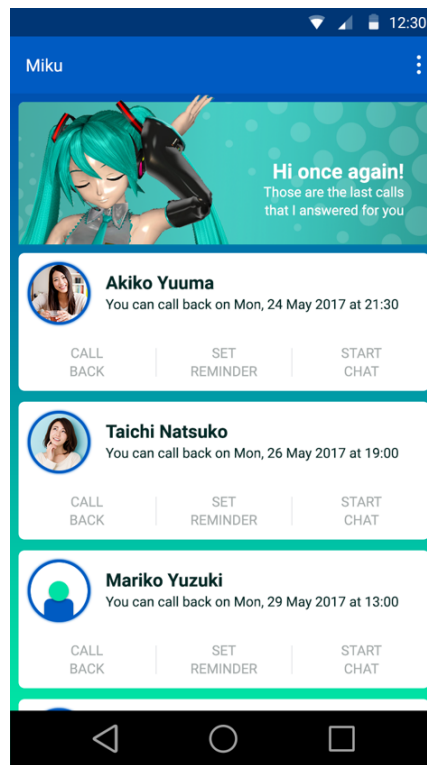


Figura 5.7: Visualização da lista de chamadas respondidas pelo assistente virtual na aplicação

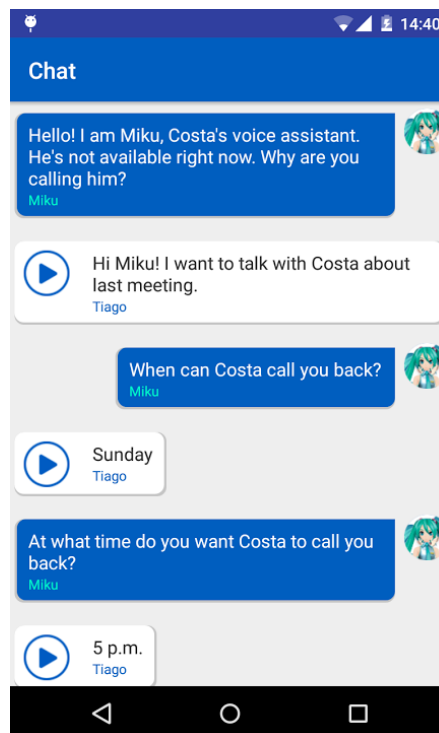


Figura 5.8: Visualização da vista de *chat* da aplicação

O fluxo conversacional contempla a terceira atividade da aplicação e permite não só visualizar todas as falas ditas pelo assistente e pelo interlocutor, assim como, reproduzir o conteúdo de áudio associado a cada fala do interlocutor. A vista desta atividade é apresentada na figura 5.8.

5.3.6 Comunicação Aplicação móvel Servidor

Como foi demonstrado na Secção 5.3.3, existe uma troca de informação entre o servidor e a aplicação móvel em tempo-real. Na verdade, sempre que o utilizador abre a aplicação, cria-se um canal de comunicação *full-duplex* através do protocolo *Web Socket*. Seguidamente, o servidor, sempre que obtém informações válidas após uma chamada, procura enviar estas mesmas informações para a aplicação. Note-se a necessidade de existência de ligação à Internet durante todo este processo. A informação recolhida do servidor é posteriormente armazenada numa base de dados *SQLite* no lado da aplicação. Neste momento a base de dados apenas contém uma tabela, que permite guardar as informações relativas a cada chamada perdida do utilizador.

5.3.7 Principais bibliotecas externas utilizadas

Para a implementação da aplicação, foram usadas bibliotecas externas. Apresentam-se de seguida as bibliotecas utilizadas:

*Card Library*¹ – Esta biblioteca agrupa uma variada lista de *Android Cards* para tornar as interfaces gráficas mais apelativas. Além disso, esta biblioteca usa a *Google CardView*, o que permite configurar e personalizar os cartões.

*AndroidAsync*² – *AndroidAsync* é uma biblioteca de protocolos de baixo nível. Esta biblioteca permitiu a implementação de um canal de comunicação utilizando o protocolo *Web Socket*. O protocolo *Web Socket* permite uma comunicação *full-duplex*, ambas as direções do canal, através de uma única conexão *TCP*.

*Glide*³ – O *Glide* consiste num *framework* de código aberto que permite uma gestão eficiente e simples de conteúdos média. Esta ferramenta retira ao programador a preocupação de gestão de memória e *cache* do disco. O seu principal objetivo é tornar a funcionalidade de *scrolling* de conteúdos o mais rápida e suave possível, assim como o redimensionamento de imagens e a amostragem de imagens remotamente.

¹<https://github.com/gabrielemariotti/cardslib>

²<https://github.com/koush/AndroidAsync>

³<https://github.com/bumptech/glide>

5.4 Conclusões

Neste capítulo ficou patente o sistema que foi desenvolvido para procurar dar resposta às questões que se levantaram. No mesmo sentido, demonstrou-se como todo o sistema está estruturado, assim como cada componente individual. De forma a tornar este sistema mais atrativo para os utilizadores, considera-se a adição de mais casos de uso. Esta é apenas uma prova de conceito e, após uma consolidação do produto, poder-se-ão tirar conclusões mais elucidativas sobre o produto. O objetivo foi criar uma alternativa para as soluções existente e tal foi conseguido.

Assistente virtual de chamadas recebidas

Capítulo 6

Validação e resultados

Neste capítulo serão validados os resultados obtidos após o desenvolvimento dos dois sistemas apresentados nos capítulos 4 e 5. Na realidade, foram realizados testes de usabilidade às provas de conceito elaboradas, assim como testes adicionais para verificar se os sistemas dão resposta às questões que se levantaram para a realização desta dissertação. Os testes de usabilidade ajudam a detetar a facilidade que uma aplicação possui para ser eficazmente manipulada pelo utilizador. O ser humano possui uma capacidade de descobrir ou resolver problemas através de experiência prática e da observação. Esta capacidade denomina-se de capacidade heurística. Para avaliar a usabilidade dos sistemas poderíamos utilizar a chamada "avaliação heurística", definida por Nielsen e Molich no ano de 1994. Jack Nielsen recomenda utilizar entre 3 a 5 avaliadores, sendo que esta regra pode variar de acordo com as necessidades do sistema em estudo. Ele estimou que, em média, com um só avaliador consegue-se detetar cerca de 35% dos problemas de usabilidade existente na interface de uma aplicação, enquanto que 5 avaliadores conseguem em média detetar 75% desses problemas.

Seguidamente apresentam-se as 10 heurísticas de usabilidade de Nielsen e Molich [heu17]:

- Visibilidade do estado do sistema
- Correspondência entre o sistema e mundo real
- Controlo por parte do utilizador e liberdade de ação
- Consistência e padrões definidos
- Prevenção de erros
- Minimizar a sobrecarga de memória do utilizador
- Flexibilidade e eficiência de utilização
- *Design* minimalista e estético

- Boas mensagens de erro
- Ajuda e documentação

Estas heurísticas resistiram ao passar dos anos, e proporcionavam aos *designers* uma forma rápida e fácil de avaliar a usabilidade de uma interface contra princípios de *design* universais. No entanto, será que devem ser estas heurísticas aplicadas a *bots*/assistente virtuais? Em [Usa16], concluiu-se que estas heurísticas ainda podem ser aplicadas no caso dos *bots*, mas definiu-se que faria mais sentido agrupar e redefinir o foco de algumas das heurísticas anteriores. Assim, mostra-se na Tabela 6.1 as heurísticas definidas para testar a usabilidade de assistentes virtuais.

Tabela 6.1: Heurísticas para *chat bots*

Visibilidade do estado do sistema e minimizar a sobrecarga de memória do utilizador	O sistema deve manter sempre os utilizadores informados sobre o que está acontecer, e dar-lhe sempre opções para solicitar informações adicionais em qualquer ponto
Correspondência entre o sistema e mundo real e boas mensagens de erro	Conhecer o público-alvo e não alterar o estilo de comunicação
Controlo por parte do utilizador e liberdade de ação e prevenção de erros	Obter confirmação do utilizador em pontos críticos e fornecer atalhos de saída para interações de vários passos.
Consistência e padrões definidos e <i>design</i> minimalista e estético	Manter o estilo de comunicação consistente e a personalidade/vos do <i>bot</i>
Flexibilidade e eficiência de utilização	Fornecer atalhos para utilizador avançados
Ajuda e documentação	Fornecer ajuda dentro do <i>bot</i>

Todos os indicadores referidos na Tabela 6.1 foram considerados para testar a usabilidade do assistente virtual para gestão de audioconferências telefónicas, como se apresentará na Secção 6.1. No entanto, optou-se por não se usarem estes indicadores para o *bot* de chamadas recebidas. Isto porque, até à data, este último assistente só possui um caso de uso funcional para interação com o interlocutor da chamada, pelo que não faria sentido ter em conta todos os indicadores.

6.1 Assistente Virtual para agendamento de conferências telefónicas

Atendendo ao carácter confidencial desta dissertação, que como já foi referido foi proposta pela empresa Wit-Software, apenas participaram pessoas internas da empresa nos testes de usabilidade deste sistema. Tal como recomenda Nielsen, participaram 5 avaliadores. Os mesmo 5 avaliadores responderam a um pequeno questionário para avaliar o cumprimento dos objetivos deste sistema.

6.1.1 Testes de usabilidade

Previamente, explicaram a cada um dos utilizadores as funcionalidades do assistente virtual, como começar cada iteração com o mesmo através do *Amazon Echo Dot*, isto é, dizer “Alexa, ask James...” e como cancelar ações. De seguida, forneceu-se aos avaliadores uma lista de tarefas principais e anotaram-se detalhes considerados importantes das iterações dos avaliadores com o assistente. A lista das tarefas solicitadas foi a seguinte:

1. Obter ajuda do assistente virtual
2. Agendar uma conferência para um dia e hora à escolha
3. Consultar a lista de conferências para hoje e obter detalhes de uma dessas conferências
4. Ligar-me à conferência número 455
5. Consultar as definições do meu *Softphone*
6. Perguntar quem é o Tiago Ferreira

Todos os utilizadores conseguiram realizar com sucesso as 6 tarefas solicitadas, embora nem sempre à primeira tentativa. As tarefas que se revelaram mais difíceis de executar pelos utilizadores foram a tarefa 2 e 6. O utilizadores procuravam muitas vezes dizer o dia do mês, o mês e o ano para o agendamento. Contudo, o sistema apenas está preparado para, ou não receber nenhuma informação de dia e hora e vai fornecendo ao utilizador possíveis falas, ou recebe de uma vez só toda a informação, mas com o dia semanal. Por outras palavras, um utilizador avançado que pretende introduzir os dados todos de uma vez deveria dizer: "Alexa, ask James to schedule a conference for friday at seven". No caso da tarefa número 6, o problema advém de o sistema, como utiliza o serviço *Amazon Lex* para interpretar as falas, nem sempre perceber o nome Tiago Ferreira. O serviço *Amazon Lex* ainda não suporta português. Confrontando os resultados dos testes com base nas 6 heurísticas definidas chegaram-se às conclusões apresentadas na Tabela 6.2.

Validação e resultados

Tabela 6.2: Resultados da análise dos teste de usabilidade com base nas seis heurísticas para *chat bots*

<p>Visibilidade do estado do sistema e minimizar a sobrecarga de memória do utilizador</p>	<p>O assistente percebe mensagens se forem bem estruturadas, ou seja, falha a perceber muitas falas de linguagem natural para determinadas tarefas. Por outro lado, por norma, oferece sempre ao utilizador informações adicionais para o próximo passo</p>
<p>Correspondência entre o sistema e mundo real e boas mensagens de erro</p>	<p>Quando as mensagens não são bem estruturadas numa primeira iteração o resultado é sempre o mesmo: "Desculpa, não percebi. Podes dizer o que pretendes de uma maneira diferente."</p>
<p>Controlo por parte do utilizador e liberdade de ação e prevenção de erros</p>	<p>Após a primeira iteração o sistema é eficiente a prevenir erros. Por exemplo, se o <i>bot</i> souber que é para agendar uma conferencia, mas não percebeu o dia em questão ou a hora, este indica essa informação ao utilizador e fica a aguardar novos enunciados da parte deste. O mesmo acontece para outra tarefas, no caso da ligação a uma conferência solicita um novo número quando o número de ligação introduzido não tem qualquer conferência agendada. Por outro lado, a qualquer momento o utilizador pode cancelar uma determinada tarefa dizendo, por exemplo, "cancel"</p>
<p>Consistência e padrões definidos e <i>design</i> minimalista e estético</p>	<p>O estilo da comunicação é sempre o mesmo</p>
<p>Flexibilidade e eficiência de utilização</p>	<p>Neste caso, o <i>bot</i> vai guardando sempre as informações introduzidas até uma ação final ser tomada. Além disso, no caso do agendamento permite ao utilizador normal introduzir informações passo a passo ou, a um utilizador avançado, dizer toda a informação necessária de uma só vez</p>
<p>Ajuda e documentação</p>	<p>O sistema tem a capacidade de fornecer ajuda ao utilizador sobre as suas funcionalidades. No entanto, esta ajuda não é permitida a meio da execução de uma tarefa</p>

6.1.2 Inquérito sumário

Elaborou-se um questionário com 3 simples questões para averiguar a eficácia do produto em responder aos objetivos desta dissertação. O questionário e os resultados do mesmo estão disponíveis no Anexo E. Os resultados não são muito significativos por dois motivos: a amostra é pequena, 5 pessoas, e todas elas estão inseridas no mesmo contexto empresarial. Analisando os resultados, verificou-se que todos os utilizadores concordaram que o sistema seria, ou poderia ser, uma mais-valia no seu dia-a-dia dentro da empresa. A Tabela 6.3 apresenta alguns dos resultados obtidos ao questionário.

Tabela 6.3: Resultados importantes ao inquérito sobre o assistente virtual para chamadas recebidas

Considera que a prova de conceito que acabou de testar teria utilidade no seu dia-a-dia na empresa?	Estaria disposto a pagar por um produto como este?	Que sugestões apresenta para melhorar este produto futuramente?
Sim	Talvez	Aumentar o leque de frases possíveis para cada tarefa.
Talvez	Não	Acrescentar funcionalidades de gestão de conferências e a eficácia em entender o que se diz.
Sim	Sim	Acrescentar funcionalidades de gestão de conferências e a eficácia em entender o que se diz.
Talvez	Não	O bot devia perceber mais frases e manter mais contexto do que lhe dizemos.
Talvez	Não	Podias aumentar o vocabulário do bot. É muito direto, tipo pergunta-resposta.

Contudo, quando se inquiriu os avaliadores se estes pagariam pelo produto, apenas um avaliador disse que sim e outro disse que talvez pagasse. Assim, conclui-se que o serviço não tem tanto valor para os utilizadores como se pensaria, pois apresentando um custo, eles não o consideraram extremamente necessário. Por último, solicitou-se aos utilizadores sugestões de melhoria da aplicação desenvolvida. Quase todos os utilizadores referiram uma melhoria no processo de comunicação e compreensão de falas aleatórias. Além disso, sugeriram a adição de funcionalidades ao sistema.

6.2 Assistente Virtual de chamadas recebidas

Tal como para o assistente de gestão de audioconferências telefónicas, foram realizados testes de usabilidade para o assistente virtual de chamadas recebidas, assim como foi elaborado um inquérito sumário após a utilização do mesmo. Os avaliadores foram os mesmos que testaram o sistema de gestão de conferências.

6.2.1 Testes de usabilidade

Para a realização dos testes de usabilidade deste sistema teve-se em conta duas interações por parte dos avaliadores: conversação com o assistente virtual quando liga ao utilizador proprietário da aplicação e a interação com a aplicação móvel, como recetor da chamada, após a chamada ser realizada. Assim, para a interação com o assistente de voz durante a chamada apenas foi solicitado aos avaliadores que ligassem à pessoa em questão e deixassem o assistente gerir o processo conversacional. Já na parte de avaliação da usabilidade da aplicação, foram solicitadas as seguintes tarefas:

1. Ativar o redirecionamento de todas as chamadas para o assistente virtual
2. Consultar a lista das últimas chamadas atendidas pelo assistente
3. Agendar um lembrete para ligar à última pessoa que ligou com a hora sugerida por esta
4. Consultar o processo conversacional da última chamadas atendida pelo assistente e reproduzir o motivo do contacto dessa chamada

Em relação ao processo conversacional, e tendo em conta que o assistente realiza questões fechadas e apenas permite um caso de uso, todo o processo é concluído facilmente. Além disso, mesmo nos casos em que o assistente não percebeu à primeira o que foi dito pelo avaliador, este solicitou a repetição da resposta à questão anterior, mantendo um processo natural de comunicação. De facto, as questões colocadas pelo assistente virtual quando não percebe à primeira tentativa o que foi dito são diferentes da primeira questão, embora com o mesmo propósito. Por outro lado, todos os utilizadores realizaram as tarefas de navegação na aplicação *Android* facilmente e sem hesitações.

6.2.2 Inquérito sumário

O inquérito realizado após a realização dos testes de utilização é constituído por 4 questões, e é apresentado, junto com os resultados do mesmo, no Anexo D. Os resultados superaram as expectativas pensadas, embora, como já se referiu, a amostra seja muito limitada. Todos os avaliadores admitiram comprar um sistema como o desenvolvido, sendo que 3 dos avaliadores responderam que sim e dois responderam talvez, à questão: "Após testar o assistente virtual anterior e a aplicação desenvolvida, estaria disposto a pagar por um sistema deste género?". Os resultados à questão

procurava perceber se o assistente elaborado era capaz de substituir os sistemas de de Correio de Voz e Voz para SMS, foram exatamente os mesmos da questão anterior, 3 utilizadores responderam que sim e 2 responderam talvez. A Tabela 6.4 expõe os resultados referidos.

Tabela 6.4: Resultados importantes ao inquérito referente ao assistente virtual de camadas recebidas

Após testar o assistente virtual anterior e a aplicação desenvolvida, estaria disposto a pagar por um sistema deste género?	Já ouviu falar nas tecnologias de Correio de Voz e Voz para SMS?	Se respondeu afirmativamente à questão anterior, acha que o sistema elaborado seria uma alternativa capaz de substituir essas tecnologias?
Sim	Sim	Sim
Talvez	Sim	Talvez
Sim	Sim	Sim
Talvez	sim	Talvez
Sim	Sim	Sim

Os avaliadores referiram ainda que seria importante melhorar a organização de conteúdos na aplicação *Android*, assim como, tornar o processo de comunicação com o assistente menos robotizado, isto é, mais natural. Na verdade, este processo ainda funciona um pouco num processo de pergunta resposta com vista à obtenção de uma data para contacto posterior.

6.3 Conclusões

Concluído este capítulo que procura validar os resultados obtidos no final do desenvolvimento deste projeto de dissertação, verifica-se que os resultados não foram totalmente positivos. Efetivamente, foram realizados testes de usabilidade e inquéritos sumários para perceber até que nível os utilizadores poderiam estar satisfeitos com os assistentes virtuais criados. Contudo, a amostra é pequena e pouco diversificada, principalmente devido a questões de confidencialidade. Assim, as conclusões finais que se podem obter não serão suficientemente fortes.

Ainda assim, e através dos resultados obtidos neste capítulo, é possível verificar que o assistente virtual para chamadas recebidas teve maior recetividade na amostra do que o assistente virtual para gestão de audioconferências telefónicas.

Validação e resultados

Capítulo 7

Conclusões e trabalho futuro

Com o trabalho desenvolvido, consegui dar-se uma resposta ao principal e basilar objetivo deste trabalho. De facto, foram desenvolvidos dois sistemas independentes, tirando partido das ferramentas e dispositivos mais recentes de processamento e entendimento de linguagem natural: um que pretendia tornar o processo de gestão de audioconferências mais simples e rápido, outro cujo objetivo era dar resposta a chamadas sem a intervenção do recetor da mesma. Naturalmente, ficam ainda algumas questões em aberto e um grande potencial de trabalho e investimento futuro. Como suprarreferido, um dos objetivos desta dissertação consistia em encontrar uma forma de tornar o processo de gestão de audioconferências mais simples e rápido. Para tal, criou-se um assistente virtual para gestão de audioconferências telefónicas. Com este software, comunicando exclusivamente por voz, os utilizadores conseguem agendar conferências, ligar-se a conferências, consultar as conferências que têm agendadas e até saber informações sobre funcionários registados na empresa *Wit-Software*. Tudo isto, tirando partido de novas tecnologias de comunicação por voz, como é o caso do dispositivo *Amazon Echo Dot*, o que permitiu responder a outro dos objetivos deste trabalho: relevar uma das possíveis utilidades destes novos dispositivos desenvolvidos. Neste sentido, a utilização do *Amazon Echo Dot* constituiu uma ferramenta importante na melhoria da forma de gerir audioconferências telefónicas. Não obstante ter sido desenvolvido um assistente virtual com diversas potencialidades, existe ainda um grande trabalho a desenvolver, no sentido de melhorar as suas funcionalidades e eficiência, tal como ficou claro no Capítulo 6. De facto, foram detetadas deficiências no processo conversacional com os utilizadores, uma vez que o assistente nem sempre consegue perceber as falas dos mesmos e está limitado a um reduzido leque de falas possíveis. As interações do assistente de voz desenvolvido também ainda não são 100% eficazes, exigindo-se, portanto, um melhoramento futuro dos fluxos de interação. Por outro lado, percebeu-se que a comunicação exclusiva por voz nem sempre é uma mais-valia. Esta comunicação acelera os processos numa fase inicial, mas torna-se frustrante em certas situações, como por exemplo quando se está a iterar sobre uma lista de conferências ou utilizadores. A existência de conteúdo visual para estes casos poderia ser uma solução. Infelizmente, à data da realização

desta dissertação, o dispositivo *Amazon Echo Show* que permite, além de comunicação por voz, apresentar conteúdo visual, ainda não tinha sido lançado no mercado. Assim, outro trabalho futuro que se propõe é a integração do atual assistente virtual para gestão de audioconferências telefônicas com este dispositivo. Sabe-se também que em breve será lançada uma funcionalidade que permitirá o envio de *push notifications* por parte da *Amazon Alexa*. A integração das novidades tecnológicas para estas ferramentas no sistema atualmente desenvolvido criará valor acrescentado para o utilizador. As *push notifications* poderão, por exemplo, permitir avisar o utilizador quando estiver uma conferência prestes a começar.

Relativamente a outro grande objetivo deste trabalho (dar resposta a chamadas sem a intervenção do recetor da mesma), foi desenvolvido então um assistente virtual de chamadas recebidas, que tira partido de uma aplicação móvel. Este sistema apresentou capacidades para competir com as tecnologias existentes de Correio de Voz e Voz para SMS e respondeu a todos os objetivos propostos, como por exemplo, no momento, este *bot* é autónomo o suficiente para comunicar com interlocutores de chamadas, facilitando, entre outras coisas, o processo de criação de lembretes para quando o interlocutor estiver disponível. Por outro lado, permite consultar e reproduzir o fluxo conversacional entre o *bot* e o interlocutor para, por exemplo, contornar possíveis falhas de transcrição das ferramentas de processamento de linguagem natural. Embora a análise relativa à eficiência/utilidade deste sistema tenha sido realizada com base numa reduzida amostra, os resultados foram consonantes com as vantagens descritas. Além disso, apurou-se que esta aplicação demonstrou ser simultaneamente eficaz, intuitiva e atrativa para o utilizador. Não se conseguiu no entanto apurar se o sistema concebido supera as tecnologias existentes, em primeiro lugar, porque os testes foram realizados com uma amostra pouco significativa e, por outro lado, porque apenas foi elaborada uma simples prova de conceito com funcionalidades básicas. Acrescente-se que, apesar de o software desenvolvido facilitar o processo de criação de lembretes, sugerindo-os ao utilizador, este não os cria automaticamente.

À data de entrega desta dissertação, ainda não foi possível apurar se ambos as provas de conceito irão ser apostas futuras da empresa *Wit-Software*.

Futuramente, para além das sugestões já apresentadas, seria importante tornar as interações com os assistentes virtuais menos robóticas. Efetivamente, e esta foi uma das críticas apontada pelos avaliadores dos sistemas desenvolvidos, as falas dos assistente não transmitem emoção e tornam as conversas aborrecidas após as primeiras interações. Por outro lado, aumentar o vocabulário dos assistentes seria algo também primordial. Sugere-se também a adição de mais casos de uso a ambos os assistentes desenvolvidos para os tornar mais completos e poderosos. No caso particular do assistente virtual para chamadas recebidas, uma possível integração futura poderá passar pela partilha da localização do utilizador da aplicação a contactos específicos. Por outras palavras, o utilizador poderia selecionar um contacto que pudesse, através da comunicação com o assistente pessoal, obter a sua localização no momento da chamada.

Referências

- [AB01] Sharon A. Alvarez e Lowell W. Busenitz. The entrepreneurship of resource-based theory. *Journal of Management*, 27(6):755–775, 2001. URL: <http://dx.doi.org/10.1177/014920630102700609>, arXiv:<http://dx.doi.org/10.1177/014920630102700609>, doi:10.1177/014920630102700609.
- [ale17] Alexa, 2017. Accessed: 2017-01-17. URL: <https://developer.amazon.com/alexa>.
- [ama17a] Amazon lex – build conversation bots, 2017. Accessed: 2016-11-05. URL: <https://aws.amazon.com/lex/>.
- [ama17b] Aws lambda - serverless compute, 2017. Accessed: 2016-11-17. URL: <https://aws.amazon.com/lambda/>.
- [ama17c] Aws lambda – computação sem servidor, 2017. Accessed: 2017-03-24. URL: <https://aws.amazon.com/pt/lambda/>.
- [ama17d] Detalhes do produto amazon lex – amazon web services, 2017. Accessed: 2017-02-23. URL: <https://aws.amazon.com/pt/lex/details/>.
- [ast17] Get started, 2017. URL: <http://www.asterisk.org/get-started>.
- [Bar77] Jon Barwise. An introduction to first-order logic. In Jon Barwise, editor, *HANDBOOK OF MATHEMATICAL LOGIC*, volume 90 of *Studies in Logic and the Foundations of Mathematics*, pages 5 – 46. Elsevier, 1977. URL: <http://www.sciencedirect.com/science/article/pii/S0049237X08710978>, doi:[http://dx.doi.org/10.1016/S0049-237X\(08\)71097-8](http://dx.doi.org/10.1016/S0049-237X(08)71097-8).
- [Bor16] Debra Borchardt. Amazon says echo was the best-selling product this holiday - sells millions, Dec 2016. Accessed: 2016-11-21. URL: <http://www.forbes.com/sites/debraborchardt/2016/12/27/amazon-says-echo-was-the-best-selling-product-this-holiday-sells-millions/#7162de7163ce>.
- [BS04] Salman A Baset e Henning Schulzrinne. An analysis of the skype peer-to-peer internet telephony protocol. *arXiv preprint cs/0412017*, 2004.
- [cez17] Veesion cez. Bizagi modeler:intuitive bpmn business and it collaboration, 2017. Accessed: 2017-05-21. URL: <http://www.bizagi.com/en/how-we-help/process-modeling>.

REFERÊNCIAS

- [CH12] Erik Cambria e Amir Hussain. *Sentic computing: Techniques, tools, and applications*, volume 2. Springer Science & Business Media, 2012.
- [Cho05] Gobinda G. Chowdhury. Natural language processing. *Annual Review of Information Science and Technology*, 37(1):51–89, jan 2005. URL: <http://dx.doi.org/10.1002/aris.1440370103>, doi:10.1002/aris.1440370103.
- [CJP03] Thomas Creamer, Peeyush Jaiswal e Christopher Pavlovski. Voice-to-text reduction for real time im/chat/sms, June 25 2003. US Patent App. 10/603,495; Accessed: 2016-11-09.
- [Con17] Infinite Conferencing. Webinars, webcasting, virtual events, 2017. Accessed: 2017-05-27. URL: http://www.infiniteconferencing.com/solutions/conference_calls.php.
- [Cor17] Stanford CoreNLP – a suite of core NLP tools, 2017.
- [CW14] Erik Cambria e Bebo White. Jumping NLP curves: A review of natural language processing research [review article]. *IEEE Computational Intelligence Magazine*, 9(2):48–57, may 2014. URL: <http://dx.doi.org/10.1109/MCI.2014.2307227>, doi:10.1109/mci.2014.2307227.
- [Día05] Amparo Lasén Díaz. Understanding mobile phone users and usage. Technical report, Newbury, Reino Unido, 2005. URL: <http://eprints.ucm.es/6477/>.
- [DS76] Michael Doyle e David Straus. *How to make meetings work*. Jove Books New York, 1976.
- [Dye94] pages 389–429. Springer Nature, 1994. URL: http://dx.doi.org/10.1007/978-0-585-29599-2_12, doi:10.1007/978-0-585-29599-2_12.
- [Gib17] Samuel Gibbs. Google home review: the smart speaker that answers almost any question, May 2017. Accessed: 2017-04-18. URL: <https://www.theguardian.com/technology/2017/may/10/google-home-smart-speaker-review-voice-controlled>.
- [Gon17] Ángel González. Amazon has sold more than 11 million echo devices, morgan stanley says, Jan 2017. Accessed: 2017-01-16. URL: <http://www.seattletimes.com/business/amazon/amazon-has-sold-more-than-11-million-echo-devices-morgan-stanley-says/>.
- [goo17] Cloud speech api is now generally available, Apr 2017. Accessed: 2017-01-22. URL: <https://cloudplatform.googleblog.com/2017/04/Cloud-Speech-API-is-now-generally-available.html>.
- [heu17] Nielsen norman group, 2017. Accessed: 2017-06-17. URL: <https://www.nngroup.com/articles/ten-usability-heuristics/>.
- [How17] Amazon Lex: How It Works. <http://docs.aws.amazon.com/lex/latest/dg/how-it-works.html>, 2017. Accessed: 2017-01-22.
- [int17] Começar a trabalhar com o webex, 2017. Accessed: 2017-05-27. URL: <https://www.webex.com.br/support/getting-started.html>.

REFERÊNCIAS

- [JV15] Sheetal Jalendry e Shradha Verma. A detail review on voice over internet protocol (voip). *International Journal of Engineering Trends and Technology (IJETT)*, 23, 2015.
- [KY15] Rojanee Khummongkol e Masao Yokota. Computer simulation of mental image processing in natural language understanding by human. In *2015 IEEE 7th International Conference on Awareness Science and Technology (iCAST)*. Institute of Electrical and Electronics Engineers (IEEE), sep 2015. URL: <http://dx.doi.org/10.1109/ICAawST.2015.7314024>, doi:10.1109/icawst.2015.7314024.
- [Law15] Stephen Lawson. This 1915 conference call made history, Jan 2015. Accessed: 2017-06-13. URL: <http://www.computerworld.com/article/2874390/this-1915-conference-call-made-history.html>.
- [Lex17] Step 2: Create an Amazon Lex Bot. <http://docs.aws.amazon.com/lex/latest/dg/gs2-create-bot.html>, 2017. Accessed: 2017-01-22.
- [LKW09] Mary C Lacity, Shaji A Khan e Leslie P Willcocks. A review of the it outsourcing literature: Insights for practice. *The Journal of Strategic Information Systems*, 18(3):130–146, 2009.
- [LLD⁺17] Chengyao Lv, Huihua Liu, Yuanxing Dong, Fangyuan Li e Yuan Liang. Using uniform-design gep for part-of-speech tagging. *Journal of Circuits, Systems and Computers*, 26(04):1750060, 2017.
- [Mad07] Nitin Madnani. Getting started on natural language processing with python. *Crossroads*, 13(4):5–5, 2007.
- [McL11] Brett McLaughlin. What is node.js?, Jul 2011. Accessed: 2017-04-27. URL: <https://www.oreilly.com/ideas/what-is-node>.
- [MDC07] Ruy Luiz Milidiú, Julio Cesar Duarte e Roberto Cavalcante. Machine learning algorithms for portuguese named entity recognition. *Inteligencia Artificial, Revista Iberoamericana de Inteligencia Artificial*, 11(36):67–75, 2007.
- [MSB⁺14] Christopher D Manning, Mihai Surdeanu, John Bauer, Jenny Rose Finkel, Steven Bethard e David McClosky. The stanford corenlp natural language processing toolkit. In *ACL (System Demonstrations)*, pages 55–60, 2014.
- [Pet03] Thomas D Petite. System and method for communicating with a remote communication unit via the public switched telephone network (pstn), September 9 2003. US Patent 6,618,578; Accessed: 2016-11-17.
- [PGK05] Martha Palmer, Daniel Gildea e Paul Kingsbury. The proposition bank: An annotated corpus of semantic roles. *Computational linguistics*, 31(1):71–106, 2005.
- [Rod17] Jesus Rodriguez. Aws lex can be a strong contender in the conversational platform space, Jan 2017. Accessed: 2017-03-05. URL: <https://medium.com/@jrodthoughts/aws-lex-can-be-a-strong-contender-in-the-conversational-platform-space-9qbbfzxjk>.

REFERÊNCIAS

- [RSC⁺02] Jonathan Rosenberg, Henning Schulzrinne, Gonzalo Camarillo, Alan Johnston, Jon Peterson, Robert Sparks, Mark Handley e Eve Schooler. Sip: session initiation protocol. Technical report, 2002.
- [SC14] Rahul Singh e Ritu Chauhan. A review paper: Voice over internet protocol. *International Journal of Enhanced Research in Management & Computer Applications*, ISSN, pages 2319–7471, 2014.
- [SJ08] Henry Sinnreich e Alan B. Johnston. *Internet Communications Using SIP: Delivering VoIP and Multimedia Services with Session Initiation Protocol (Networking Council)*. Wiley, 2008. URL: <https://www.amazon.com/Internet-Communications-Using-SIP-Delivering-ebook/dp/B00805JR20%3FSubscriptionId%3D0JYN1NVW651KCA56C102%26tag%3Dtechkie-20%26linkCode%3Dxm2%26camp%3D2025%26creative%3D165953%26creativeASIN%3DB00805JR20>.
- [SJ12] Henry Sinnreich e Alan B Johnston. *Internet communications using SIP: Delivering VoIP and multimedia services with Session Initiation Protocol*, volume 27. John Wiley & Sons, 2012.
- [SLC17] Shiliang Sun, Chen Luo e Junyu Chen. A review of natural language processing techniques for opinion mining systems. *Information Fusion*, 36:10 – 25, 2017. URL: [//www.sciencedirect.com/science/article/pii/S1566253516301117](http://www.sciencedirect.com/science/article/pii/S1566253516301117), doi:<http://dx.doi.org/10.1016/j.inffus.2016.10.004>.
- [SS13] Muhammad Sarwar e Tariq Rahim Soomro. Impact of smartphone’s on society. *European Journal of Scientific Research*, 98(2):216–226, 2013.
- [sta16] Stanford corenlp, 2016. Accessed: 2016-11-03. URL: <http://stanfordnlp.github.io/CoreNLP/memory-time.html>.
- [tel17] Phone conference controls, 2017. Accessed: 2017-06-05. URL: <https://www.conferencecalling.com/telephone.html>.
- [try17] Building a chatbot: analysis and limitations of modern platforms, Jan 2017. Accessed: 2017-04-14. URL: <https://tryolabs.com/blog/2017/01/25/building-a-chatbot-analysis--limitations-of-modern-platforms/?-1>.
- [twi17a] Twilio, 2017. Accessed: 2016-11-21. URL: <https://www.twilio.com/voice/sip-interface>.
- [twi17b] Twiml: The twilio markup language, 2017. Accessed: 2017-04-03. URL: <https://www.twilio.com/docs/api/twiml>.
- [Usa16] Usability heuristics for bots – chatbots magazine, May 2016. Accessed: 2017-06-14. URL: <https://chatbotsmagazine.com/usability-heuristics-for-bots-7075132d2c92>.
- [Vin17] James Vincent. Amazon officially unveils touchscreen echo show, May 2017. Accessed: 2017-06-06. URL: <https://www.theverge.com/circuitbreaker/2017/5/9/15590978/amazon-echo-show-touchscreen-price-release-date>.

REFERÊNCIAS

- [voc12] Hatsune miku, 2012. URL: http://vocaloid.wikia.com/wiki/Hatsune_Miku.
- [vod14] Termos e condições do serviço vodafone voz para sms. <http://www.vodafone.pt/main/Ajuda/termoscondicoes/vodafone-voz-sms.htm>, 2014. Accessed: 2017-01-28.
- [Wat17a] Peter Watson. What is a conference call - learn how a conference call works, 2017. Accessed: 2017-05-25. URL: <https://daisygroup.com/resource/what-is-a-conference-call/>.
- [Wat17b] Peter Watson. What is a conference call? the rise of conference call technology, 2017. Accessed: 2017-05-25. URL: <https://daisygroup.com/resource/what-is-a-conference-call-the-rise-of-conference-call-technology/>.
- [wel17] Welcome to API.AI! <https://console.api.ai/api-client/#/getStarted>, 2017. Accessed: 2017-01-22.
- [wha17] About whatsapp, 2017. Accessed: 2017-01-06. URL: <https://www.whatsapp.com/about/>.

REFERÊNCIAS

Anexo A

Interaction Model - Alexa Skill

```
1 {
2   "intents": [
3     {
4       "name": "AMAZON.CancelIntent",
5       "samples": []
6     },
7     {
8       "name": "AMAZON.HelpIntent",
9       "samples": []
10    },
11    {
12      "name": "AMAZON.StopIntent",
13      "samples": []
14    },
15    {
16      "name": "CH_CONFERENCEPLIES",
17      "samples": [],
18      "slots": []
19    },
20    {
21      "name": "CH_DIALMEIN",
22      "samples": [
23        "Call me to the conference number {CH_DIALMEINP}"
24      ],
25      "slots": [
26        {
27          "name": "CH_DIALMEINP",
28          "type": "AMAZON.NUMBER",
29          "samples": []
30        }
31      ]
32    },
33  ]
}
```

Interaction Model - Alexa Skill

```
34     "name": "CH_HELP",
35     "samples": [
36         "help"
37     ],
38     "slots": []
39 },
40 {
41     "name": "CH_MUTE",
42     "samples": [
43         "mute {contact}"
44     ],
45     "slots": [
46         {
47             "name": "contact",
48             "type": "CONTACTS",
49             "samples": []
50         }
51     ]
52 },
53 {
54     "name": "chitChat",
55     "samples": [
56         "{do text|command}",
57         "{how text|command}"
58     ],
59     "slots": [
60         {
61             "name": "command",
62             "type": "Literal",
63             "samples": []
64         }
65     ]
66 },
67 {
68     "name": "CONFERENCE_GUESTS",
69     "samples": [
70         "guests for this conference",
71         "conference guests"
72     ],
73     "slots": []
74 },
75 {
76     "name": "CONFERENCE_PARTICIPANTS",
77     "samples": [
78         "conference participants"
79     ],
80     "slots": []
81 },
82 {
```

Iteration Model - Alexa Skill

```
83     "name": "CONFGEST",
84     "samples": [
85         "{conference}",
86         "{conference} one",
87         "{conference} please",
88         "{conference} thanks",
89         "show me the {conference}"
90     ],
91     "slots": [
92         {
93             "name": "conference",
94             "type": "TEST",
95             "samples": []
96         }
97     ]
98 },
99 {
100     "name": "CONTACTS_INTENT_TEST",
101     "samples": [
102         "{contact}",
103         "add {contact}"
104     ],
105     "slots": [
106         {
107             "name": "contact",
108             "type": "CONTACTS",
109             "samples": []
110         }
111     ]
112 },
113 {
114     "name": "HOURSNUMBERS",
115     "samples": [
116         "{hours}",
117         "{hours} hours"
118     ],
119     "slots": [
120         {
121             "name": "hours",
122             "type": "AMAZON.NUMBER",
123             "samples": []
124         }
125     ]
126 },
127 {
128     "name": "LOOKCONFERENCES",
129     "samples": [
130         "next conferences",
131         "show me conferences"
```

Interaction Model - Alexa Skill

```
132     ],
133     "slots": []
134   },
135   {
136     "name": "SCHEDULE_CONFERENCE_WITH_DAY",
137     "samples": [
138       "schedule a conference for {weekday}"
139     ],
140     "slots": [
141       {
142         "name": "weekday",
143         "type": "WEEK_DAYS",
144         "samples": []
145       }
146     ]
147   },
148   {
149     "name": "SCHEDULE_CONFERENCE_WITH_DAY_HOUR",
150     "samples": [
151       "schedule a conference for {weekday} at {hour} "
152     ],
153     "slots": [
154       {
155         "name": "weekday",
156         "type": "WEEK_DAYS",
157         "samples": []
158       },
159       {
160         "name": "hour",
161         "type": "AMAZON.NUMBER",
162         "samples": []
163       }
164     ]
165   },
166   {
167     "name": "START_CONFERENCE_SCHEDULER",
168     "samples": [
169       "schedule conference",
170       "arrange a conference",
171       "plan a conference"
172     ],
173     "slots": []
174   },
175   {
176     "name": "VA_LISTTODAYCONFERENCES",
177     "samples": [
178       "conferences for today",
179       "show my today's conferences",
180       "what are my conferences for today"
```

Iteration Model - Alexa Skill

```
181     ],
182     "slots": []
183   },
184   {
185     "name": "VA_OUTBOUNDCALL",
186     "samples": [
187       "Call {contact} to this conference"
188     ],
189     "slots": [
190       {
191         "name": "contact",
192         "type": "CONTACTS",
193         "samples": []
194       }
195     ]
196   },
197   {
198     "name": "VA_SOFTPHONE_SETTINGS",
199     "samples": [
200       "my softphone settings",
201       "softphone settings"
202     ],
203     "slots": []
204   },
205   {
206     "name": "VA_WhoIs",
207     "samples": [
208       "Who is {contact}"
209     ],
210     "slots": [
211       {
212         "name": "contact",
213         "type": "CONTACTS",
214         "samples": []
215       }
216     ]
217   },
218   {
219     "name": "WEEKDAYS",
220     "samples": [
221       "{weekday}"
222     ],
223     "slots": [
224       {
225         "name": "weekday",
226         "type": "WEEK_DAYS",
227         "samples": []
228       }
229     ]
230   }
231 ]
```

Interaction Model - Alexa Skill

```
230     }
231 ],
232 "types": [
233   {
234     "name": "CONTACTS",
235     "values": [
236       {
237         "name": {
238           "value": "Tiago"
239         }
240       },
241       {
242         "name": {
243           "value": "Tiago Ferreira"
244         }
245       },
246       {
247         "name": {
248           "value": "Paulo Sousa"
249         }
250       },
251       {
252         "name": {
253           "value": "Paulo"
254         }
255       },
256       {
257         "name": {
258           "value": "Jorge"
259         }
260       },
261       {
262         "name": {
263           "value": "Jorge Sousa"
264         }
265       },
266       {
267         "name": {
268           "value": "Joao Costa"
269         }
270       },
271       {
272         "name": {
273           "value": "Joao"
274         }
275       },
276       {
277         "name": {
278           "value": "Joao "
```

Iteration Model - Alexa Skill

```
279     }
280   },
281   {
282     "name": {
283       "value": "Joao Costa"
284     }
285   },
286   {
287     "name": {
288       "value": "Ricardo"
289     }
290   },
291   {
292     "name": {
293       "value": "Ricardo Loureiro"
294     }
295   }
296 ]
297 },
298 {
299   "name": "TEST",
300   "values": [
301     {
302       "name": {
303         "value": "test"
304       }
305     },
306     {
307       "name": {
308         "value": "add"
309       }
310     },
311     {
312       "name": {
313         "value": "done"
314       }
315     },
316     {
317       "name": {
318         "value": "yes"
319       }
320     },
321     {
322       "name": {
323         "value": "no"
324       }
325     },
326     {
327       "name": {
```

Interaction Model - Alexa Skill

```
328         "value": "today"
329     }
330 },
331 {
332     "name": {
333         "value": "another day"
334     }
335 },
336 {
337     "name": {
338         "value": "next"
339     }
340 },
341 {
342     "name": {
343         "value": "details"
344     }
345 },
346 {
347     "name": {
348         "value": "next week"
349     }
350 },
351 {
352     "name": {
353         "value": "cancel"
354     }
355 },
356 {
357     "name": {
358         "value": "stop"
359     }
360 },
361 {
362     "name": {
363         "value": "call"
364     }
365 },
366 {
367     "name": {
368         "value": "same"
369     }
370 }
371 ]
372 },
373 {
374     "name": "WEEK_DAYS",
375     "values": [
376     {
```


Iteration Model - Alexa Skill

```
377     "name": {
378         "value": "Monday"
379     }
380 },
381 {
382     "name": {
383         "value": "Tuesday"
384     }
385 },
386 {
387     "name": {
388         "value": "Wednesday"
389     }
390 },
391 {
392     "name": {
393         "value": "Thursday"
394     }
395 },
396 {
397     "name": {
398         "value": "Friday"
399     }
400 },
401 {
402     "name": {
403         "value": "Saturday"
404     }
405 },
406 {
407     "name": {
408         "value": "Sunday"
409     }
410 },
411 {
412     "name": {
413         "value": "Today"
414     }
415 }
416 ]
417 }
418 ]
419 }
```

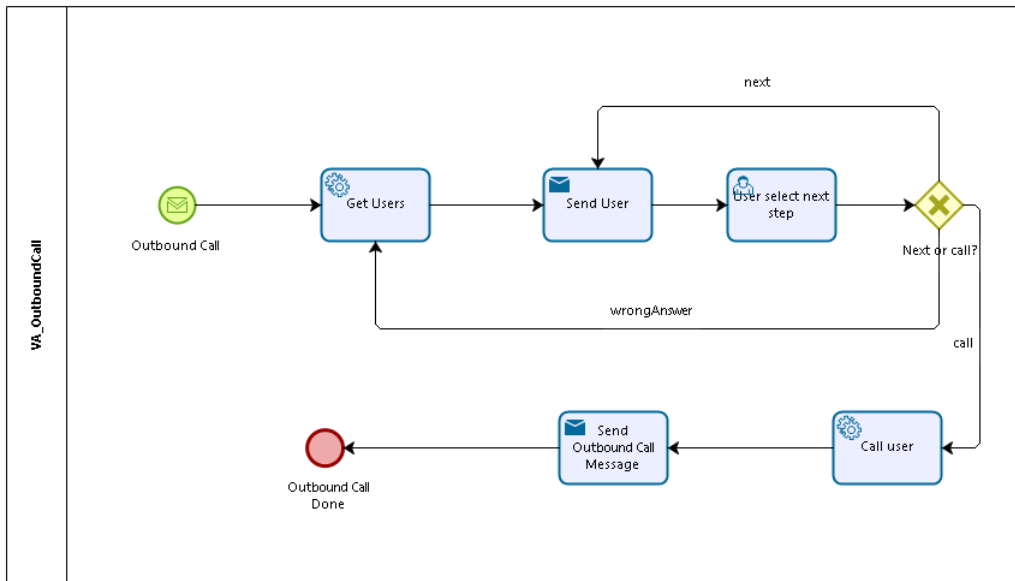
Interaction Model - Alexa Skill

Anexo B

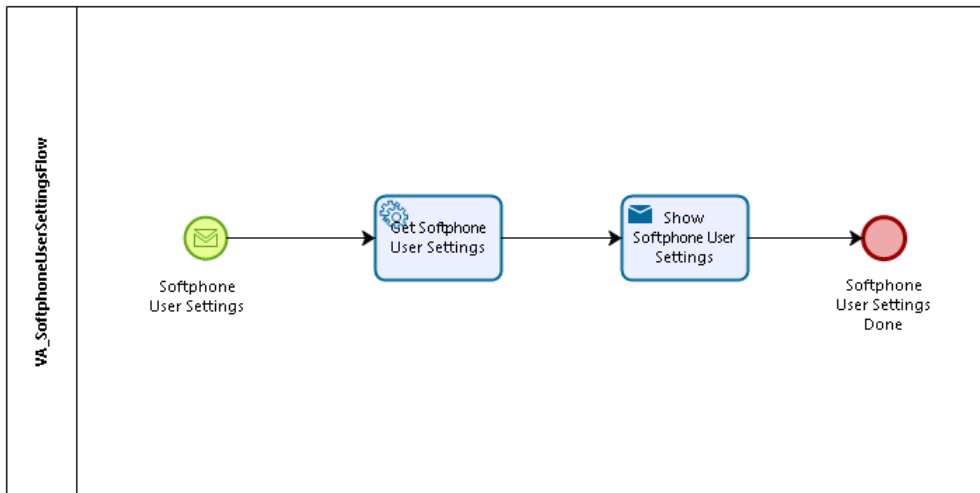
Fluxos - Bizagi Modeler

Neste anexo demonstram-se todos os fluxos usados no sistema que foi criado, é feita uma separação entre os fluxos criados de raiz, os fluxos adaptados e os fluxos que se mantiveram inalterados. De qualquer maneira, mesmo os fluxos que se mantiveram, o código associado a esses mesmo fluxos foi adaptado para funcionar com a *Amazon Alexa*. Como foi referido no Capítulo 4, alguns fluxos já estavam em funcionamento num assistente virtual que comunica com o *Facebook Messenger*.

B.1 Fluxos criados de raiz

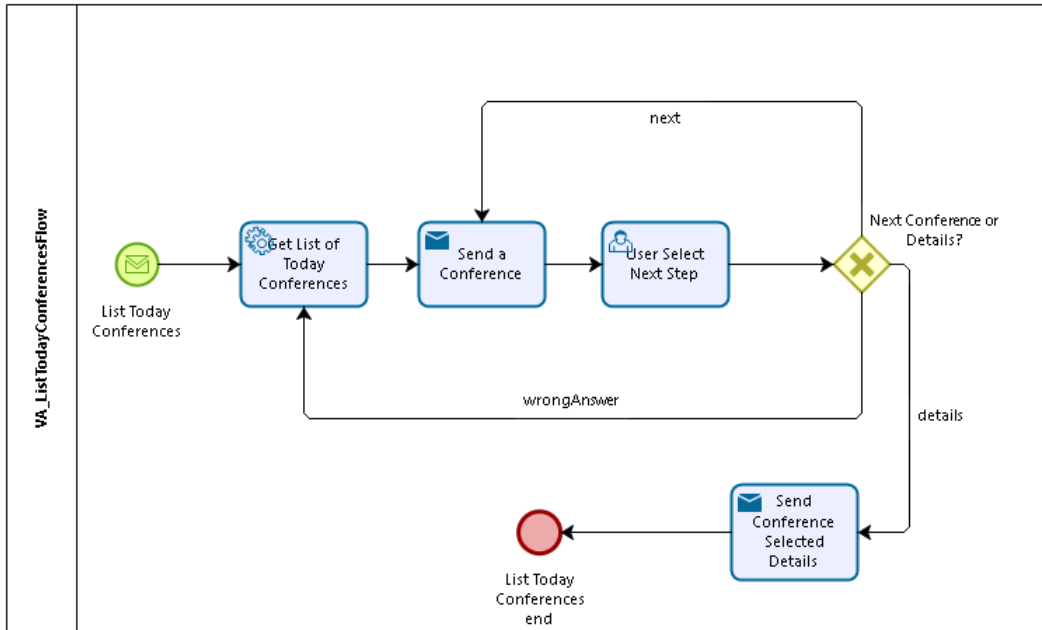


Powered by
bizagi
Modeler

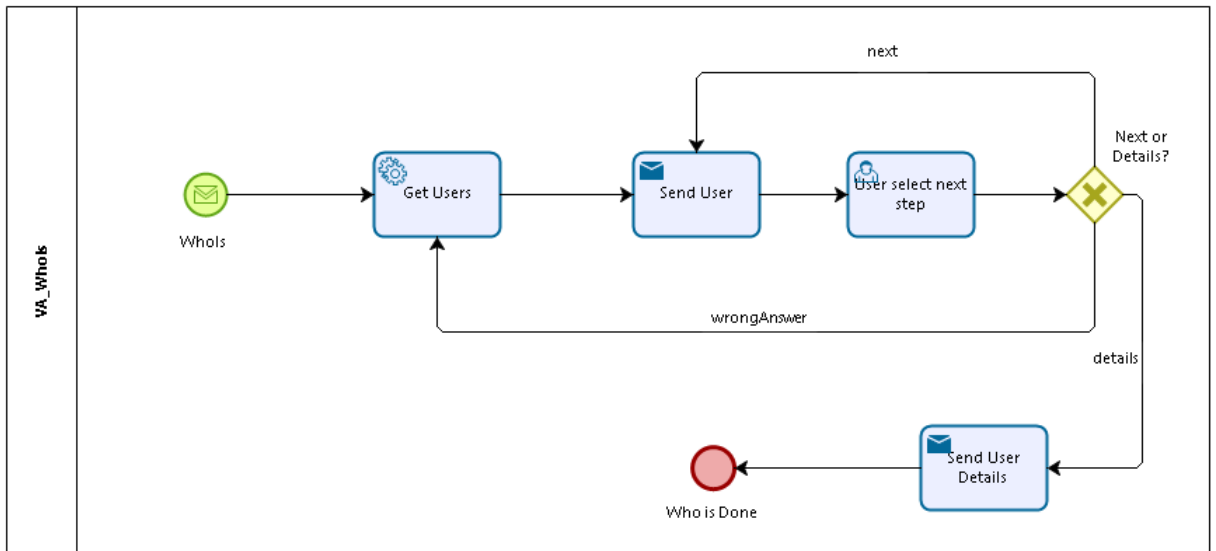


Powered by
bizagi
Modeler

Fluxos - Bizagi Modeler

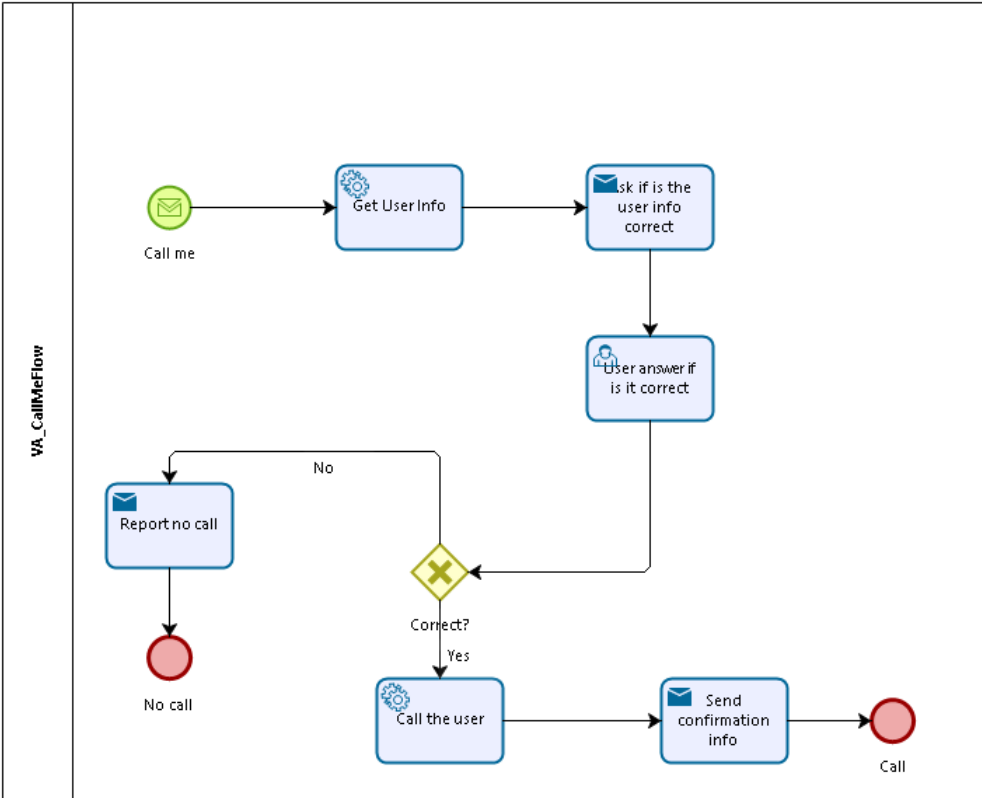


Powered by
bizagi
Modeler

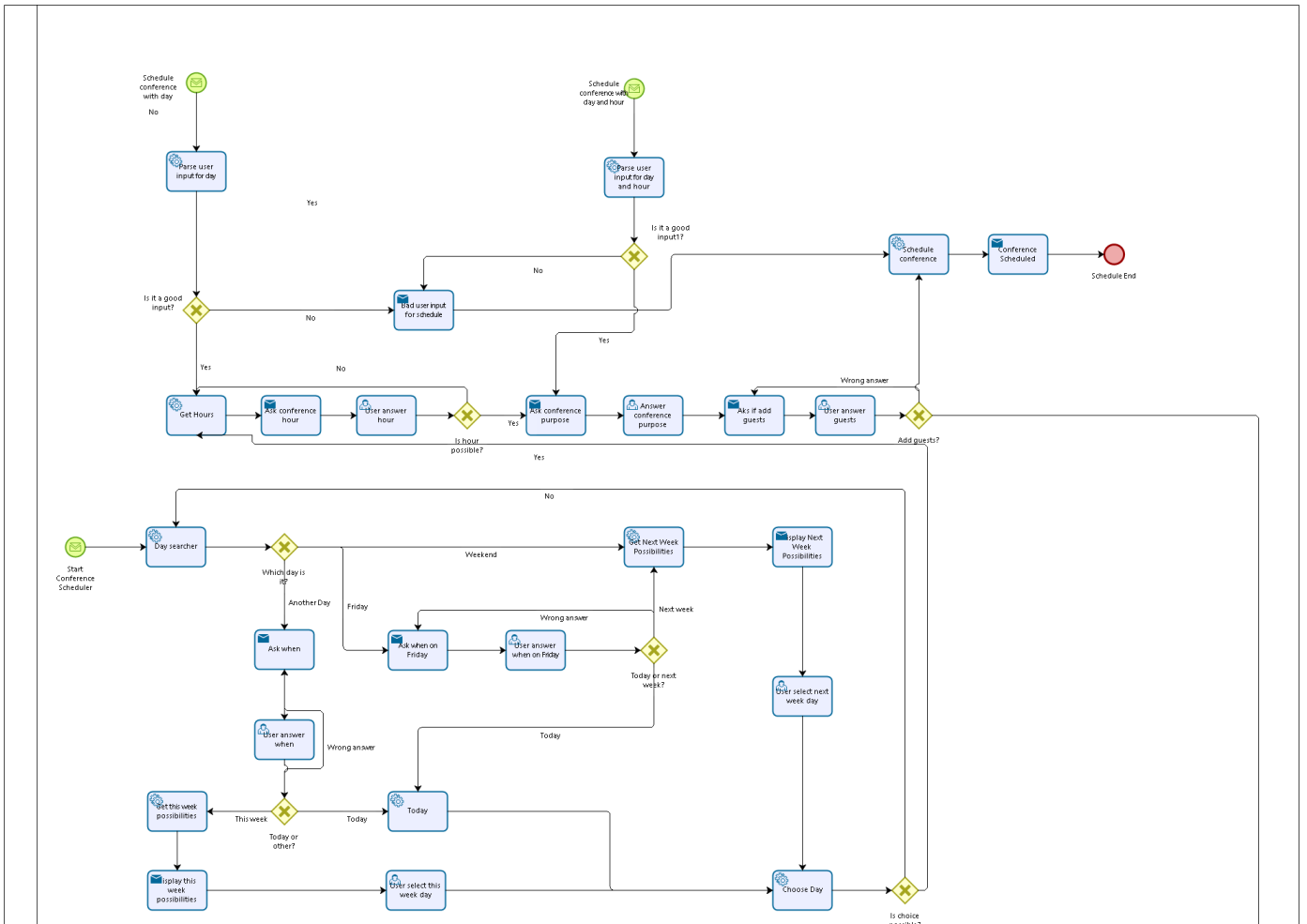


Powered by
bizagi
Modeler

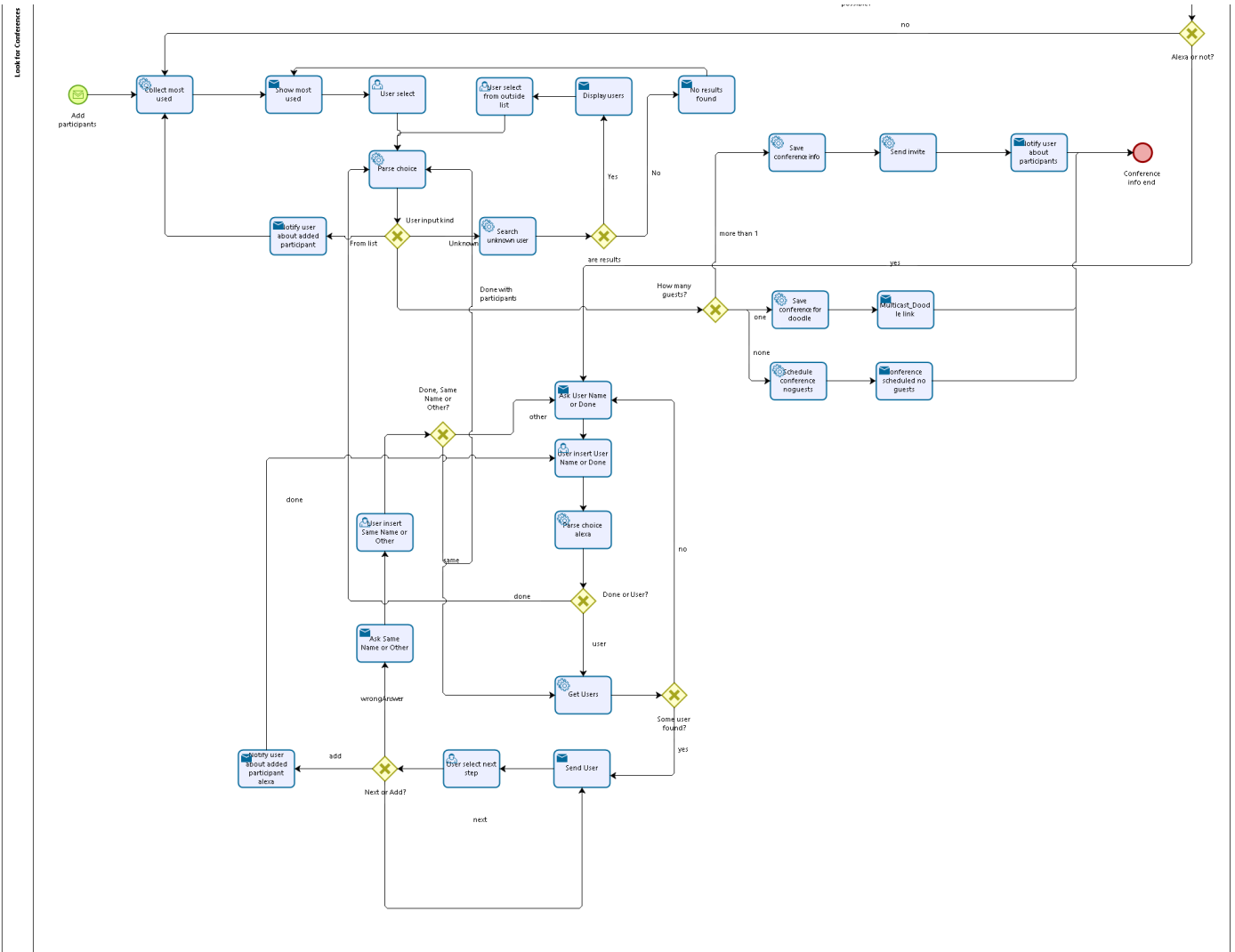
Fluxos - Bizagi Modeler



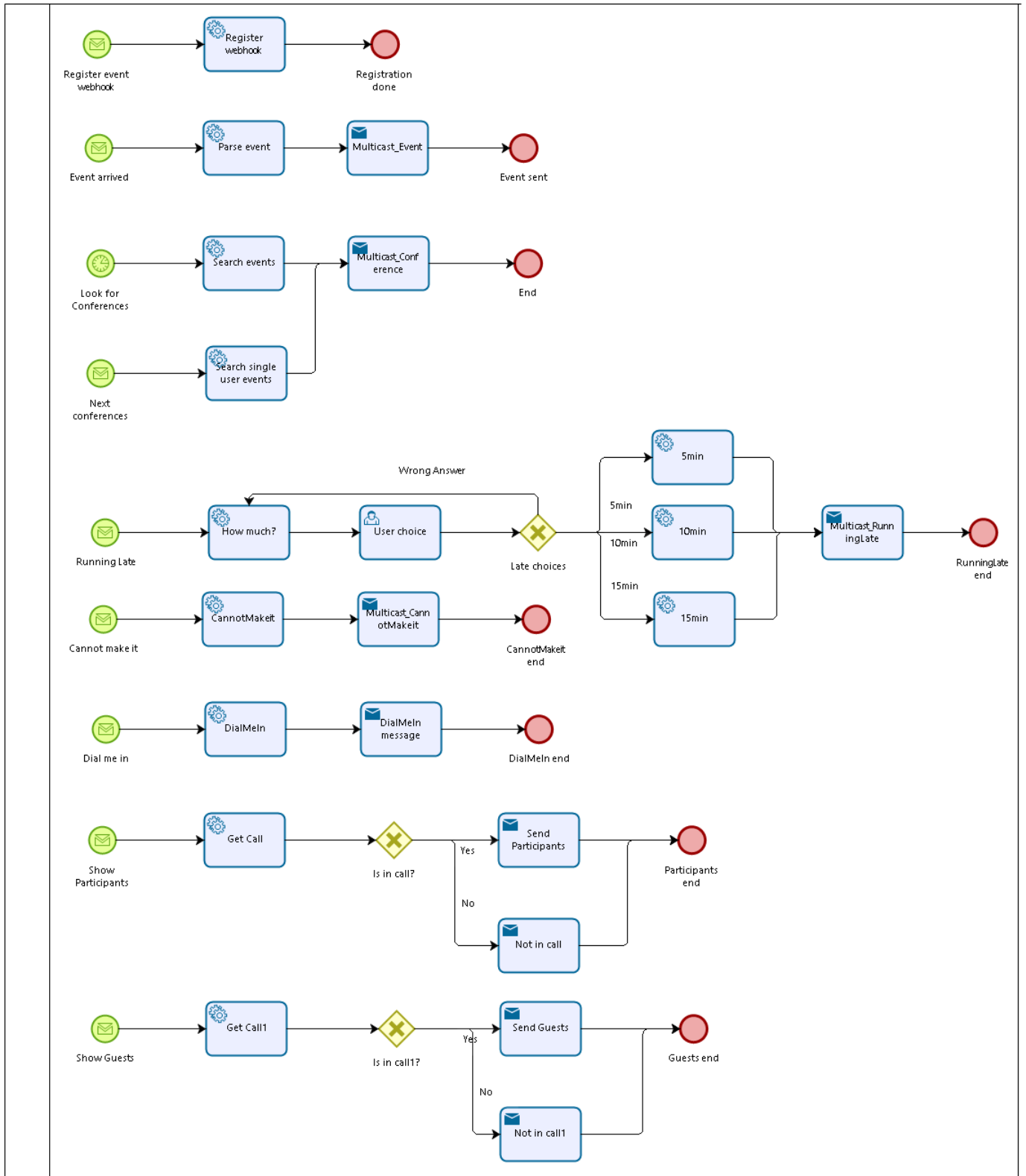
B.2 Fluxos adaptados



Fluxos - Bizagi Modeler

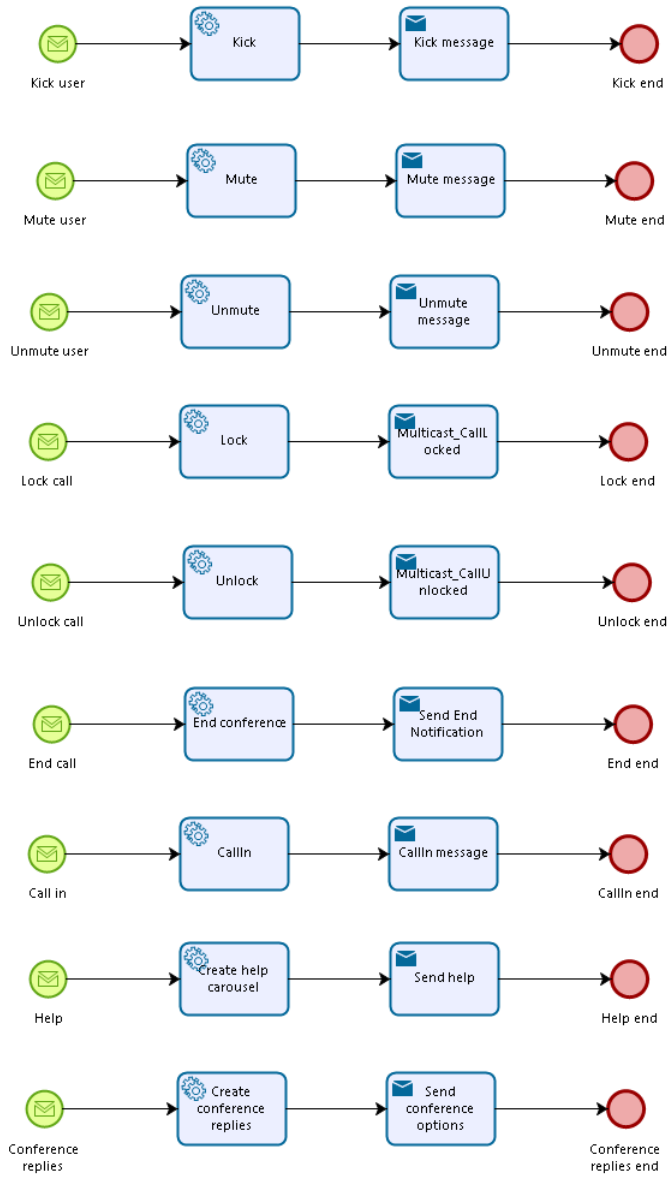


B.3 Fluxos sem alterações

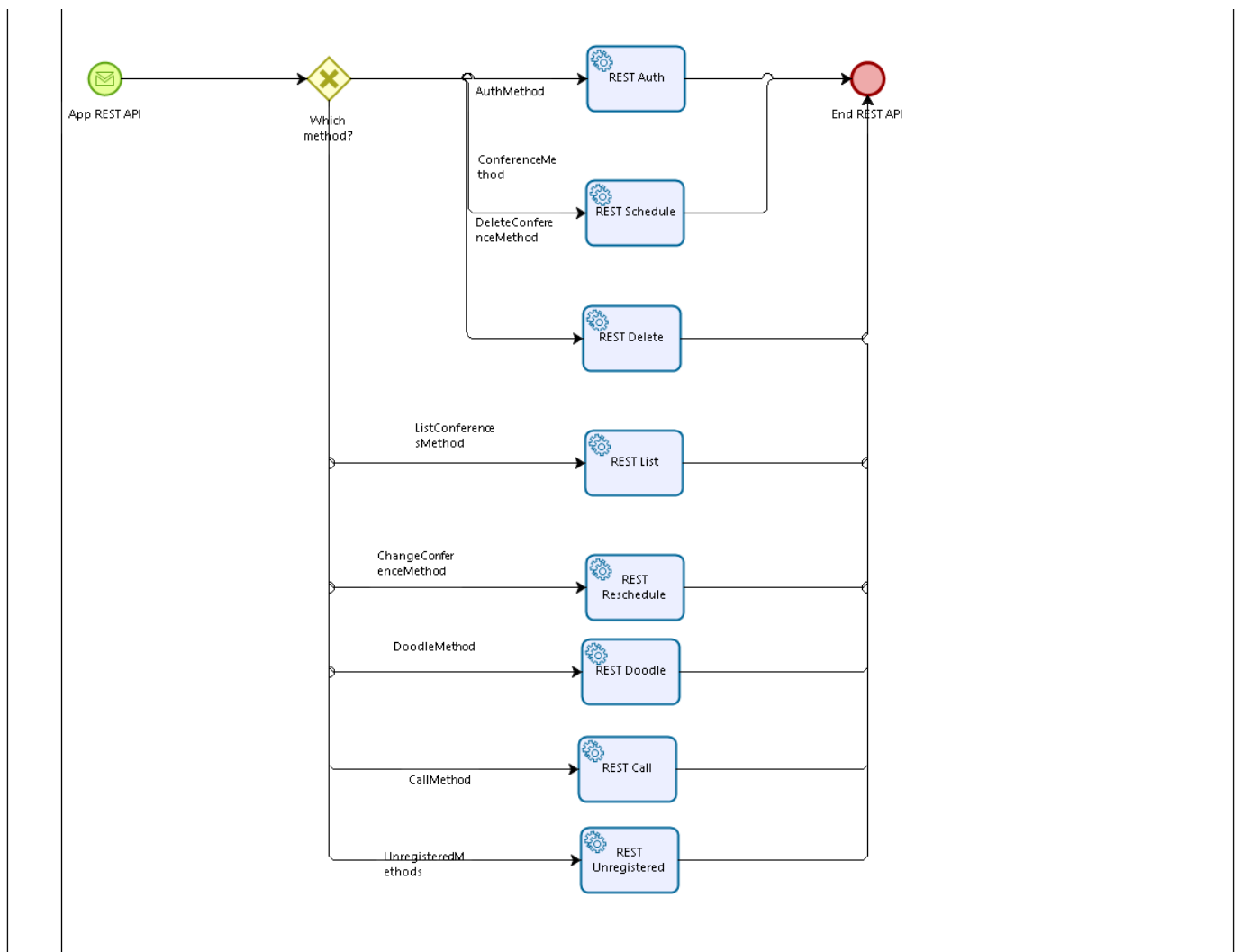


Fluxos - Bizagi Modeler

Look for Conferences



Fluxos - Bizagi Modeler



Fluxos - Bizagi Modeler

Anexo C

Inquérito introdutório - Assistente virtual de chamadas recebidas

C.1 Inquérito

Assistente Virtual para comunicação empresarial

No âmbito do projeto de dissertação "Assistentes Virtuais para comunicação empresarial", do mestrado em Engenharia Informática e Computação, desenvolveu-se o seguinte questionário. O objetivo deste questionário é avaliar o interesse da utilização de um sistema alternativo aos sistemas atuais, que ajude a gerir as chamadas perdidas por um utilizador que se encontra ocupado.

Agradeço desde já a colaboração, para alguma questão contacte: joao.carlos.costa@fe.up.pt

*Obrigatório

Idade *

- Menos de 18 anos de idade
- 18 a 25 anos de idade
- 26 a 40 anos de idade
- 41 a 60 anos de idade
- 61 ou mais anos de idade

Já ouviu falar em correio de voz ou no sistema voz para SMS, ambas ferramentas que oferecem soluções para chamadas perdidas? *

- Sim
- Não

Qual ou quais das tecnologias já teve oportunidade de utilizar? *

- Correio de voz
- Voz para SMS
- Não aplicável

Inquérito introdutório - Assistente virtual de chamadas recebidas

Em que medida ficou satisfeito com a sua qualidade de serviço?

- Nada satisfeito
- Pouco satisfeito
- Satisfeito
- Muito Satisfeito

Na altura de contactar alguém, opta por realizar uma chamada nativa, isto é, uma chamada regular através da rede pública de telefonia comutada, ou liga através de aplicações VoIP, como o Skype ou o WhatsApp? *

- Chamada nativa
- Aplicações VoIP

Consideraria a utilização de um assistente virtual que comunicaria com a pessoa que o contactou a fim de averiguar o motivo do contacto e, por exemplo, criar automaticamente um lembrete na sua agenda numa hora em estivesse disponível e acordada com o outro utilizador? *

- Sim
- Talvez
- Não

Se respondeu negativamente à questão anterior, apresente um motivo para ter escolhido essa opção

A sua resposta

SUBMITER

C.2 Respostas

Carimbo de data/hora	Idade	Ja ouviu falar em correio de voz ou no sistema voz para SMS, ambas ferramentas que oferecem soluções para chamadas perdidas?	Qual ou quais das tecnologias já teve oportunidade de utilizar?	Em que medida ficou satisfeito com o sua qualidade de serviço?	Na altura de contactar alguém, opta por realizar uma chamada nativa, isto é, uma chamada regular através da rede publica de telefonia comutada ou liga através de aplicações VoIP, como o Skype ou o WhatsApp?	Consideraria a utilização de um assistente virtual que comunicaria com a pessoa que o contactou a fim de averiguar o motivo do contacto e, por exemplo, criar automaticamente um lembrete na sua agenda numa hora em estivesse disponível e acordada com o outro utilizador?	Se respondeu negativamente à questão anterior, apresente um motivo para ter escolhido essa opção	Estaria disposto a pagar por um sistema como o assistente virtual apresentado?
6/6/2017 12:40:27	18 a 25 anos de idade	Sim	Correio de voz, Voz para SMS	Nada satisfeito	Chamada nativa	Sim		Talvez
6/6/2017 12:47:11	18 a 25 anos de idade	Sim	Correio de voz	Satisfeito	Chamada nativa	Sim		Sim
6/6/2017 12:58:15	18 a 25 anos de idade	Sim	Correio de voz, Voz para SMS	Satisfeito	Aplicações Voip	Talvez		Talvez
6/6/2017 13:01:15	26 a 40 anos de idade	Sim	Correio de voz	Pouco satisfeito	Chamada nativa	Talvez		Não
6/6/2017 13:01:34	18 a 25 anos de idade	Sim	Correio de voz, Voz para SMS	Pouco satisfeito	Chamada nativa	Sim		Talvez
6/6/2017 13:02:18	18 a 25 anos de idade	Sim	Não aplicável	Satisfeito	Chamada nativa	Não	A conversa é entre pessoas, não entre máquinas	Não
6/6/2017 13:02:32	26 a 40 anos de idade	Não	Não aplicável		Chamada nativa	Sim		Talvez
6/6/2017 13:07:19	18 a 25 anos de idade	Não	Não aplicável	Nada satisfeito	Chamada nativa	Não		Não
6/6/2017 13:08:35	18 a 25 anos de idade	Sim	Voz para SMS	Pouco satisfeito	Chamada nativa	Não	Não considero que me traga qualquer vantagem prática. Nota: Escreve-se "overiguar" e não "oviriguar", tal como "automaticamente" e não "automaticamente". Somos todos engenheiros, mas não custa saber a diferença.	Não
6/6/2017 13:09:03	26 a 40 anos de idade	Sim	Correio de voz	Satisfeito	Chamada nativa	Talvez		Talvez
6/6/2017 13:09:46	18 a 25 anos de idade	Sim	Não aplicável		Chamada nativa	Talvez		Não
6/6/2017 13:12:18	18 a 25 anos de idade	Sim	Correio de voz, Voz para SMS	Nada satisfeito	Chamada nativa	Sim		Não
6/6/2017 13:12:49	26 a 40 anos de idade	Sim	Correio de voz	Pouco satisfeito	Aplicações Voip	Talvez		Talvez
6/6/2017 13:14:31	18 a 25 anos de idade	Sim	Correio de voz, Voz para SMS	Satisfeito	Chamada nativa	Sim		Talvez
6/6/2017 13:15:26	18 a 25 anos de idade	Sim	Correio de voz, Voz para SMS	Pouco satisfeito	Chamada nativa	Talvez		Não
6/6/2017 13:15:37	26 a 40 anos de idade	Não	Correio de voz	Satisfeito	Chamada nativa	Talvez	Depende se o assistente necessitaria de algum custo adicional e qual esse custo.	Talvez
6/6/2017 13:16:04	18 a 25 anos de idade	Sim	Correio de voz, Voz para SMS	Pouco satisfeito	Chamada nativa	Talvez		Não

Inquérito introdutório - Assistente virtual de chamadas recebidas

Corrimão de data/hora	Idade	Já ouviu falar em correio de voz ou no sistema voz para SMS, ambas ferramentas que oferecem soluções para chamadas perdidas?	Qual ou quais das tecnologias já teve oportunidade de utilizar?	Em que medida ficou satisfeito com a sua qualidade de serviço?	Na altura de contactar alguém, opta por realizar uma chamada nativa, isto é, uma chamada regular através da rede pública de telefonia comutada, ou liga através de aplicações VoIP, como o Skype ou o WhatsApp?	Consideraria a utilização de um assistente virtual que comunicaria com a pessoa que a contactou a fim de averiguar o motivo do contacto e, por exemplo, criar automaticamente um lembrete no seu agenda numa hora em estivesse disponível e acordada com o outro utilizador?	Se respondeu negativamente à questão anterior, apresente um motivo para ter escolhido essa opção	Estaria disposto a pagar por um sistema como o assistente virtual apresentado?
6/6/2017 13:16:54	18 a 25 anos de idade	Sim	Correio de voz	Satisfeito	Chamada nativa	Não	Podem ser utilizados para fazer reverse engineering do horário do utilizador	Não
6/6/2017 13:17:42	18 a 25 anos de idade	Sim	Correio de voz, Voz para SMS	Satisfeito	Chamada nativa	Sim		Talvez
6/6/2017 13:18:31	18 a 25 anos de idade	Não	Não aplicável		Chamada nativa	Não		Não
6/6/2017 13:18:41	18 a 25 anos de idade	Sim	Correio de voz	Satisfeito	Chamada nativa	Sim		Não
6/6/2017 13:20:13	18 a 25 anos de idade	Sim	Correio de voz	Satisfeito	Chamada nativa	Sim	Mais prático e útil	Não
6/6/2017 13:20:57	18 a 25 anos de idade	Não	Não aplicável		Chamada nativa	Talvez		Talvez
6/6/2017 13:22:12	18 a 25 anos de idade	Sim	Correio de voz, Voz para SMS	Pouco satisfeito	Chamada nativa	Talvez	Acho confuso	Não
6/6/2017 13:22:47	18 a 25 anos de idade	Sim	Correio de voz	Satisfeito	Chamada nativa	Talvez		Talvez
6/6/2017 13:24:34	26 a 40 anos de idade	Sim	Correio de voz	Satisfeito	Chamada nativa	Talvez		Talvez
6/6/2017 13:26:51	26 a 40 anos de idade	Sim	Correio de voz	Satisfeito	Chamada nativa	Sim		Sim
6/6/2017 13:27:49	18 a 25 anos de idade	Sim	Não aplicável	Nada satisfeito	Chamada nativa	Talvez		Talvez
6/6/2017 13:31:19	18 a 25 anos de idade	Sim	Correio de voz	Satisfeito	Chamada nativa	Talvez		Não
6/6/2017 13:32:30	26 a 40 anos de idade	Sim	Correio de voz	Muito Satisfeito	Chamada nativa	Sim		Talvez
6/6/2017 13:33:14	18 a 25 anos de idade	Sim	Correio de voz	Satisfeito	Aplicações VoIP	Talvez		Não
6/6/2017 13:34:07	18 a 25 anos de idade	Sim	Correio de voz	Satisfeito	Chamada nativa	Talvez		Não
6/6/2017 13:34:37	18 a 25 anos de idade	Sim	Correio de voz, Voz para SMS	Satisfeito	Chamada nativa	Talvez		Não
6/6/2017 13:36:20	18 a 25 anos de idade	Não	Não aplicável		Chamada nativa	Sim		Talvez
6/6/2017 13:40:19	18 a 25 anos de idade	Sim	Correio de voz, Voz para SMS	Satisfeito	Chamada nativa	Talvez		Não
6/6/2017 13:41:27	18 a 25 anos de idade	Sim	Voz para SMS, Não aplicável	Pouco satisfeito	Chamada nativa	Sim		Não
6/6/2017 13:41:35	18 a 25 anos de idade	Sim	Correio de voz	Pouco satisfeito	Aplicações VoIP	Talvez		Talvez
6/6/2017 13:41:58	18 a 25 anos de idade	Sim	Correio de voz	Pouco satisfeito	Chamada nativa	Talvez		Não
6/6/2017 13:45:14	18 a 25 anos de idade	Sim	Não aplicável		Chamada nativa	Não		Não
6/6/2017 13:45:27	18 a 25 anos de idade	Sim	Voz para SMS	Pouco satisfeito	Chamada nativa	Talvez		Não
6/6/2017 13:45:36	18 a 25 anos de idade	Sim	Correio de voz	Satisfeito	Chamada nativa	Sim		Não

Inquérito introdutório - Assistente virtual de chamadas recebidas

Carimbo de data/hora	Idade	Já ouviu falar em correio de voz ou no sistema voz para SMS, ambas ferramentas que oferecem soluções para chamadas perdidas?	Qual ou quais das tecnologias já teve oportunidade de utilizar?	Em que medida ficou satisfeito com a sua qualidade de serviço?	No altura de contactar alguém, opta por realizar uma chamada nativa, isto é, uma chamada regular através da rede pública de telefonia comutada, ou liga através de aplicações VoIP, como o Skype ou o WhatsApp?	Consideraria a utilização de um assistente virtual que comunicaria com a pessoa que o contactou a fim de averiguar o motivo do contacto e, por exemplo, criar automaticamente um lembrete na sua agenda numa hora em estivesse disponível e acordada com o outro utilizador?	Se respondeu negativamente à questão anterior, apresente um motivo para ter escolhido essa opção	Estaria disposto a pagar por um sistema como o assistente virtual apresentado?
6/6/2017 13:47:44	18 a 25 anos de idade	Não	Voz para SMS	Pouco satisfeito	Chamada nativa	Talvez	Acho que é desnecessário. Se a pessoa não pode atender devolve a chamada quando poder ou então nós próprios ligamos mais tarde sem ter que ter mais uma aplicação inútil no telemóvel.	Não
6/6/2017 13:51:34	18 a 25 anos de idade	Sim	Correio de voz	Satisfeito	Chamada nativa	Não		Não
6/6/2017 13:52:36	18 a 25 anos de idade	Sim	Correio de voz	Pouco satisfeito	Chamada nativa	Talvez		Não
6/6/2017 14:00:32	18 a 25 anos de idade	Sim	Correio de voz, Voz para SMS	Nada satisfeito	Chamada nativa	Sim		Talvez
6/6/2017 14:02:26	26 a 40 anos de idade	Sim	Não aplicável		Chamada nativa	Talvez		Não
6/6/2017 14:02:34	18 a 25 anos de idade	Sim	Correio de voz	Satisfeito	Chamada nativa	Talvez		Não
6/6/2017 14:06:29	18 a 25 anos de idade	Sim	Correio de voz	Satisfeito	Chamada nativa	Sim		Não
6/6/2017 14:14:24	18 a 25 anos de idade	Não	Correio de voz	Satisfeito	Chamada nativa	Talvez		Talvez
6/6/2017 14:17:40	18 a 25 anos de idade	Sim	Voz para SMS	Pouco satisfeito	Chamada nativa	Talvez		Talvez
6/6/2017 14:21:15	18 a 25 anos de idade	Sim	Não aplicável		Chamada nativa	Talvez		Sim
6/6/2017 14:22:23	18 a 25 anos de idade	Sim	Correio de voz	Satisfeito	Chamada nativa	Sim		Talvez
6/6/2017 14:24:09	18 a 25 anos de idade	Sim	Correio de voz	Satisfeito	Chamada nativa	Sim		Talvez
6/6/2017 14:27:21	18 a 25 anos de idade	Sim	Correio de voz	Pouco satisfeito	Chamada nativa	Sim	não é necessário, é mais fácil ligar de novo mais tarde	Não
6/6/2017 14:28:55	18 a 25 anos de idade	Sim	Correio de voz	Satisfeito	Chamada nativa	Não		Não
6/6/2017 14:30:11	18 a 25 anos de idade	Sim	Correio de voz, Voz para SMS	Satisfeito	Chamada nativa	Sim		Não
6/6/2017 14:31:11	18 a 25 anos de idade	Sim	Correio de voz	Pouco satisfeito	Chamada nativa	Talvez		Não
6/6/2017 14:32:26	18 a 25 anos de idade	Não	Não aplicável		Chamada nativa	Talvez		Não
6/6/2017 14:38:43	18 a 25 anos de idade	Sim	Correio de voz, Voz para SMS	Satisfeito	Chamada nativa	Talvez		Não

Inquérito introdutório - Assistente virtual de chamadas recebidas

Corrimbo de data/hora	Idade	Já ouviu falar em correio de voz ou no sistema voz para SMS, ambas ferramentas que oferecem soluções para chamadas perdidas?	Qual ou quais das tecnologias já teve oportunidade de utilizar?	Em que medida ficou satisfeito com a sua qualidade de serviço?	Na altura de contactar alguém, opta por realizar uma chamada regular através da rede pública de telefonia comutada, ou liga através de aplicações VoIP, como o Skype ou o WhatsApp?	Consideraria a utilização de um assistente virtual que comunicaria com a pessoa que contactou o fim de averiguar o motivo do contacto e, por exemplo, criar automaticamente um lembrete na sua agenda numa hora em estivesse disponível e acordada com o outro utilizador?	Se respondeu negativamente à questão anterior, apresente um motivo para ter escolhido essa opção	Estaria disposto a pagar por um sistema como o assistente virtual apresentado?
6/6/2017 14:41:36	18 a 25 anos de idade	Sim	Correio de voz		Chamada nativa	Não	Se utilizar o hipotético assistente virtual estaria a perder contacto humano, algo que hoje em dia é cada vez mais questionável, uma vez que se julga estar a perder. Desta forma, ao implementar este tipo de assistentes estaríamos a caminhar para esse caminho.	Não
6/6/2017 14:42:19	18 a 25 anos de idade	Sim	Correio de voz	Pouco satisfeito	Chamada nativa	Sim		Talvez
6/6/2017 14:45:06	18 a 25 anos de idade	Sim	Voz para SMS	Satisfeito	Chamada nativa	Não		Não
6/6/2017 14:50:43	18 a 25 anos de idade	Sim	Correio de voz, Voz para SMS	Muito Satisfeito	Chamada nativa	Talvez		Talvez
6/6/2017 14:51:08	18 a 25 anos de idade	Não	Não aplicável		Chamada nativa	Talvez		Talvez
6/6/2017 14:51:19	18 a 25 anos de idade	Sim	Correio de voz, Voz para SMS	Satisfeito	Chamada nativa	Talvez		Talvez
6/6/2017 14:52:08	18 a 25 anos de idade	Sim	Correio de voz, Voz para SMS	Pouco satisfeito	Chamada nativa	Talvez		Talvez
6/6/2017 15:04:57	18 a 25 anos de idade	Sim	Correio de voz, Voz para SMS	Satisfeito	Aplicações VoIP	Sim		Não
6/6/2017 15:07:25	26 a 40 anos de idade	Sim	Não aplicável		Chamada nativa	Talvez		Não
6/6/2017 15:09:33	18 a 25 anos de idade	Sim	Correio de voz, Voz para SMS	Nada satisfeito	Chamada nativa	Não	Se foi feita uma chamada, significa que existe a necessidade de contacto pessoal directo. Além do mais, há um grande problema de falta de privacidade.	Não
6/6/2017 15:12:35	18 a 25 anos de idade	Sim	Correio de voz	Satisfeito	Chamada nativa	Sim		Não
6/6/2017 15:12:48	26 a 40 anos de idade	Sim	Não aplicável		Chamada nativa	Não	Muito complicado. Não preciso	Não
6/6/2017 15:21:37	18 a 25 anos de idade	Sim	Não aplicável		Chamada nativa	Sim		Talvez
6/6/2017 15:21:59	18 a 25 anos de idade	Sim	Correio de voz	Satisfeito	Chamada nativa	Talvez		Não
6/6/2017 15:35:01	18 a 25 anos de idade	Sim	Correio de voz	Nada satisfeito	Chamada nativa	Sim		Talvez
6/6/2017 15:39:58	41 a 60 anos de idade	Sim	Correio de voz	Pouco satisfeito	Aplicações VoIP	Sim		Talvez
6/6/2017 15:41:33	18 a 25 anos de idade	Sim	Correio de voz	Satisfeito	Chamada nativa	Talvez		Talvez

Inquérito introdutório - Assistente virtual de chamadas recebidas

Carimbo de data/hora	Idade	Já ouviu falar em correio de voz ou no sistema voz para SMS, ambas ferramentas que oferecem soluções para chamadas perdidas?	Qual ou quais da tecnologias já teve oportunidade de utilizar?	Em que medida ficou satisfeito com a sua qualidade de serviço?	No altura de contactar alguém, opta por realizar uma chamada nativa, isto é, uma chamada regular através da rede pública de telefonia comutada, ou liga através de aplicações Voip, como o Skype ou o WhatsApp?	Consideraria a utilização de um assistente virtual que comunicaria com a pessoa que o contactou o fim de averiguar o motivo do contacto e, por exemplo, criar automaticamente um lembrete na sua agenda numa hora em estivesse disponível e acordada com a outra utilizador?	Se respondeu negativamente à questão anterior, apresente um motivo para ter escolhido essa opção	Estaria disposto a pagar por um sistema como o assistente virtual apresentado?
6/6/2017 15:53:18	18 a 25 anos de idade	Sim	Correio de voz	Muito Satisfeito	Chamada nativa	Talvez		Talvez
6/6/2017 16:03:13	18 a 25 anos de idade	Não	Não aplicável		Chamada nativa	Talvez		Talvez
6/6/2017 16:15:46	18 a 25 anos de idade	Sim	Não aplicável		Chamada nativa	Talvez		Não
6/6/2017 16:29:12	18 a 25 anos de idade	Sim	Correio de voz	Satisfeito	Chamada nativa	Talvez		Não
6/6/2017 16:46:32	18 a 25 anos de idade	Sim	Correio de voz	Pouco satisfeito	Chamada nativa	Sim		Não
6/6/2017 17:06:25	18 a 25 anos de idade	Sim	Não aplicável		Chamada nativa	Talvez		Não
6/6/2017 17:13:31	18 a 25 anos de idade	Sim	Correio de voz	Satisfeito	Chamada nativa	Sim		
6/6/2017 17:39:13	18 a 25 anos de idade	Sim	Não aplicável		Aplicações Voip	Sim		
6/6/2017 18:35:46	18 a 25 anos de idade	Sim	Correio de voz	Pouco satisfeito	Chamada nativa	Não	Acho Impessoal.	
6/6/2017 18:45:06	18 a 25 anos de idade	Sim	Correio de voz, Voz para SMS	Nada satisfeito	Aplicações Voip	Sim		
6/6/2017 19:03:05	18 a 25 anos de idade	Sim	Correio de voz	Satisfeito	Chamada nativa	Talvez		
6/6/2017 19:18:34	26 a 40 anos de idade	Sim	Não aplicável		Aplicações Voip	Talvez		
6/6/2017 20:02:55	26 a 40 anos de idade	Sim	Correio de voz	Nada satisfeito	Aplicações Voip	Sim		
6/6/2017 20:15:11	18 a 25 anos de idade	Sim	Correio de voz	Satisfeito	Chamada nativa	Talvez		
6/6/2017 20:22:51	18 a 25 anos de idade	Sim	Correio de voz	Satisfeito	Aplicações Voip	Talvez		
6/6/2017 20:32:06	18 a 25 anos de idade	Sim	Correio de voz	Pouco satisfeito	Chamada nativa	Sim		
6/6/2017 21:04:23	18 a 25 anos de idade	Sim	Correio de voz		Chamada nativa	Sim		
6/6/2017 21:45:22	18 a 25 anos de idade	Sim	Não aplicável		Chamada nativa	Sim		
6/6/2017 23:01:21	18 a 25 anos de idade	Sim	Correio de voz, Voz para SMS	Satisfeito	Aplicações Voip	Sim		
6/6/2017 23:01:50	26 a 40 anos de idade	Sim	Correio de voz		Chamada nativa	Talvez		
6/7/2017 0:19:59	18 a 25 anos de idade	Sim	Correio de voz	Pouco satisfeito	Chamada nativa	Sim		
6/7/2017 0:26:13	18 a 25 anos de idade	Não	Não aplicável		Chamada nativa	Sim		
6/7/2017 1:23:51	18 a 25 anos de idade	Sim	Correio de voz	Satisfeito	Chamada nativa	Sim		
6/7/2017 9:46:17	18 a 25 anos de idade	Sim	Correio de voz, Voz para SMS	Satisfeito	Chamada nativa	Talvez		
6/7/2017 10:38:50	18 a 25 anos de idade	Sim	Não aplicável		Aplicações Voip	Talvez		
6/7/2017 11:03:00	18 a 25 anos de idade	Sim	Correio de voz	Satisfeito	Chamada nativa	Talvez		
6/7/2017 11:14:56	18 a 25 anos de idade	Sim	Correio de voz	Satisfeito	Chamada nativa	Talvez		

Inquérito introdutório - Assistente virtual de chamadas recebidas

Anexo D

Inquérito final - Assistente virtual de chamadas recebidas

D.1 Inquérito

Assistente virtual para dar resposta a chamadas perdidas

O objetivo deste questionário é avaliar o interesse de um sistema que permita dar resposta às chamadas perdidas por um utilizador que se encontra ocupado.

*Obrigatório

Após testar o assistente virtual anterior e a aplicação desenvolvida, estaria disposto a pagar por um sistema deste género? *

- Sim
 Não
 Talvez

Já ouviu falar nas tecnologias de Correio de Voz e Voz para SMS? *

- Sim
 Não

Se respondeu afirmativamente à questão anterior, acha que o sistema elaborado seria uma alternativa capaz de substituir essas tecnologias?

- Sim
 Não
 Talvez

Apresente uma sugestão para melhorar este sistema.

A sua resposta

SUBMITER

D.2 Respostas

Carimbo de data/hora	Após testar o assistente virtual anterior e a aplicação desenvolvida, estaria disposto a pagar por um sistema deste género?	Já ouviu falar nas tecnologias de Correio de Voz e Voz para SMS?	Se respondeu afirmativamente à questão anterior, acha que o sistema elaborado seria uma alternativa capaz de substituir essas tecnologias?	Apresente uma sugestão para melhorar este sistema.
6/6/2017 13:13:50	Sim	Sim	Sim	Melhorar a comunicação do assistente. Não tem emoção.
6/6/2017 13:19:29	Talvez	Sim	Talvez	Permitir dar mais informação ao assistente.
6/6/2017 13:28:59	Sim	Sim	Sim	A app podia ser melhorada.
6/6/2017 13:50:46	Talvez	Sim	Talvez	Melhora o processo de pergunta resposta para algo mais direto, por exemplo.
6/6/2017 14:07:23	Sim	Sim	Sim	Talvez adicionar mais caso de utilização à aplicação.

Anexo E

Inquérito final - Assistente virtual para gestão de audioconferências telefônicas

E.1 Inquérito

Assistente virtual para gestão de audioconferências telefônicas

O objetivo deste questionário consiste em averiguar se a prova de conceito que acabou de testar poderá ser uma alternativa às soluções existentes e se acrescenta valor as essa mesma soluções.

*Obrigatório

Considera que a prova de conceito que acabou de testar teria utilidade no seu dia-a-dia na empresa? *

- Sim
 Não
 Talvez

Estaria disposto a pagar por um produto como este? *

- Sim
 Não
 Talvez

Que sugestões apresenta para melhorar este produto futuramente? *

À sua resposta

SUBMITER

E.2 Respostas

<i>Carimbo de data/hora</i>	<i>Considera que a prova de conceito que acabou de testar teria utilidade no seu dia-a-dia na empresa?</i>	<i>Estaria disposto a pagar por um produto como este?</i>	<i>Que sugestões apresenta para melhorar este produto futuramente?</i>
6/8/2017 16:06:48	Sim	Talvez	Aumentar o leque de frases possíveis para cada tarefa.
6/8/2017 16:26:53	Talvez	Não	Acréscetar funcionalidades de gestão de conferências e a eficácia em entender o que se diz.
6/8/2017 16:48:25	Sim	Sim	O bot devia perceber mais frases e manter mais contexto do que lhe dizemos.
6/8/2017 17:03:58	Talvez	Não	Devia permitir uma conversação mais aberta e, por exemplo, notificar-te das conferências que terás em breve.
6/8/2017 17:35:41	Talvez	Não	Podias aumentar o vocabulário do bot. É muito direto, tipo pergunta-resposta.